Unit IV: Risk Management

Risk, Risk categories, Identification, Assessment, Planning and management, PERT technique, Monte Carlo approach

Risk:

"An uncertain event or condition that, if occurs, has a positive or negative effect on a projects objectives"

or

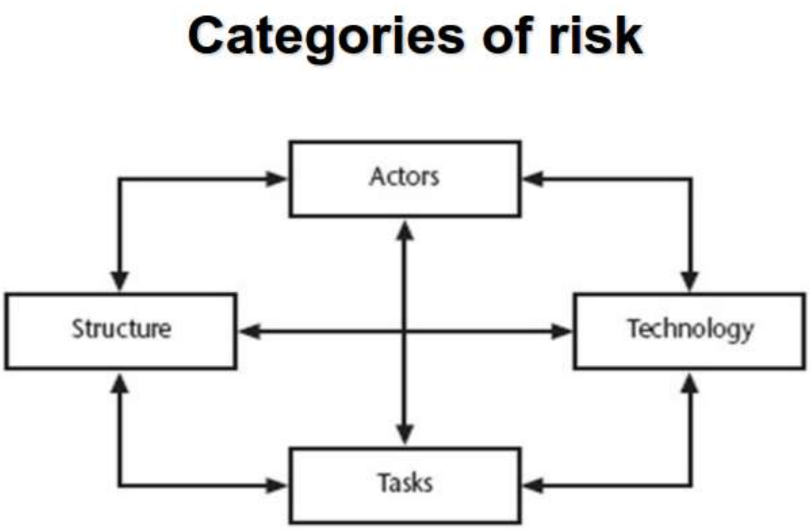
Risk is an expectation of loss, a potential problem that may or may not occur in the future. It is generally caused due to lack of information, control or time.

Or

A possibility of suffering from loss in software development process is called a software risk.

Risk categories:

The below model represents a sociotechnical model of risk.



The box labeled **"Actors"** refers to all the people involved in the development of the application. These include the various development specialists, the different user groups, and managers with differing responsibilities. A typical risk in this area is that high staff turnover leads to information being lost.

The box labeled **"Technology"** refers to both the technology used to implement the application and that embedded in the delivered products. **"Structure"** refers to management structures and systems including those affecting planning and control. **"Tasks"** relates to the work to be carried out.

Identification:

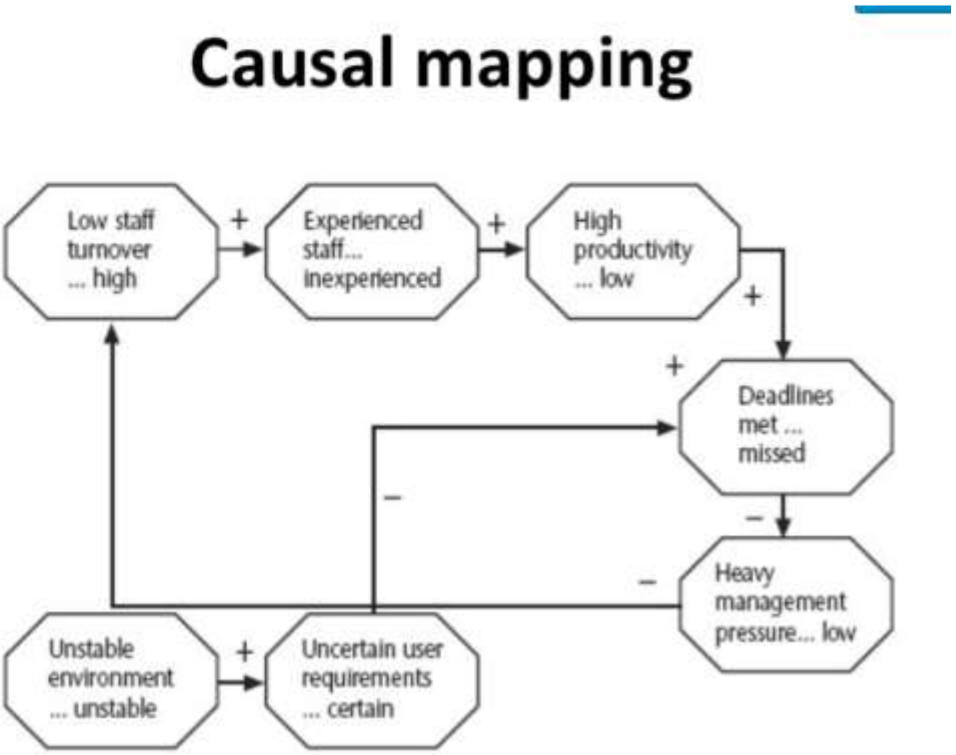
The two approaches to the identification of risks are the use of **checklists** and **brainstorming**.

Checklists: Checklists for risk identification can be developed based on historical information and knowledge that has been accumulated frfom previous similar projects and from other sources of information. **One advantage of using a checklist is that risk identification is quick and simple. One disadvantage is that it is impossible to build an exhaustive checklist of risks, and the user may be effectively limited to the categories in the list.** Care should be taken to explore items that do not appear on a standard checklist if they seem relevant to the specific project. The checklist should itemize all types of possible risks to the project. It is important to review the checklist as a formal step of every project-closing procedure to improve the list of potential risks, to improve the description of risks.

Brainstorming: When objectives are stated clearly and understood by the participants, a brainstorming session drawing on the creativity of the participants can be used to generate a list of risks. In a well facilitated brainstorming session, the participants are collaborators, comprising a team that works together to articulate the risks that may be known by some in the group. In the session, risks that are known unknowns may emerge, and perhaps even some risks that were previously unknown unknowns may become known.

Facilitating a brainstorming session takes special leadership skills, and, in some organizations, members of the internal audit and ERM staff have been trained and certified to conduct risk brainstorming sessions. In addition to well-trained facilitators, the participants need to understand the ERM framework and how the brainstorming session fits into the ERM process. The participants may very well be required to do some preparation prior to the session.

Casual Mapping:



A causal map is a network diagram representing causes and effects. The diagram contains two basic elements: concepts which are the nodes in the network and causal relationships which are represented by the arcs between the nodes. Concepts are considered as the variables of the system and in some notations carry either positive or negative sign implying the type of the causal relationship and effect. It should be noted that causal maps are often referred to as "cognitive maps", which generally represent people's perceptions of the relationships between cause and effect in a situation.

Assessment:

Having identified the risks that might affect our project we need some way of assessing their importance. Some risks will be relatively unimportant (for example, the risk that some of the documentation is delivered a day late), whereas some will be of major significance (such as the risk that the software is delivered late). Some are quite likely to occur (it is quite likely, for example, that one of the software developers in a team will take a few days sick leave during a lengthy project), whereas others are relatively unlikely (hardware failure causing loss of completed code,

perhaps).
The probability of a hazard's occurring is known as the risk likelihood; the effect that the resulting problem w ill have on the project, if it occurs, is known as the risk impact and the importance of the risk is know n as the risk value or risk exposure. The risk value is calculated as:

risk exposure = potential damange (risk likelihood) x probability of occurrence (risk impact)

Ideally the risk impact is estimated in monetary terms and the likelihood assessed as a probability. In that case the risk exposure will represent an expected cost in the same sense that we calculated expected costs and benefits when discussing cost-benefit analysis. The risk exposures for various risks can then be compared w ith each other to assess the relative importance of each risk and they can be directly compared with the costs and likelihoods of success of various contingency plans.

Many risk managers use a simple scoring method to provide a quantitative measure for assessing each risk. Some just categorize likelihoods and impacts as high, medium or low, but this form of ranking does not allow the calculation of a risk exposure. A better and popular approach is to score the likelihood and impact on a scale of. say. I to 10 where the hazard that is most likely to occur receives a score of 10 and the least likely a score of I.
Ranking likelihoods and impacts on a scale of I to 10 is relatively easy, but most risk managers will attempt to assign scores in a more meaningful way such that, for example, a likelihood scoring 8 is considered twice as likely as one with a score of 4.

Impact measures, scored on a similar scale, must take into account the total risk to the project. This must include the following potential costs:
• the cost of delays to scheduled dates for deliverables;
• cost overruns caused by using additional or more expensive resources;
• the costs incurred or implicit in any compromise to the system's quality or functionality.
Table illustrates part of Amanda's risk value assessment. Notice that the hazard with the highest risk value might not be the one that is most likely nor the one with the greatest potential impact.

Ref	Hazard	Likelihood	Impact	Risk Exposure
R1	Changes to requirements specification during coding	8	8	64
R2	Specification takes longer than expected	3	7	21
R3	Staff sickness affecting critical path activities	5	7	35
R4	Staff sickness affecting non-critical activities	10	3	10
R5	Module coding takes longer than expected	4	5	20
R6	Module testing demonstrates errors or deficiencies in design	4	8	32

Planning:

Risk Acceptance
Risk Avoidance
Risk Reduction and Mitigation
Risk Transfer

Risk Acceptance:

This is the do-nothing option. We could decide that the damage inflicted by some risks would be less that the costs of action that might reduce the probability of a risk happening.

