# UNIT 3 PROJECT PLANNING AND CONTROL

**Structure** **Page Nos.**



## 3.0 INTRODUCTION

The important aspects of Project Management are discussed in this unit. An effective Manager is essential for successful project execution. It is important when organising a project to ensure that every person knows his or her role in the project and is aware of corporate objectives. Various charting techniques are discussed as a part of Project scheduling. A schedule has two primary functions. It is both a plan and a device for measuring progress. This unit deals with Project Management concepts like project standards, conflicts, modifications, project outsourcing and completion of the project.

## 3.1 OBJECTIVES

After going through this unit, you should be able to:

- know about the project planning;
- schedule a project, and
- know various project standards.

## 3.2 PROJECT PLANNING AND CONTROL

The planning, design, and installation of a system termed a project and is directed by a project leader who uses the available resources to produce a new or better system for the organisations.

In large companies, the installation of a computer system may take years and involve thousands of people. Planning for smaller projects also requires effective management controls to realise the desired results. Thus, project planning for any company has the following four main steps:

1. Organising the resources available for the project.

2. Scheduling the events of the project.
3. Evaluating the progress.
4. Establishing standards for the project.

An effective manager is essential for successful project planning. The techniques of project planning are not a substitute for good management, but merely a tool to be used by managers to achieve better results. Only effective management can complete the project on time, within budget, and with satisfactory results.

To achieve a goal, there should be a strong determination and it should always be in one's mind. Consequently, defining the objectives is the first action taken in any project. Along with defining objectives, corporate management must assign priorities to various projects that are underway and clarify the relationship between systems projects and existing systems. A systems project requires extensive interaction between developers and users. Users are, of course, preoccupied with day-to-day operations, and it cannot be assumed that they will be enthusiastic about participating in a system study. However, only when management clearly defines the importance of user participation in systems development, then the necessary co-operation can be expected from users.

Many systems are designed and implemented through project teams headed by a project leader. The team may be relatively small during the feasibility stage, consisting of a few highly qualified systems people, users, and managers. During the design phase, when more detailed work is required, the size of the team normally increases. A typical project team would have a senior systems analyst as project leader, supported by junior analysts, programmers. When the project has been completed, the team is disbanded and each member is reassigned to another project.

To organise a project, the project leader must determine who all are required for the project, when they are available, and for how long their services can be expected. The key people required in a systems project are often the key people in the day-to-day operations of an organisation, and they probably will have to continue their normal routines as they participate in the systems project. In organising their efforts, the project leader must avoid scheduling important project activities when the users are very busy with normal duties. For users who are "always busy", plans must be made to utilise overtime or/and shift personnel, to free the key systems users for participation in the project.

It is important when organising a project to ensure that every person knows his or her role in the project and is aware of corporate objectives. This is accomplished through formal training as well as informal conversations.

The project leader is solely responsible for the completion of a project, but obviously cannot do it alone. Responsibilities must be defined precisely, and overlapping responsibilities avoided. When the phase of a project breaks down, is behind schedule, or is over budget, the leader of a well-organised project will be able to identify the person who can provide information and, perhaps, the solution to the problem.

Besides organising people, the project leader must budget money and order equipment. Acquainting people with their responsibilities and enabling them to discharge these responsibilities is the essence of leadership.

## 3.3 PROJECT SCHEDULING

The charting techniques are the scheduling tools of the project planner. Even the simplest project should be charted so that progress can be measured. The Gantt chart is effective in simple projects, especially when the interrelationships among events are not too complex. Complicated scheduling usually requires a PERT chart.

Included in the tasks to be scheduled in a normal data processing project are systems design, programming, file and data base creation, program testing, conversion, documentation, and training. The project planner must anticipate problem areas that inevitably develop and allow for delays in obtaining approvals at key check-points in the project.

Projects are organised into modules, or segments, of related tasks. Modular planning has following advantages: it facilitates assigning responsibility and measuring processes, and it further allows systems analysts to work in concentrated areas of projects so they can master every aspect of that portion of the system.