Risk Avoidance:

A risk avoidance methodology attempts to minimize vulnerabilities which can pose a threat. Risk avoidance and mitigation can be achieved through policy and procedure, training and education and technology implementations. Risk avoidance is the elimination of hazards, activities and exposures that can negatively affect an organization's assets.

Risk Reduction and Mitigation:

One of the major risk management techniques. Taking precautionary measures to reduce the likelihood of a loss, or to reduce the severity of a possible loss, for example, installing a security system.Risk mitigation implementation is the process of executing risk mitigation actions. Risk mitigation progress monitoring includes tracking identified risks, identifying new risks, and evaluating risk process effectiveness throughout the projec

Risk Transfer:

Risk transfer is a risk management and control strategy that involves the contractual shifting of a pure risk from one party to another. One example is the purchase of an insurance policy, by which a specified risk of loss is passed from the policyholder to the insurer. When done effectively, risk transfer allocates risk equitably, placing responsibility for risk on designated parties consistent with their ability to control and insure against that risk. Liability should ideally rest with whichever party has the most control over the sources of potential liability**.**

Risk Management:

Contingency
Deciding on the Risk Actions
Creating and Maintaining the Risk Register

Contingency:

The possibility of contingency planning arises when mitigation efforts fail and risk becomes a reality. Contingency planning is used to monitor risks and invoke a predetermined response. According to the plan, a trigger is set up. If the trigger is reached, the contingency plan is put into effect. Contingency planning involves maintaining an alternative plan if the original plan fails. A simple example could be the savings people make for a rainy day. Contingency plans are a must for the top 20 percent of the risks identified. These plans are put to use after the risks become a reality.

Deciding on the Risk Actions:

For each actual risk specific actions have to be planned. Whatever the countermeasures that are considered, they must be cost-effective. On those occasions where a risk exposure value can be calculated as a financial valuse using the (value of damage) * (probability of occurence) formula. Cost effectiveness of a risk reduction action can be assessed by calculating the risk reduction leverage (RRL).

$$RRL= (RE_{before} - RE_{after}) / (\text{cost of risk reduction})$$

$\mathbf{RE_{before}}$ is the risk exposure before risk reduction actions have been taken. $\mathbf{Re_{after}}$ is the risk exposure after risk reduction actions.

Creating and maintaining the Risk Register:

When the project planners have picked out and examined what appeat to be the most threatening risks of the project they need to record their findings in a ***risk register***.

Risk Record					
Risk ID		Risk Title			
Owner		Date Raised		Status	
Risk Description:					
Impact description:					
Recommended risk mitigation:					
Probability or impact values:					

PERT Technique:

The *Program Evaluation and Review Technique* (PERT) is a network model that allows for variations in activity completion times. In a PERT network model, each activity is represented by a line (or *arc*), and each milestone (i.e. the completion of an activity) is represented by a *node*.

The activities are labelled alphabetically, and the expected time required for each activity is also indicated. The critical path is the pathway through the project network that takes the longest to complete, and will determine the overall time required to complete the project. Bear in mind that for a complex project with many activities and task dependencies, there can be more than one critical path through the network, and that the critical path can change. PERT planning involves the following steps:

Identify activities and milestones- the tasks required to complete the project, and the events that mark the beginning and end of each activity, are listed in a table.

Determine the proper sequence of the activities - this step may be combined with step 1, if the order in which activities must be performed is relatively easy to determine.

Construct a network diagram- using the results of steps 1 and 2, a network diagram is drawn which shows activities as arrowed lines, and milestones as circles. Software packages are available that can automatically produce a network diagram from tabular information

Estimate the time required for each activity- any consistent unit of time can be used, although days and weeks are a common.

Determine the critical path- the critical path is determined by adding the activity times for each sequence and determining the longest path in the project. If the activity time for activities in other paths is significantly extended, the critical path may change. The amount of time that a non-critical path activity can be extended without delaying the project is referred to as its slack time.

Update the PERT chart as the project progresses- as the project progresses, estimated times can be replaced with actual times.

The time shown for each project activity when creating the network diagram is the time that the task is expected to take based on a range of possibilities that can be defined as:

The optimistic time (a)- the minimum time required to complete a task

The pessimistic time(b)- the maximum time required to complete a task

The most likely time(m)- an estimate of how long the task will actually take

The expected time (the time that will appear on the network diagram) is defined as the average time the task would require if it were repeated a number of times over a period of time, and can be calculated using the following formula:

expected time = (optimistic time + (4 x most likely time) + pessimistic time) / 6

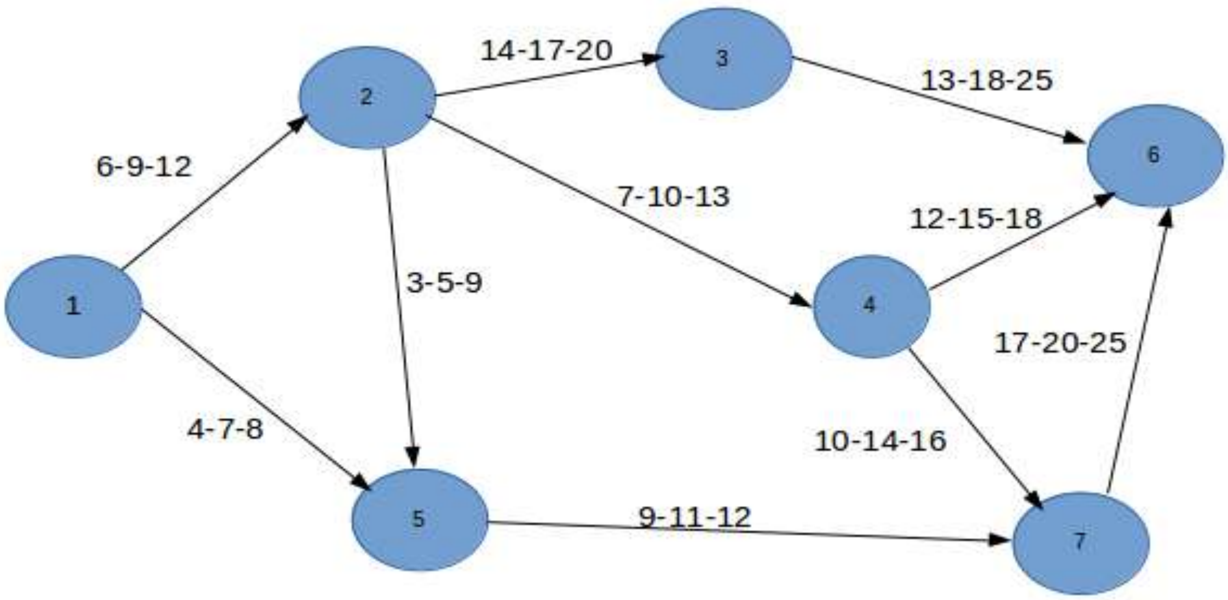
expected time (TE)= (a + (4 * m) + b) / 6

At the same time, the possible variance (V) of the estimate is calculated as below:

V = (b -a)^2 / 6^2

Now, following is the process we follow with the two values:

- For every activity in the critical path, E and V are calculated.
- Then, the total of all Es are taken. This is the overall expected completion time for the project.
- Now, the corresponding V is added to each activity of the critical path. This is the variance for the entire project. This is done only for the activities in the critical path as only the critical path activities can accelerate or delay the project duration.
- Then, standard deviation of the project is calculated. This equals to the square root of the variance (V).
- Now, the normal probability distribution is used for calculating the project completion time with the desired probability.



Example:

Activity	a	m	b	V	te
1-2	6	9	12	1	9
1-5	4	7	8	0.44	6.67
2-3	14	17	20	1	17
2-4	7	10	13	1	10
2-5	3	5	9	1	5.33
3-7	13	18	25	4	18.3
4-6	10	14	16	1	13.66
4-7	12	15	18	1	15
5-6	9	11	12	0.25	10.8
6-7	17	20	25	1.78	20.33

Monte Carlo approach:

Having been named after the principality famous for its casinos, the term Monte Carlo Analysis conjures images of an intricate strategy aimed at maximizing one's earnings in a casino game. However, Monte Carlo Analysis refers to a technique in project management where a manager computes and calculates the total project cost and the project schedule many times.

This is done using a set of input values that have been selected after careful deliberation of probability distributions or potential costs or potential durations.

Importance of the Monte Carlo Analysis

The Monte Carlo Analysis is important in project management as it allows a project manager to calculate a probable total cost of a project as well as to find a range or a potential date of completion for the project.

Since a Monte Carlo Analysis uses quantified data, this allows project managers to better communicate with senior management, especially when the latter is pushing for impractical project completion dates or unrealistic project costs.

Also, this type of an analysis allows the project managers to quantify perils and ambiguities in project schedules.

For Example: Consider the model described above: we are estimating the total time it will take to complete a particular project. In this case, it's a construction project, with three parts. The parts have to be done one after the other, so the total time for the project will be the sum of the three parts. All the times are in months.