A schedule must be flexible because unexpected events may occur that may affect the schedule of development of the system. Seldom do systems projects meet the original schedule at each milestone. This does not imply that schedules are made to be broken, but a schedule cannot be so rigid that when the unexpected occurs, subsequent events cannot be rescheduled.

A schedule has two primary functions, it is both a plan and a device for measuring progress. The key steps in a schedule are called milestones, or checkpoints. As the project progresses, the date on which each milestone is completed is compared with the date for which it was projected. In any project, frequent progress reviews take place in which the status of events is reported and evaluated. If, in the original planning stage, the important milestones were anticipated correctly, reporting them as completed, late, ahead of schedule, or on time has significance to the status of the project. The status of fringe events is relatively insignificant. Here, the value of the PERT network as a tool for determining the relative importance of milestones is apparent.

The status of projects is often reported in terms of percentage of completion. As a simplified reporting device, this is effective and allows easy communication with top management. The problem with reporting of percentage of completion is that events on the critical path are not emphasised. A project may have 90 percent of its events complete, but if one of the incomplete events is on the critical path and is two years late, then the project may be in serious trouble.

Accurate scheduling requires extensive experience. The novice scheduler almost always does not allow enough time for activities. Even when requirements are carefully gathered, as in PERT network, some areas of delay are not apparent. For example, an inexperienced person may not realise that equipment or forms are often delivered late. Moreover, lead time must be provided for approvals in several areas, such as file design and I/O forms.

At the outset of the project, the project leader must determine the reporting format. Is status to be measured in days, weeks, and tens of weeks, or percentage of each job done? When are status reports to be made? Are reports to be made orally, in writing, or in chart form?

When a project is behind schedule, corrective steps must be taken. Establishing milestones is meaningless unless the project manager can enforce adherence to schedule. Enforcement is a normal managerial duty. If a project leader cannot enforce a schedule, someone else should be leading the project. If one area is consistently behind schedule, or over budget, the project leader must discuss the problem with the individuals responsible and take corrective action. The following are variety of options that are open to the project leader:

- Increase the budget.
- Increase manpower in the form of overtime or additional people.
- Add equipment.
- Change priorities.
- Replace the individual responsible.

The project leader must determine the real cause of unsatisfactory progress. Perhaps, it is a budgetary or personnel problem. Too often, a major cause of apparently unsatisfactory performance is that the original schedule estimates were wrong, and that progress is as good as can be expected under the circumstances.

Projects have many target dates, but few deadlines. The project leader must distinguish one from the other. When target dates are missed, there may be some grumbling, but missed deadlines result in financial loss to the organisation.

At the outset of a project, the project leader should not become committed to the unreasonable target dates or deadlines. Unreasonable deadlines are costly because unnecessary effort is made to meet them. Moreover, failing to meet them often creates morale problems.

The project leader must remember that this is not the only project under way and delays will occur routinely simply because another, more important project may have to be handled first. Schedules are highly dependent upon priorities and should be planned accordingly.

**The Problem of Capacity**

The problem of capacity occurs when a system component is not large enough. Capacity problems are specially common in organisations that experience peak periods of business. During peak periods, inadequate processing capacity, transmission capacity, storage capacity, staff capacity, and the like may all exist. Capacity problems are also evident in rapidly growing organisations. With growth, smaller-capacity equipment soon becomes too small; smaller staff groups soon become overworked. In either case, some expansion is needed to handle the increasing volume of business.

Many system problems are directed at solving capacity problems. Since it is often difficult to justify the purchase of new equipment or the hiring of new staff, people tend to put off such decisions until the very last moment (may not be applicable to all organisations). Consequently, when the systems group is contacted, the problem of capacity is easy to spot; the difficulty, however, lies in knowing how to handle the problem. For example, an analyst might be forced to suggest a short-term solution to the problem. This is done to gain time towards the formulation of a longer-term solution. For instance, an analyst might recommend: "Let's hire five part-time employees to help us get through the peak period." When a short-term approach fails, the analyst may be tempted to implement a quick-fix computer-based solution. Unfortunately, this solution carries with it, the associated danger of creating an even more severe system problem in the near future.