In the simplest case, we create a single estimate for each of the three parts of the project. This model gives us a result for the total time: 14 months. But this value is based on three estimates, each of which is an unknown value. It might be a good estimate, but this model can't tell us anything about risk. How likely is it that the project will be completed on time? To create a model we can use in a Monte Carlo simulation, we create three estimates for each part of the project. For each task, we estimate the minimum and maximum expected time (based on our experience, or expertise, or historical information). We use these with the “most likely” estimate, the one that we used above:

Task	Time Estimate			
Job 1	5 Months			
Job 2	4 Months			
Job 3	5 Months			
Total	14 Months			

Task	Minimum	Most Likely	Maximum	
Job 1	4 Months	5 Months	7 Months	
Job 2	3 Months	4 Months	6 Months	
Job 3	4 Months	5 Months	6 Months	
Total	11 Months	14 Months	19 Months	

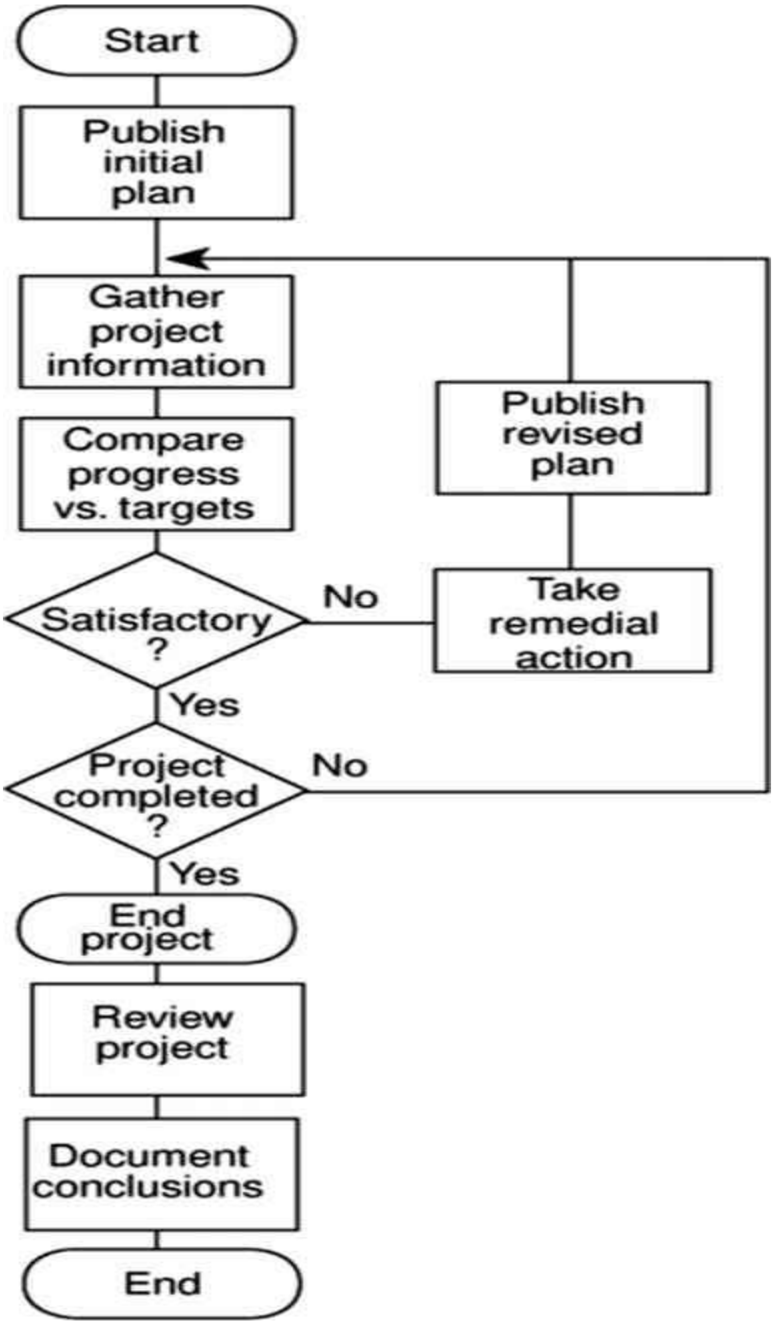
This model contains a bit more information. Now there is a range of possible outcomes. The project might be completed in as little as 11 months, or as long as 19 months. In the Monte Carlo simulation, we will randomly generate values for each of the tasks, then calculate the total time to completion1 . The simulation will be run 500 times. Based on the results of the simulation, we will be able to describe some of the characteristics of the risk in the model. To test the likelihood of a particular result, we count how many times the model returned that result in the simulation. In this case, we want to know how many times the result was less than or equal to a particular number of months.

Time	Number of Times (Out of 500)	Percent of Total (Rounded)	
12 Months	1	0%	
13 Months	31	6%	
14 Months	171	34%	
15 Months	394	79%	
16 Months	482	96%	
17 Months	499	100%	
18 Months	500	100%	

Unit V: Project Monitoring & Control , Resource Allocation

Creating a framework for monitoring & control, Progress monitoring, Cost monitoring, Earned value Analysis, Defects Tracking, Issues Tracking, Status reports, Types of Resources, Identifying resource requirements, Resource

Creating a framework for Monitoring & Control:



--Exercising control over a project and ensuring that targets are met is a matter of regular monitoring, finding out what is happening, and comparing it with current targets.
--If there is a mismatch between the planned outcomes and the actual ones then either replanning is needed to bring the project back on target or the target will have to be revised.
-- The above Figure illustrates a model of the project control cycle and shows how, once the initial project plan has been published, project control is a continual process of monitoring progress against that plan and, where necessary, revising the plan to take account of deviations.
-- It also illustrates the important steps that must be taken after completion of the project so that the experienced gained in any one project can feed into the planning stages of future projects.

Responsibility:

--The overall responsibility for ensuring satisfactory progress on a project is often the role of the project steering committee or Project Board.
--Day-to-day responsibility will rest with theproject manager and, in all but the smallest of projects, aspects of this can be delegated to team leaders.
-- The above Figure illustrates the typical reporting structure found with medium and The concept of a reporting large projects.
--With small projects (employing around half a dozen or fewer staff) individual team members usually report directly to the project manager
Reporting may be oral or written, formal or informal, or regular or ad hoc and some examples of each type are given in Table.

Report Type	Examples	Comment
Oral formal regular	weekly or monthly progress meetings	while reports may be oral formal written minutes should be kept
Oral formal ad hoc	end-of-stage review meetings	while largely oral, likely to receive and generate written reports
Written formal regular	Job sheets, progress reports	normally weekly using forms
Written formal ad hoc	exception reports, change reports	often provides early warning; must be backed up by formal reporting
Oral informal ad hoc	canteen discussion, social interaction	

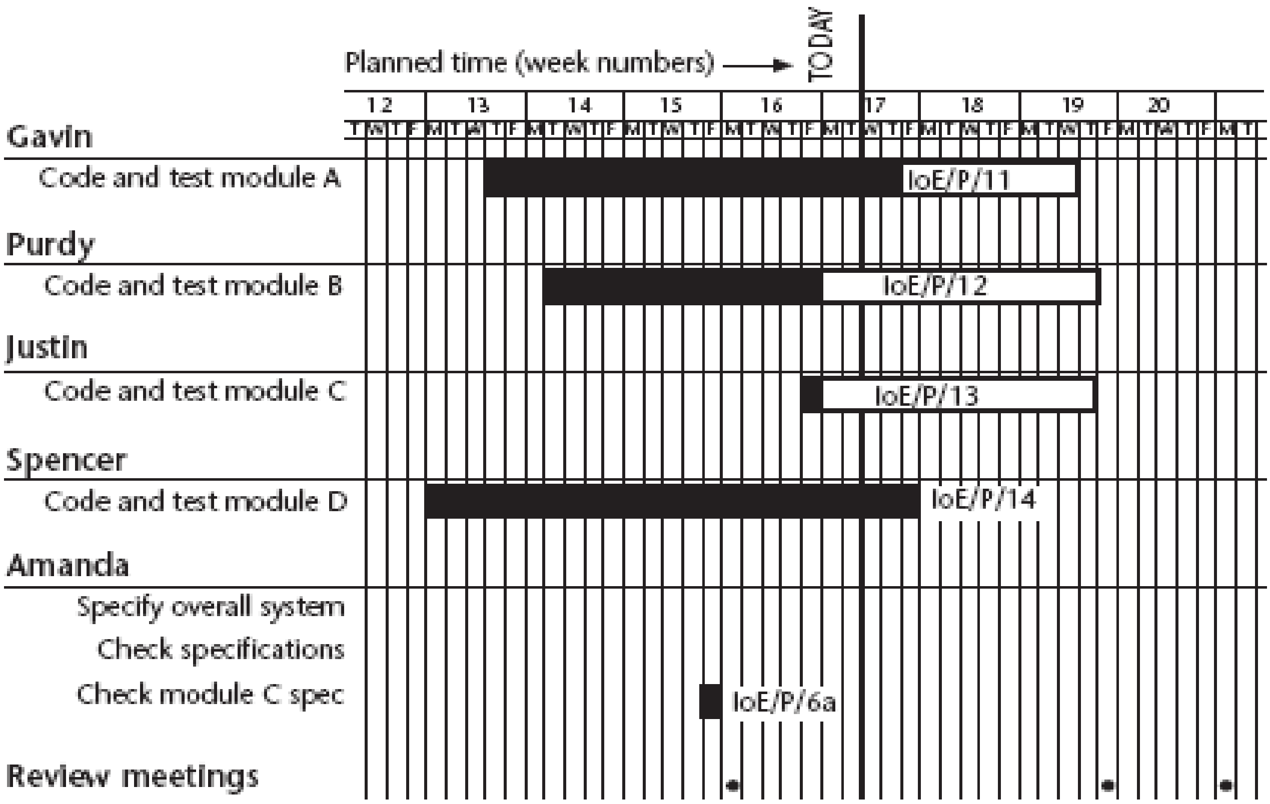
Assessing progress:
Progress assessment will normally be made on the basis of information collected and collated at regular intervals or when specific events occur. Wherever possible, this information will be objective and tangible - whether or not a particular report has been delivered, for example. However, such end-of-activity deliverables might not occur sufficiently frequently throughout the life of the project. Here progress assessment will have to rely on the judgement of the team members who are carrying out the project activities.