### The Problem of Throughput

The problem of Throughput may be viewed as the reverse of the problem of capacity. Throughput deals with the efficiency of a system. If system capacity is high and production is low, a problem of throughput occurs.

Consider the following example:

Five programmers are assigned to a fairly straightforward programming assignment consisting of 10,000 lines of computer code. After thirty days of coding, the programming team is evaluated. It is discovered that they have completed 6000 usable lines of code. Now, if each programmer worked eight hours a day, a total of 1200 hours would have been expended on the project and the average production rate for each programmer would be 5 lines of code per hour (6000 lines divided by 1200 hours). These findings might lead the analyst to conclude that there is a problem of throughput.

Similar to the problem of capacity, the problem of throughput may be much easier to spot than to treat. When repeated equipment breakdowns lead to low rates of production (and when the equipment has been purchased and cannot be returned), an organisation can badger the vendor into fixing the equipment but can achieve little more short of legal action. Likewise, when groups of people exhibit low rates of production, such as the five-person programming team, the problem becomes even more complicated. Of course, a manager must be able to determine the root of the problem for any improvement in throughput.

### Evaluating the Problem

Suppose that a problem has been identified. The next step is problem evaluation, which consists of asking the following questions: Why is it important to solve the problem? What are possible solutions to the problem? What types of benefits can be expected once the problem is solved? There will be times when an analyst will recommend that no project be started to resolve a problem, as the next example demonstrates.

Suppose that an analyst discovers that the real problem lies with the supervisor of an area. Because of mistakes made by him, the throughput rate is 20 per cent less than that had been expected. However, suppose, next, that the supervisor is new to the job, is smart enough to realise where mistakes were made, and knows how not to repeat them in the future. Given this situation, the analyst might close the book on this project, recommending that no action be taken at this time.

Consider a different set of circumstances. Suppose that an analyst determines that a problem of low throughput can be traced to a computer printer. Suppose, further that the problem must be corrected. Once the problem has been identified, the analyst would prepare a solutions table to list possible problem solutions and the expected benefits from each. Sometimes, the best solution is not at all evident. The analyst might recommend that further study is required to determine which of the possible solutions is best.

In this section, we have spent considerably more time examining how an analyst identifies a problem compared with how the problem is evaluated. This uneven split also occurs in practice. As a general rule, analysts spend 75 per cent of the project-definition phase of analysis defining the problem and 25 per cent evaluating and documenting their findings. Note also that we have limited our discussion to major types of system problems. Because of this limitation, you might ask, "What about the problems of communication" among group members; with management; about system security? Does the analyst evaluate these types of problems? Although, our discussion has been restricted to more technical problems, individual or group problems also occur in a systems environment and require identification and evaluation.

Still another limitation is the coverage given to determining the feasibility of taking some action to solve a problem. The concept of feasibility entails the joint questions of "Can something be done?" and, if so, "Should it be done given a particular set of circumstances?"

Another limitation is the coverage given to tools whom the analyst can use to identify and evaluate system problems. These tools are needed when the problems are not self-evident.

Organisations face various types of problems during their course of operations and come across opportunities or situations, which could be converted into profitable solutions. Whenever there is an opportunity and /or problem in the existing system of operations or when a system is being developed for the first time, the organisation considers designing a new system for information processing.