Setting checkpoints:
It is essential to set a series of checkpoints in the initial activity plan. Checkpoints may be:
-- Regular (Monthly, for example)
-- Tied to specific events such as the production of a report or other deliverable.

Taking snap-shots:
The frequency with which the a manager needs to receive information about progress will depend upon the size and degree of risk of the project or that part of the project under their control. Team leaders, for example, need to assess progress daily (particularly when employing inexperienced staff) whereas project managers may find weekly or monthly reporting appropriate. In general, the higher the level, the less frequent and less detailed the reporting needs to be.
There are, however, strong arguments in favour of formal weekly collection of information from staff carrying out activities. Collecting data at the end of each week ensures that information is provided while memories are still relatively fresh and provides a mechanism for individuals to review and reflect upon their progress during the past few days.

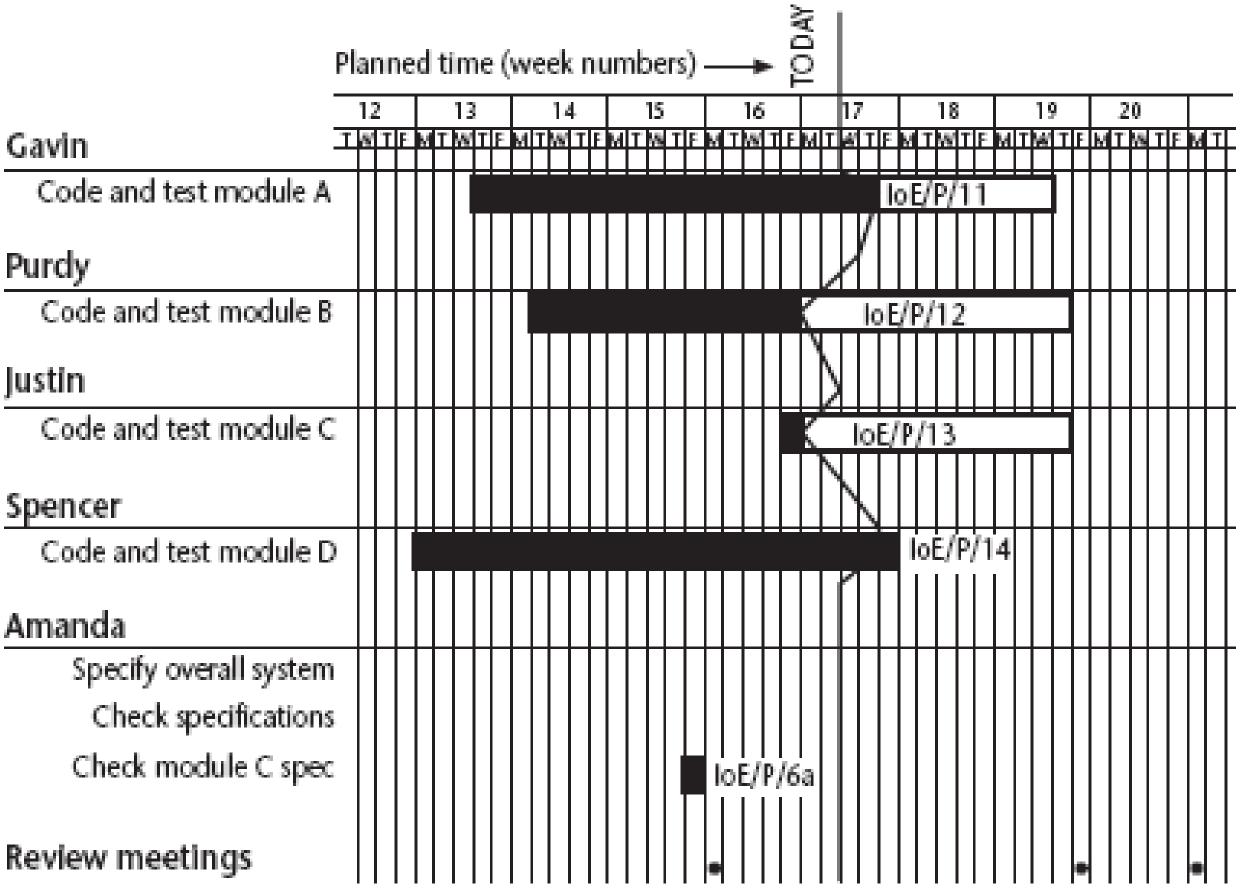
Progress Monitoring:
Having collected data about project progress, a manager needs some way of presenting that data to greatest effect. In this section, we look at some methods of presenting a picture of the project and its future. Some of these methods (such as Gantt charts) provide a static picture, a single snap-shot, whereas others (such as time-line charts) try to show how the project has progressed and changed through time.

The Gantt chart:



--One of the simplest and oldest techniques for tracking project progress is the Gantt chart. This is essentially an activity bar chart indicating scheduled activity dates. Reported progress is recorded on the chart (normally by shading activity bars) and a 'today cursor' provides an immediate visual indication of which activities are ahead or behind schedule. Figure 9.5 shows an example of Gantt chart as at the end of Tuesday of week 17. Code & test module D has been completed ahead of schedule and code & test module A appears also to be ahead of schedule. The coding and testing of the other two modules are behind schedule.

The slip chart:

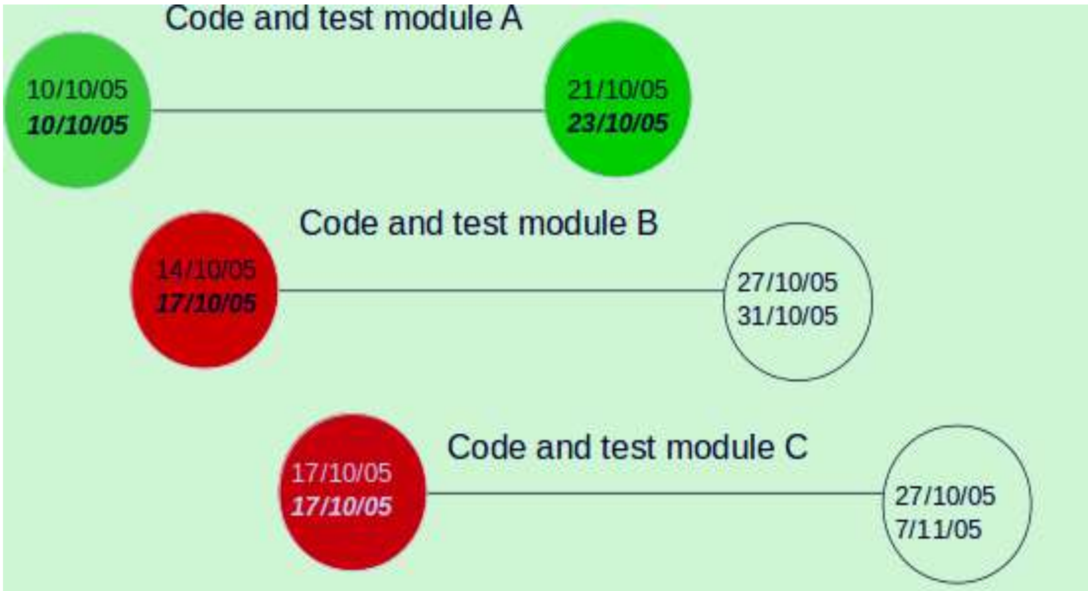


A slip chart is a very similar alternative favoured by some project managerswho believe it provides a more striking visual indication of those activities that are not progressing to schedule - the more the slip line bends, the greater the variation from the plan. Additional slip lines are added at intervals and, as they build up, the project manager will gain an idea as to whether the project is improving (subsequent slip lines bend less) or not. A very jagged slip line indicates a need for rescheduling.

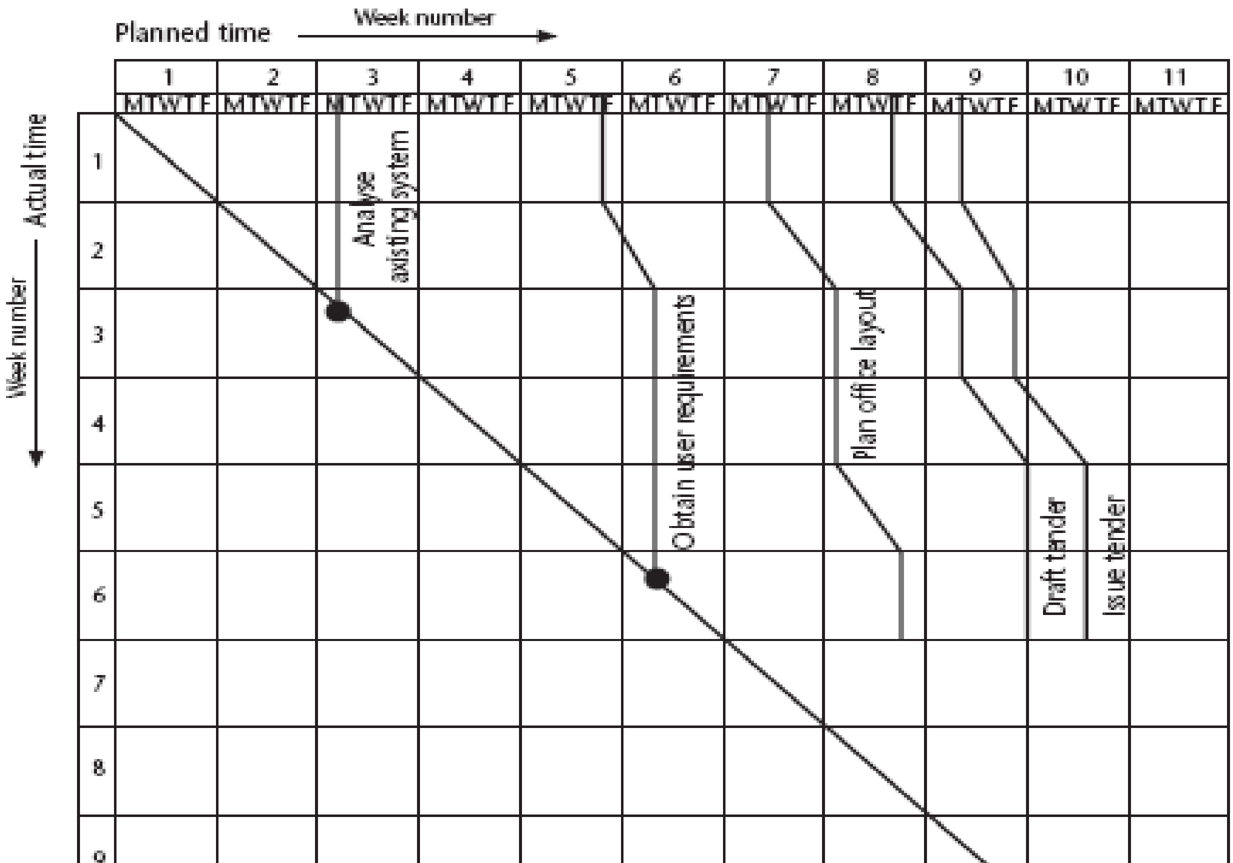
Ball charts

A somewhat more striking way of showing whether or not targets have been met is to use a ball chart as in Figure. In this version of the ball chart, the circles indicate start and completion points for activities. The circles initially contain the original scheduled dates. Whenever revisions are produced these are added as second dates in the appropriate circle until an activity is actually started or completed when the relevant date replaces the revised estimate (**in bold italic**). Circles will therefore contain only two dates, the original and most recent target dates, or the original and actual dates.

Where the actual start or finish date for an activity is later than the target date, the circle is coloured red (dark grey in Figure 9.7) - where an actual date is on time or earlier than the target then the circle is coloured green (light grey in Figure 9.7).
Such charts are frequently placed in a prominent position and the colour coded balls provide a constant reminder to the project team. Where more than one team is working in close proximity, such a highly visible record of achievement can encourage competitiveness between teams.



The timeline:



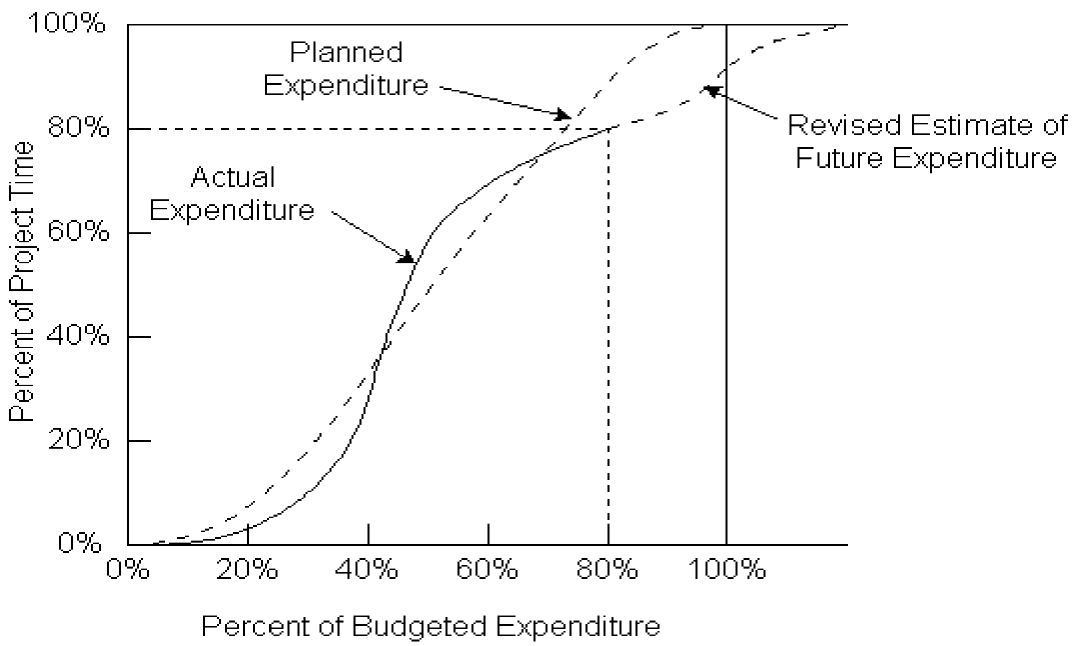
One disadvantage of the charts described so far is that they do not show clearly the slippage of the project completion date through the life of the project. Knowing the current state of a project helps

in revising plans to bring it back on target, but analysing and understanding trends helps to avoid slippage in future projects.
The timeline chart is a method of recording and displaying the way in which targets have changed throughout the duration of the project.
Planned time is plotted along the horizontal axis and elapsed time down the vertical axis. The lines meandering down the chart represent scheduled activity completion dates - at the start of the project analyse existing system is scheduled to be completed by the Tuesday of week 3, obtain user requirements by Thursday of week 5, issue tender, the final activity, by Tuesday of week 9, and so on.
At the end of the first week Project Manager reviews these target dates and leaves them as they are - lines are therefore drawn vertically downwards from the target dates to the end of week one on the actual time axis.

Cost monitoring:

Expenditure monitoring is an important component of project control. Not only in itself, but also because it provides an indication of the effort that has gone into (or at least been charged to) a project. A project might be on time but only because more money has been spent on activities than originally budgeted. A cumulative expenditure chart such as that shown in Figure provides a simple method of comparing actual and planned expenditure. By itself it is not particularly meaningful - for example, illustrate a project that is running late or one that is on time but has shown substantial costs savings! We need to take account of the current status of the project activities before attempting to interpret the meaning of recorded expenditure.

Cost charts become much more useful if we add projected future costs calculated by adding the estimated costs of uncompleted work to the costs already incurred. Where a computer-based planning tool is used, revision of cost schedules is generally provided automatically once actual expenditure has been recorded.



Earned value Analysis:

Earned Value Analysis has gained in popularity in recent years and may be seen as a refinement of the cost monitoring discussed in the previous section. Earned Value Analysis is based on assigning a 'value' to each task or work package (as identified in the WBS) based on the original expenditure forecasts. The assigned value is the original budgeted cost for the item and is known as the baseline budget or budgeted cost of work scheduled (BCWS). A task that has not started is assigned the value zero and when it has been completed, it, and hence the project, is credited with the value of the task. The total value credited to a project at any point is known as the earned value or budgeted cost of work performed (BCWP) and this can be represented as a value or as a percentage of the BCWS.