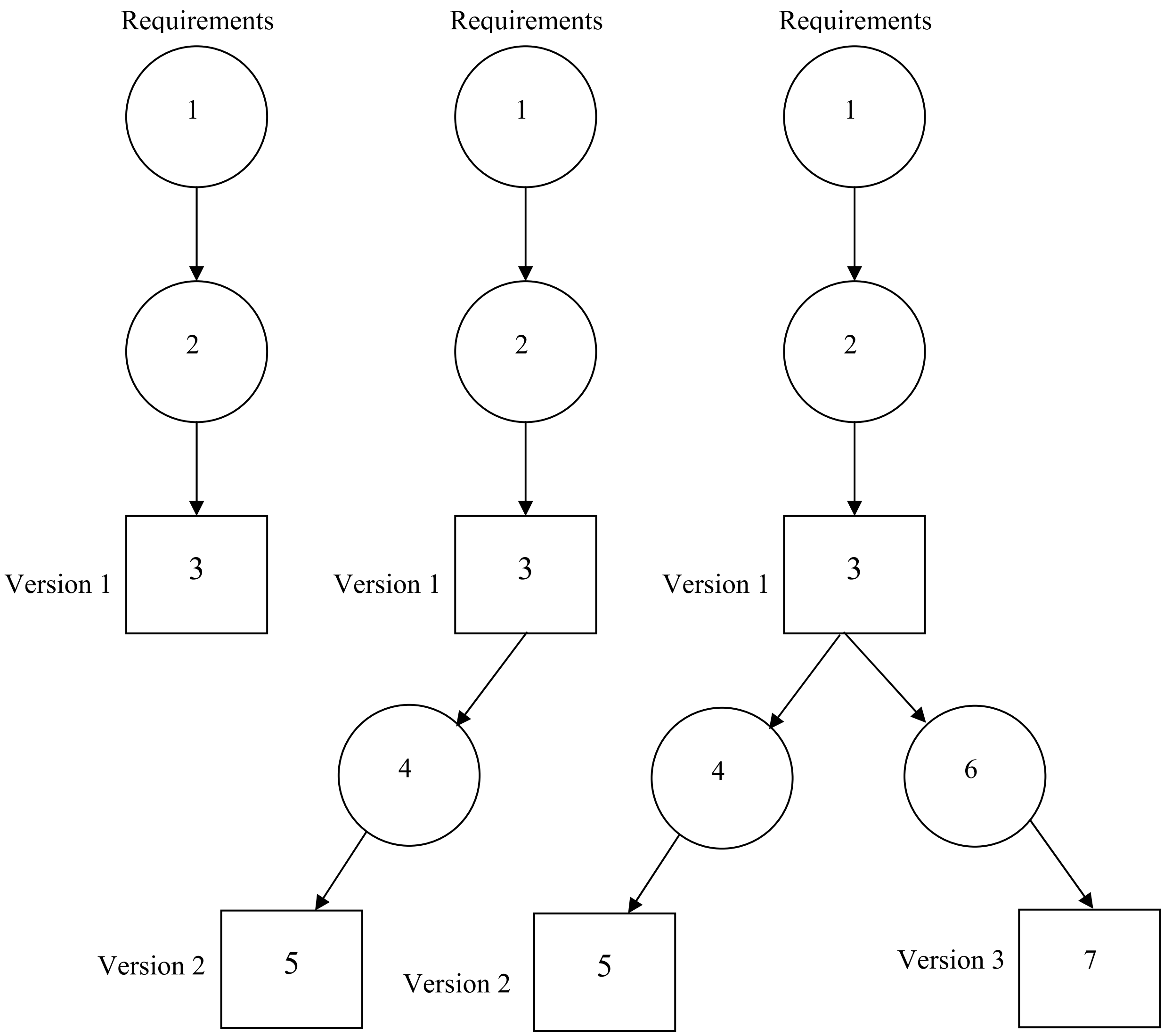


**Figure 3.1 : Sequential Design of a Program Family**

**Sources of Problem/Opportunity**

Organisations usually face problems or have opportunity due to the following:

- Launch of a new product or commencement of a plant or branch

- A new market or new process
- Failure of an existing system
- Inefficiency of an existing system
- Structural error in the existing system, etc.