Where tasks have been started but are not yet complete, some consistent method of assigning an earned value must be applied. Common methods in software projects are:

• T**he 0/100 technique:** Where a task is assigned a value of zero until such time that it is completed when it is given a value of 100% of the budgeted value;

• **The 50/50 technique:** Where a task is assigned a value of 50% of its value as soon as it is started and then given a value of 100% once it is complete;

• T**he milestone technique:** Where a task is given a value based on the achievement of milestones that have been assigned values as part of the original budget plan.

Of these, we prefer the 0/100 technique. The 50/50 technique can give a false sense of security by over-valuing the reporting of activity starts. The milestone technique might be appropriate for activities with a long duration estimate but, in such cases, it is better to break that activity into a number of smaller ones.

The baseline budget:

The first stage in setting up an earned value analysis is to create the baseline budget. The baseline budget is based on the project plan and shows the forecast growth in earned value through time. Earned value may be measured in monetary values but, in the case of staff-intensive projects such as software development, it is common to measure earned value in person-hours or workdays.

Task	Budgeted workdays	Scheduled completion	Cumulative workdays	% cumulative earned
Specify overall system	34	34	34	14.5
Specify module A	20	54	84	35.44
Check specifications	2	56	86	36.28
Design module A	7	63	97	40.93

Monitoring earned value:

Having created the baseline budget, the next task is to monitor earned value as the project progresses. This is done by monitoring the completion of tasks (or activity starts and milestone achievements in the case of the other crediting techniques). As well as recording BCWP, the actual cost of each task can be collected as **actual cost of work performed**, ACWP. This is shown in Figure, which, in this case, records the values as percentages of the total budgeted cost.

Budget variance:

This can be calculated as ACWP - BCWS and indicates the degree to which actual costs differ from those planned.

Schedule variance:

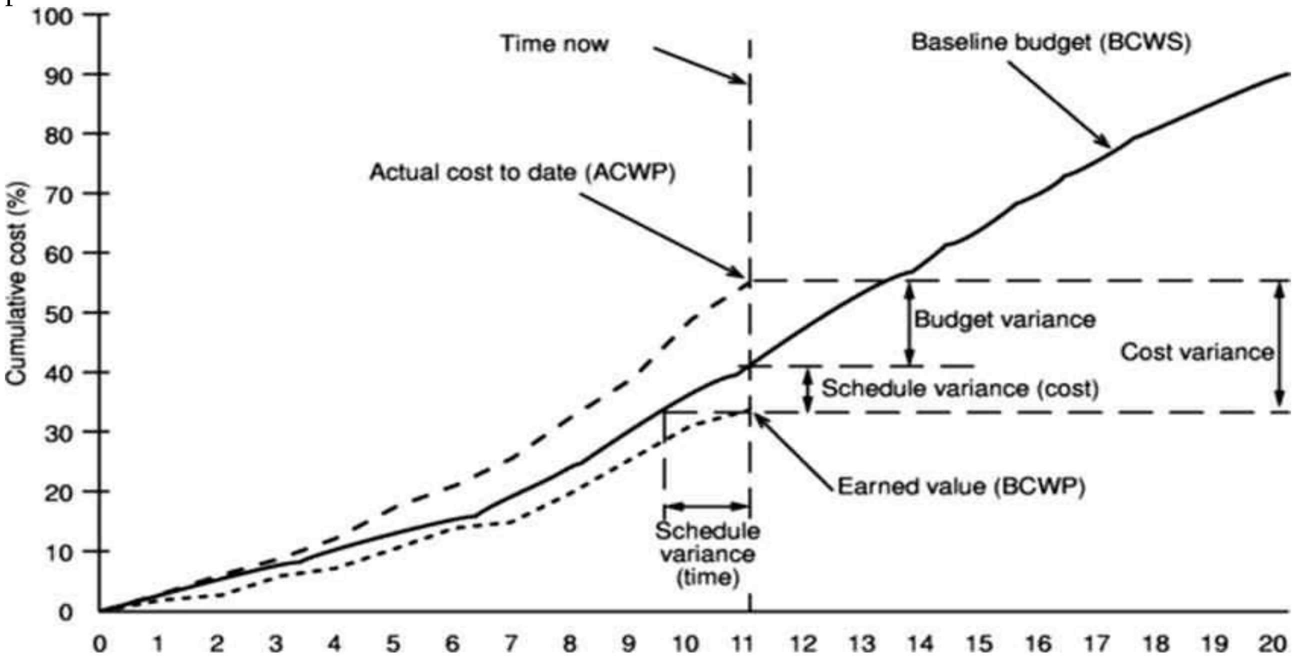
The schedule variance is measured in cost terms as BCWP -BCWS and indicates the degree to which the value of completed work differs from that planned.

Cost variance:

This is calculated as BCWP - ACWP and indicates the difference between the budgeted cost and the actual cost of completed work. It is also an indicator of the accuracy of the original cost estimates.

Performance ratios:

Two ratios are commonly tracked: the cost performance index (CPI = BCWP/ACWP) and the schedule performance index (SPI = BCWP/ BCWS). They can be thought of as a 'value-for-money* indices. A value greater than one indicates that work is being completed better than planned whereas a value of less than one means that work is costing more than and/or proceeding more slowly than planned.



Defects Tracking:

-- Tracking Defects is a key point in project tracking.

-- Once information about a defect is entered in this system, it remains open until it has been fixed.

-- The defect is marked as "Closed" when its removal has been verified. In this way is defect is logged and tracked to closure.

-- Sometimes project managers track the state of a defect by recording the date of submission and the date of closure foreach defect.

-- At the end of the project ideally no open defects should remain.

-- At any time, the project manager can check the overall rate of defect injection and defect closure

Defect Tracking Log

Project Name		Program Manager	

Defect No.	Date Created	Created By	Defect Description
1			
2			
3			
4			
5			
6			
7			
8			
9			

Issues Tracking:

-- Inevitably, many small jobs or clarifications come up during a project. These problems are called as Issues.

-- Managing issues is an important task for any project manager because they potentially delay a project.

-- Because any team member can raise an issue, the list of issues can be very large.

-- Formal methods are therefore useful for tracking them. For this purpose project's usually open an ***issue log***.

--In this **issue log**, issues are recorded as they arise, along with related information.

-- When issues are closed they are marked as closed.

-- For this ***issue log*** project managers can use a spreadsheet, documnet or use the defect tracking system.

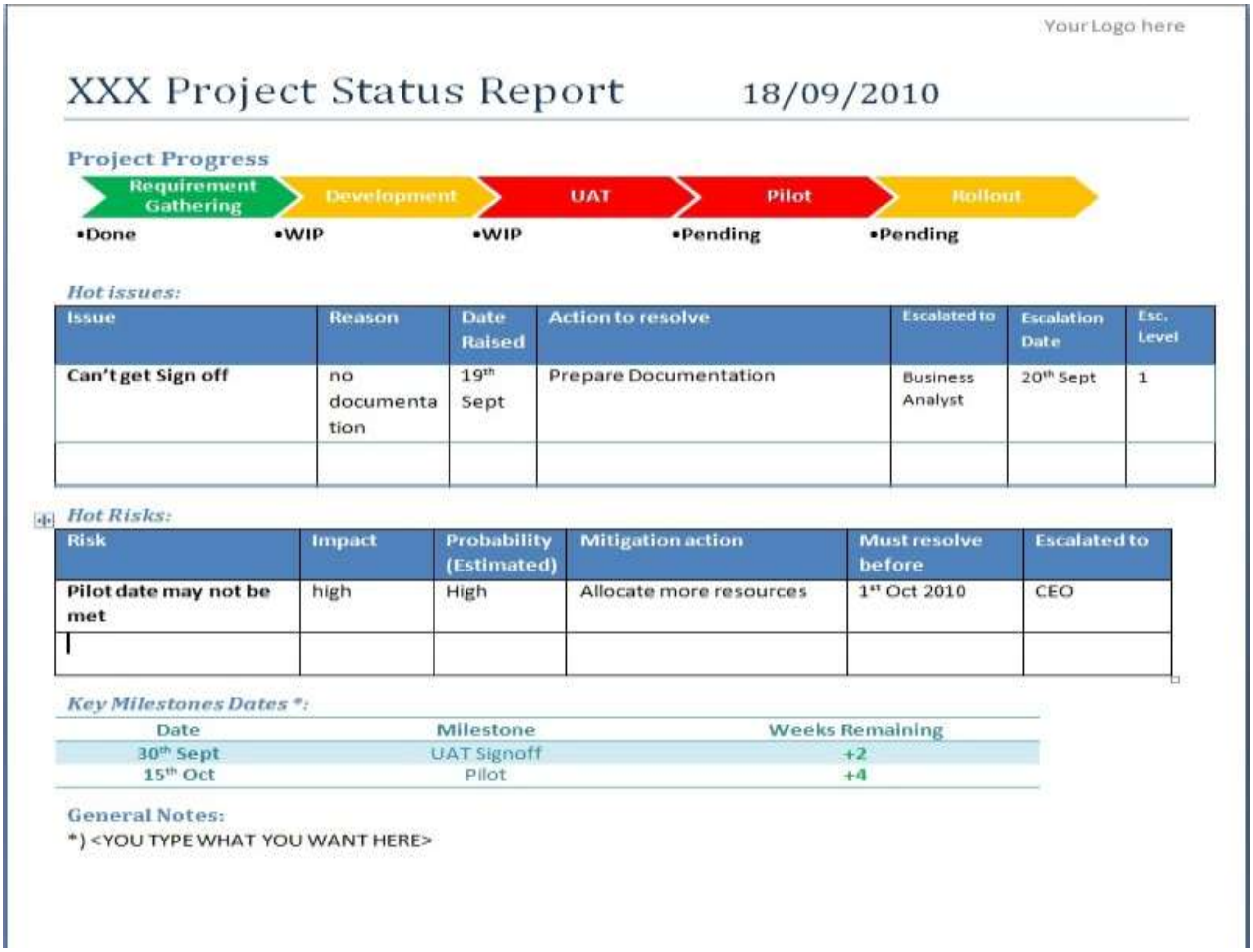
Status reports:

-- Status reports are the main mechanism for regularly communicating the state of the project to senior management and the customer.