Thus, a thorough analysis of the situation needs to be made. Not only the above listed reasons, but there may exist some additional reasons that arise due to the specific organisation.

**Problem Identification and Definition**

For identifying problems/opportunities, we consider the following:

- The performance of the system
- The information being supplied and its form
- The economy of processing
- The control of information processing
- Security of Data and Software.

Scheduling of a software project can be correlated to prioritising various tasks (jobs) with respect to their cost, time and duration. Scheduling can be done with resource constraint or time constraint in mind. Depending upon the project, scheduling methods can be static or dynamic in implementation.

**Scheduling Techniques**

The following are various types of scheduling techniques in software engineering are:

- **Work Breakdown Structure:** The project is scheduled in various phases following a bottom-up or top-down approach. A tree-like structure is followed without any loops. At each phase or step, milestone and deliverables are mentioned with respect to requirements. The work breakdown structure shows the overall breakup flow of the project and does not indicate any parallel flow. *Figure 3.2* depicts an example of a work breakdown structure.

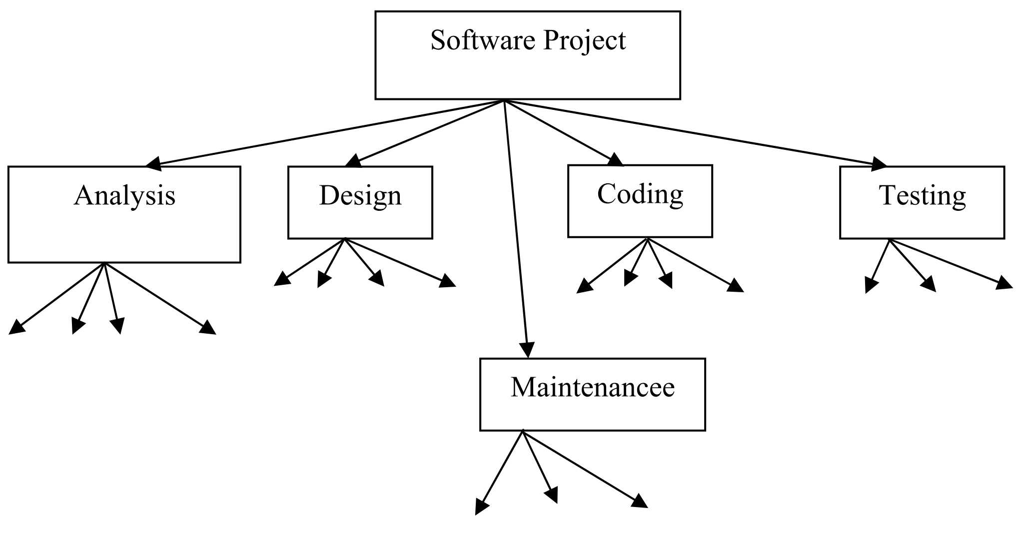


**Figure 3.2: An example WBS**

The project is split into requirement and analysis, design, coding, testing and maintenance phase. Further, requirement and analysis is divided into R1,R2 ….

Rn; design is divided into D1, D2, D3, ….,Dn; coding is divided into C1,C2..Cn; testing is divided into T1,T2.. Tn; and maintenance is divided into M1, M2.. Mn. If the project is complex, then further sub-division is done. Upon the completion of each stage, integration is done.

- **Flow Graph :** Various modules are represented as nodes with edges connecting nodes. Dependency between nodes is shown by flow of data between nodes. Nodes indicate milestones and deliverables with the corresponding module implemented. Cycles are not allowed in the graph. Start and end nodes indicate the source and terminating nodes of the flow. *Figure 3.3* depicts a flow graph.

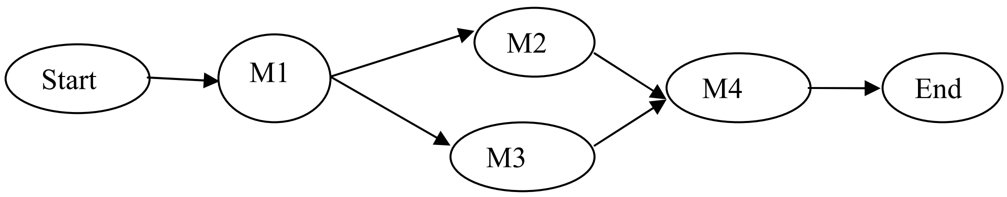


**Figure 3.3 : Flow Graph**

M1 is the starting module and the data flows to M2 and M3. The combined data from M2 and M3 flow to M4 and finally the project terminates. In certain projects, time schedule is also associated with each module. The arrows indicate the flow of information between modules.

- **Gantt Chart or Time Line Charts :** A Gantt chart can be developed for the entire project or a separate chart can be developed for each function. A tabular form is maintained where rows indicate the tasks with milestones and columns indicate duration ( weeks/months) . The horizontal bars that spans across columns indicate duration of the task. *Figure 3.4* depicts a Gantt Chart. The circles indicate the milestones.

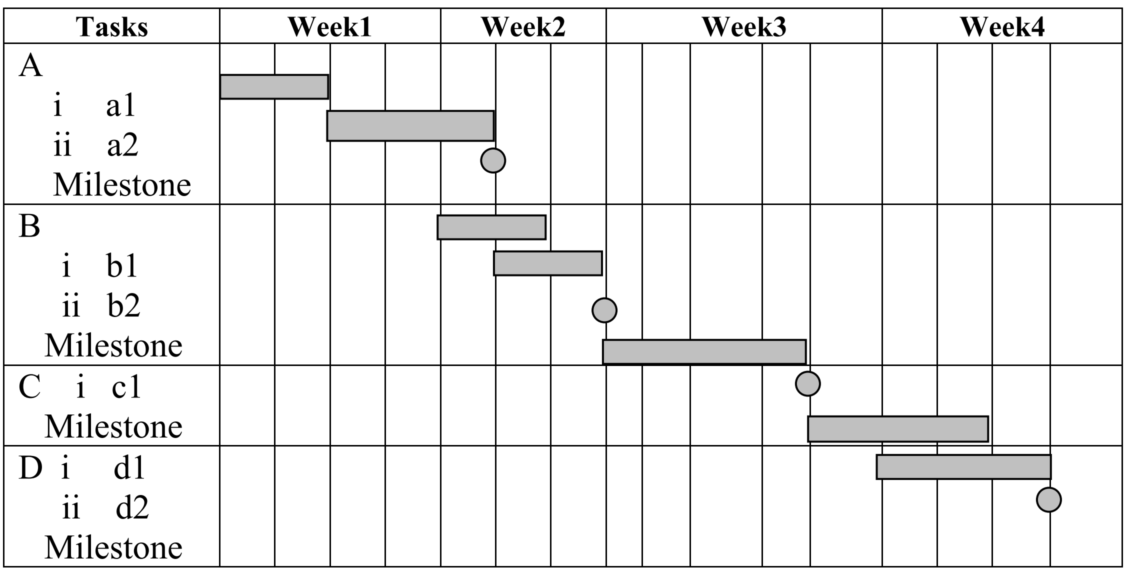


**Figure 3.4: Gantt Chart**

- **Program Evaluation Review Technique :** Mainly used for high-risk projects with various estimation parameters. For each module in a project, duration is estimated as follows:

1. Time taken to complete a project or module under normal conditions, *tnormal.*

2. Time taken to complete a project or module with minimum time (all resources available), *tmin* .

3. Time taken to complete a project or module with maximum time (resource constraints), *tmax.*

4. Time taken to complete a project from previous related history, *thistory.*

An average of tnormal, tmin, tmax and thistory is taken depending upon the project. Sometimes, various weights are added as 4*tnormal, 5*tmin, 0.9*tmax and 2*thistory to estimate the time for a project or module. Parameter fixing is done by the project manager.