-- The parties who recieve the status reports are specified in the project management plan.

-- Typically, status reports are generated weekly and contains these items:

- Customer complaints
- Milestones achieved this week
- Milestones missed this week and reaons for them
- Milestones planned for the next week
- Issuse requiring clarification or attention
- Estimated work versus available time by milestoness.
- Escalation



Your Logo here

XXX Project Status Report 18/09/2010

Project Progress

•Done •WIP •WIP •Pending •Pending

Hot issues:

Issue	Reason	Date Raised	Action to resolve	Escalated to	Escalation Date	Esc. Level
Can't get Sign off	no documentation	19th Sept	Prepare Documentation	Business Analyst	20th Sept	1

Hot Risks:

Risk	Impact	Probability (Estimated)	Mitigation action	Must resolve before	Escalated to
Pilot date may not be met	high	High	Allocate more resources	1st Oct 2010	CEO

Key Milestones Dates *:

Date	Milestone	Weeks Remaining
30th Sept	UAT Signoff	+2
15th Oct	Pilot	+4

General Notes:

*) <YOU TYPE WHAT YOU WANT HERE>

Types of Resources:

A resource is any item or person required for the execution of the project. This covers many things - from paper clips to key personnel - and it is unlikely that we would wish to itemize every resource required, let alone draw up a schedule for their use! Stationery and other standard office supplies, for example, need not normally be the concern of the project manager - ensuring there is always an adequate supply is the role of the office manager. The project manager must concentrate on those resources where there is a possibility that, without planning, they might not be sufficiently available when required.

Some resources, such as a project manager, will be required for the duration of the project whereas others, such as a specific software developer, might be required for a single activity. The former, while vital to the success of the project, does not require the same level of scheduling as the latter. Individual programmers, for example, might be committed to working on a number of projects and it will be important to book their time well in advance.

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

Labour: The main items in this category will be members of the development project team such as the project manager, systems analysts and software developers. liqually important will be the quality assurance team and other support staff and any employees of the client organization who might be required to undertake or participate in specific activities.

Equipment: Obvious items will include workstations and other computing and office equipment. We must not forget that staff also need basic equipment such as desks and chairs.

Materials: Materials are items that are consumed, rather than equipment that is used. They are of little consequence in most software projects but can be important for some - software that is to be widely distributed might, for example, require supplies of floppy disks to be specially obtained.

Space: For projects that are undertaken w ith existing staff, space is normally readily available. If any additional staff (recruited or contracted) should be needed then office space will need to be found.

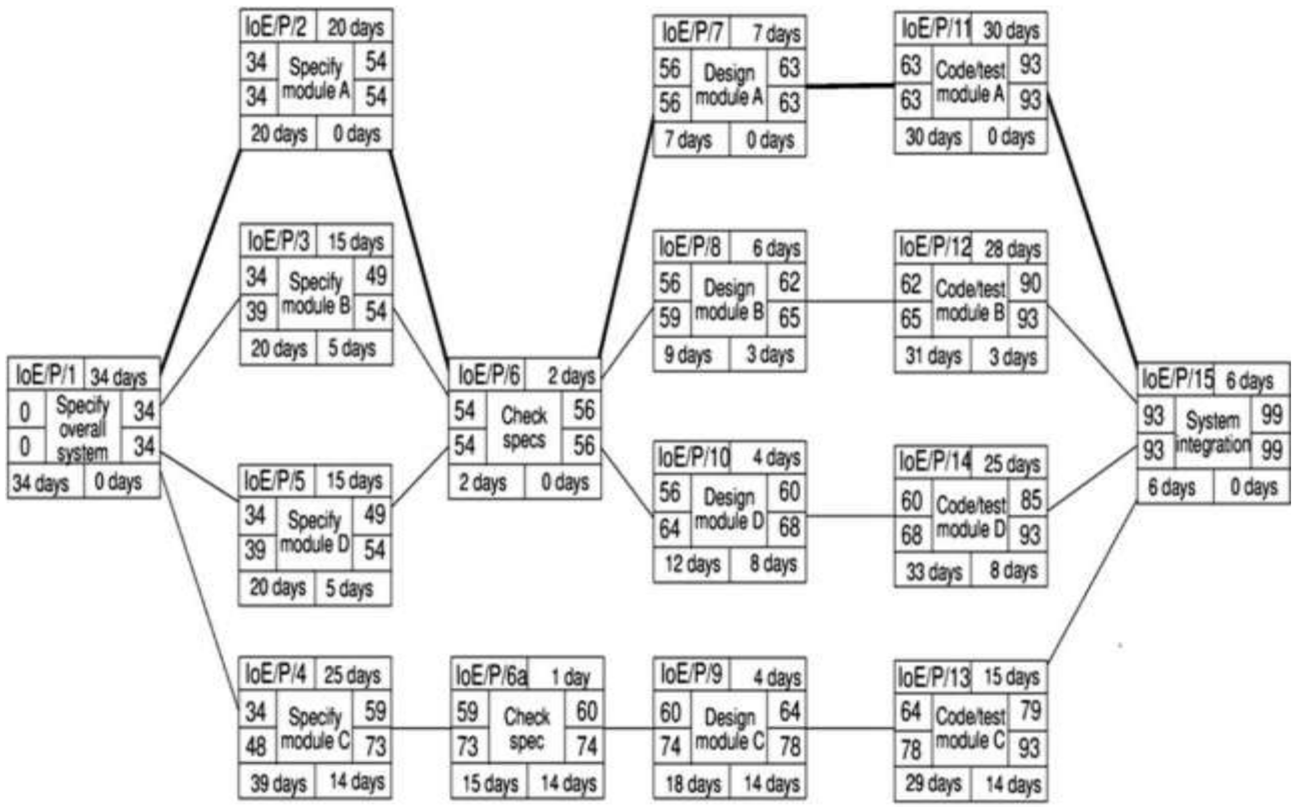
Services: Some projects will require procurement of specialist services -development of a wide area distributed system, for example, requires scheduling of telecommunications services.

Time is the resource that is being offset against the other primary The cost of money as a resources - project time-scales can sometimes be reduced by increasing other resource is a factor taken resources and will almost certainly be extended if they arc unexpectedly into account in DCF reduced. evaluation.