☞ **Check Your Progress 1**

1) The problem of ________ occurs when a system component is not large enough.

2) The problem of __________ may be viewed as the reverse of the problem of capacity.

3) For scheduling of projects, the ________ is effective in simple projects, especially when the interrelationships among events are not too complex.

## 3.4 PROJECT STANDARDS

Initially, the project must establish the objectives of each phase. Each phase must be of a controllable size, and every task within the phase must be spelled out. The project leader and the individual responsible for the phase must agree upon the human skills and other resources required to accomplish each task. They also must agree upon the expected outputs from these tasks. Ultimately, they must decide upon the measurable outputs that will be examined throughout the project to evaluate progress.

For each phase of the project, the status of time to complete tasks, personnel utilisation, and unforeseen problems should be reported to the project leader. This report is then reviewed by the project leader and the managers concerned to evaluate progress. Also evaluated is the quality of work, as reflected in the outputs from each phase. This periodic review has four main tasks:

1. Review project progress.
2. Analyse the impact of delays on the entire project.
3. Examine any problems existing in the quality of the data.
4. Anticipate developing problems.

### 3.4.1 Project Conflicts

Disagreements arise in most systems projects. Lack of clearly defined objectives and standards for acceptable performance are major causes of disagreement. A project leader should be sure to have these definitions in writing before undertaking the project.

The project manager is responsible for settling disputes that arise within the scope of the project. This is usually done by calling together all concerned parties so that the disputes can be resolved. When the parties cannot reach agreement, the project manager will have the final say in the matter.

Implied in this procedure is that all the parties are aware of what authority the project leader and project manager had. Too often, this may be vague. In general, however, the project manager must have the backing of whoever originally requested the project. When top management is the requester, the authority of the project leader is rarely questioned. When the project has been requested by a specific department, branch, or individuals representing areas of equal or greater status than the project leader, substantial challenges to the project leader's authority may occur. An irony regarding systems projects is that large projects are usually more successful than smaller ones because the request comes from top people up in the corporate structure.

### 3.4.2 Project Modifications

For a variety of reasons, changes must sometimes be made in a project while it is under way. Requests for changes must be evaluated carefully, according to following criteria:

1. The impact on the present schedule.
2. The impact on the resources available for the project.
3. The cost
4. The effect on the deadlines for the system.

Poor planning is the primary cause for changes in existing projects. It has four other consequences as well :

***Obsolescence:*** The systems in an organisation tend to become obsolete quickly.

***Poor Follow-up:*** The responsibility for follow-up falls upon corporate executives instead of project leaders.

***Inflexibility:*** Systems become inflexible, so that minor modifications result in extensive program writing.

***Lack of Documentation:*** Procedure writing and documentation are, in general, neglected.

### 3.4.3 Completing The Project

Projects often run on schedule and within budget for most of their existence, only to fall behind during the final stages. Perhaps, the last 10 per cent of a project is the most difficult to complete, because, enthusiasm wanes at that point and people look forward to newer challenges. Often, too, the final phases of a project are least interesting. Documentation must be updated and completed, detailed problems solved and annoyances cleared up. Discipline on the part of the project team is required through out the project that will lead to a complete and functioning system.

## 3.5 PROJECT OUTSOURCING

Sooner or later, every company that develops computer software asks a fundamental question: "Is there a way that we can get the software and systems that we need at a lower price?" The answer to this question is not a simple one, and the emotional discussions that occur in response to the question always lead to a single word: "outsourcing".

In concept, outsourcing is extremely simple. Software Engineering activities are contracted to a third party who does the work at a lower cost, and hopefully, of a higher quality. Software work conducted within a company is reduced to a contract management activity.

The decision to outsource can be either strategic or tactical. At the strategic level, business managers consider whether a significant portion of all software work can be contracted to others. At the tactical level, a project manager determines whether part or all of a project can be best accomplished by subcontracting some portion of the software work.