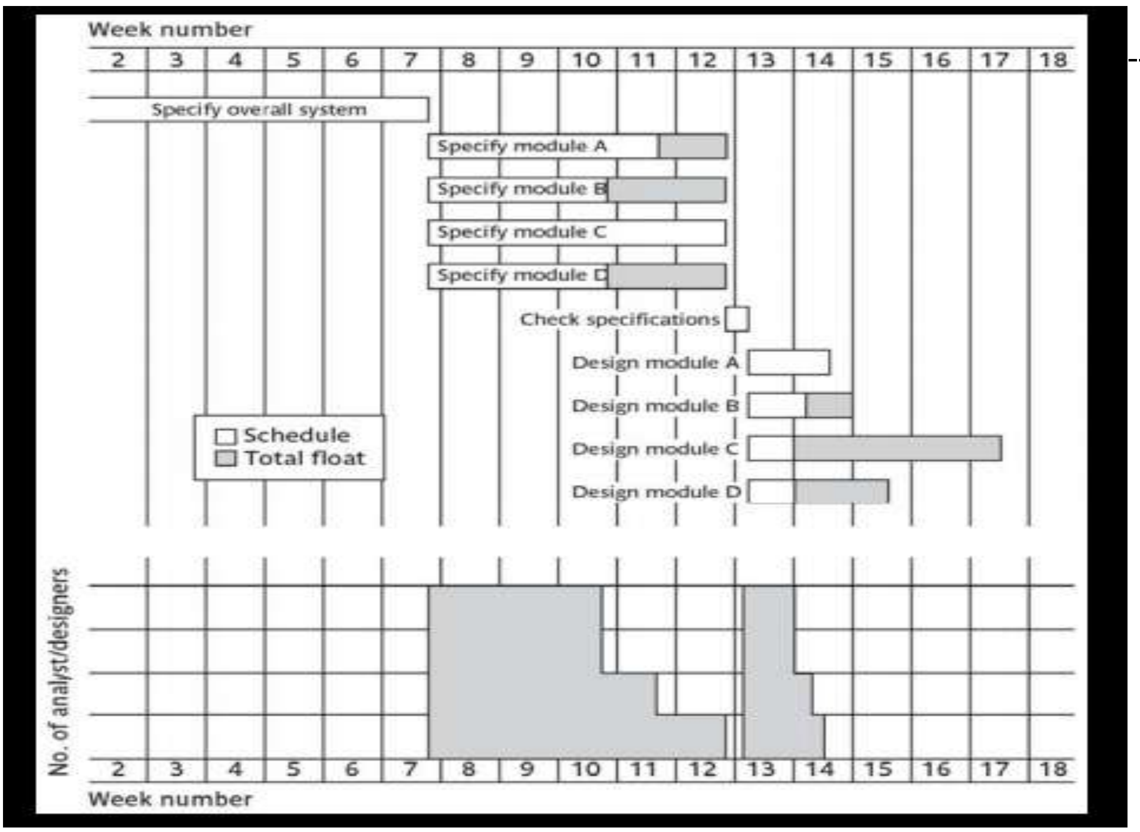
Money: Money is a secondary resource - it is used to buy other resources and will be consumed as other resources are used. It is similar to other resources in that it is available at a cost - in this case interest charges.

Identifying resource requirements:

The first step in producing a resource allocation plan is to list the resources that will be required along with the expected level of demand. This will normally be done by considering each activity in turn and identifying the resources required. It is likely, however, that there will also be resources required that are not activity specific but are part of the project's infrastructure (such as the project manager) or required to support other resources (office space, for example, might be required to house contract software developers).



Resource Scheduling:



Having produced the resource requirements list, the next stage is to map this onto the activity plan to assess the distribution of resources required over the duration of the project.
-- This is best done by representing the activity plan as a bar chart and using this to produce a

resource histogram for each resource.
-- Changing the level of resources on a project over time, particularly personnel, generally adds to the cost of a project. Recruiting staff has costs and even where staff are transferred internally, time will be needed for familiarization w ith the new project environment.
-- The resource histogram in Figure poses **particular problems** in that it calls for two analyst-designers to be idle for eleven days, one for six days and one for two days between the specification and design stage.
-- White rectangles indicate when an activity is scheduled and shaded rectangles the total floatt.

Resource Smoothing:

“A technique that adjusts the activities of a schedule model such that the requirements for resources on the project do not exceed certain predefined resource limits.”

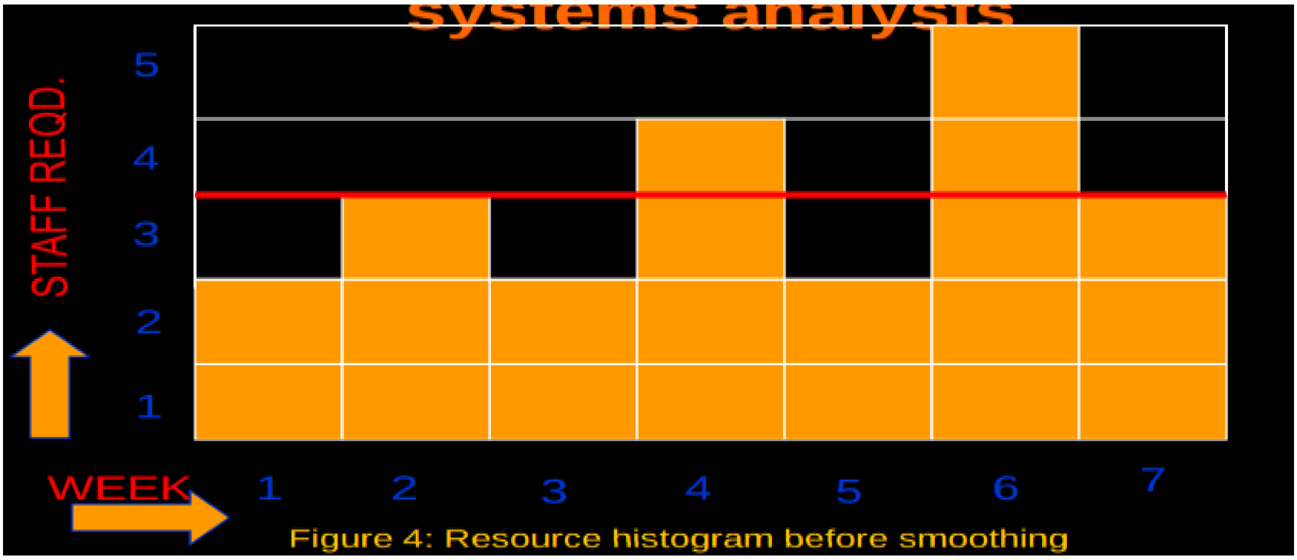
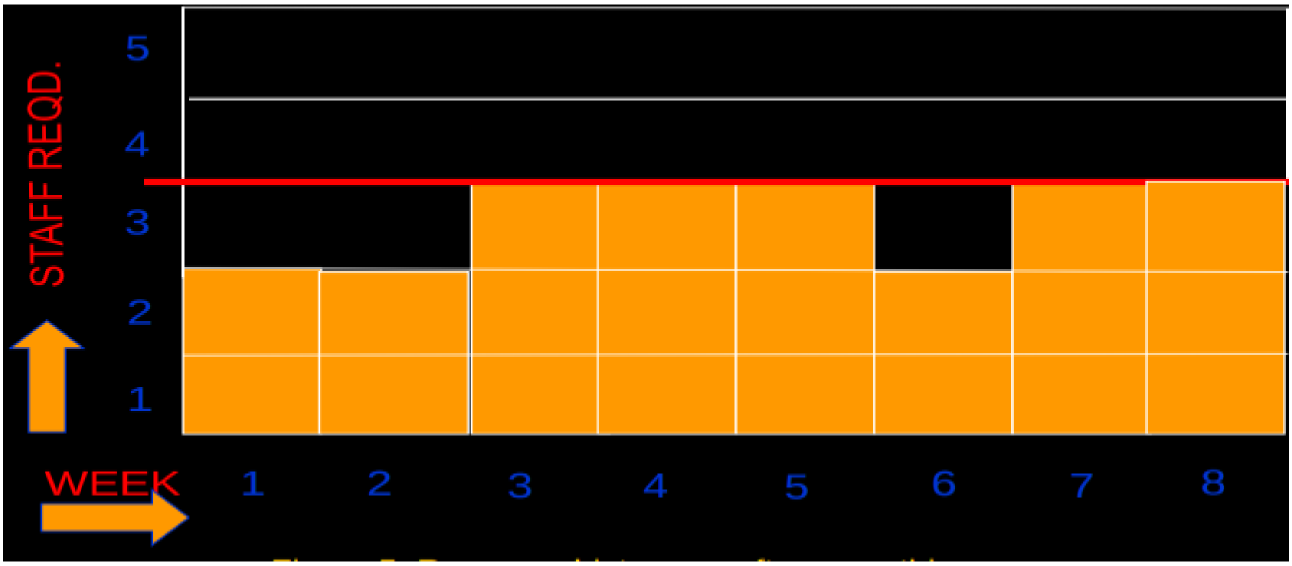


Figure 4: Resource histogram before smoothing



In practice, resources have to be allocated to a project on an activity-by-activity basis and finding the 'best' allocation can be time consuming and difficult. As soon as a member of the project team is allocated to an activity that activity acquires a scheduled start and finish date and the team member becomes unavailable for other activities for that period. Thus, allocating a resource to one activity limits the flexibility for resource allocation and scheduling of other activities.

It is therefore helpful to prioritize activities so that resources can be allocated to competing activities in some rational order. The priority must always be to allocate resources to critical path activities and then to those activities that are most likely to affect others. In that way, lower priority activities are made to fit around the more critical, already scheduled activities. Of the various ways of prioritizing activities, two are described below.

--**Total float priority:** Activities are ordered according to their total float, those with the smallest total float having the highest priority. In the simplest application of this method, activities are allocated resources in ascending order of total float. However, as scheduling proceeds, activities will be delayed (if resources are not available at their earliest start dates) and total floats will be reduced. It is therefore desirable to recalculate floats (and hence reorder the list) each time an activity is delayed.

-- **Ordered list priority:** With this method, activities that can proceed at the same time are ordered according to a set of simple criteria. An example of this is Burman's priority list, which takes into account activity duration as well as total float:

1. shortest critical activity;
2. critical activities;
3. shortest non-critical activity;
4. non-critical activity with least float;
5. non-critical activities.