Regardless of the breadth of focus, the outsourcing decision is often a financial one.

On the positive side, cost savings can usually be achieved by reducing the number of software people and the facilities (e.g., computers, infrastructure) that support them. On the negative side, a company loses some control over the software that it needs. Since software is a technology that differentiates its systems, services, and products, a company runs the risk of putting the fate of its competitiveness into the hands of a third party.

The trend towards outsourcing will undoubtedly continue. The only way to blunt the trend is to recognise that software work in the twenty-first century may be extremely competitive at all levels.

### ☞ Check Your Progress 2

1) At the tactical level, a ________ determines whether part or all of a project can be best accomplished by subcontracting some portion of the software work.

## 3.6 THE SOFTWARE PROJECT PLAN

Planning is very important in every aspect of development work. Good managers carefully monitor developments at various phases. Improper planning leads to failure of the project. Software project plan can be viewed as the following :

1. Within the organisation: How the project is to be implemented? What are various constraints (time, cost, staff) ? What is market strategy?
2. With respect to the customer: Weekly or timely meetings with the customer with presentations on status reports. Customer feedback is also taken and further modifications and developments are done. Project milestones and deliverables are also presented to the customer.

For a successful software project, the following steps can be followed:

- Select a project
    - Identifying project's aims and objectives
    - Understanding requirements and specification
    - Methods of analysis, design and implementation
    - Testing techniques
    - Documentation

- Project milestones and deliverables
- Budget allocation
  - Exceeding limits within control
- Project Estimates
  - Cost
  - Time
  - Size of code
  - Duration
- Resource Allocation
  - Hardware
  - Software
  - Previous relevant project information
  - Digital Library
- Risk Management
  - Risk Avoidance
  - Risk Detection
  - Risk Control
  - Risk Recovery
- Scheduling techniques
  - Work Breakdown Structure
  - Activity Graph
  - Critical path method
  - Gantt Chart
  - Program Evaluation Review Technique
- People
  - Staff Recruitment
  - Team management
  - Customer interaction
- Quality control and standard

All of the above methods/techniques are not covered in this unit. The student is advised to study references for necessary information.

## 3.7 SUMMARY

To achieve a goal, there should be a strong determination and it should always be in one's mind. Consequently, defining the objectives is the first action taken in any project. Along with defining objectives, corporate management must assign priorities to various projects that are underway and clarify the relationship between systems projects and existing systems. A systems project requires extensive interaction between developers and users. Users are, of course, preoccupied with day-to-day operations, and it cannot be assumed that they will be enthusiastic about participating in a system study. However, only when management clearly defines the importance of user participation in systems development, then the necessary cooperation can be expected from users.

A schedule has two primary functions, it is both a plan and a device for measuring progress. The key steps in a schedule are called milestones, or checkpoints. As the project progresses, the date on which each milestone is completed is compared with the date for which it was projected. In any project, frequent progress reviews take place in which the status of events is reported and evaluated. If, in the original planning stage, the important milestones were anticipated correctly, reporting them as completed, late, ahead of schedule, or on time has significance to the status of the project. The status of fringe events is relatively insignificant. Here, the value of the PERT network as a tool for determining the relative importance of milestones is apparent.

The project manager is responsible for settling disputes that arise within the scope of the project. This is usually done by calling together all concerned parties so that the disputes can be resolved. When the parties cannot reach agreement, the project manager will have the final say in the matter.

## 3.8 SOLUTIONS /ANSWERS

### Check Your Progress 1

1) capacity
2) throughput
3) Gantt chart

### Check Your Progress 2

1) project manager

## 3.9 FURTHER READINGS

1) *Software Engineering,* Ian Sommerville; Sixth Edition, 2001, Pearson Education.

2) *Software Engineering – A Practitioner's Approach*, Roger S. Pressman; McGraw-Hill International Edition.

### Reference websites

- **http://www.rspa.com**
- **http://www.ieee.org**
- **http://www.ncst.ernet.in**