have the information you need to plan the critical path schedule more accurately and have more of a guarantee you'll meet your project deadline.

You also need to consider other changes or constraints that might change the project schedule. The more you can account for these unexpected events or risks, the more accurate your critical path schedule will be. If time is added to the project because of these constraints, that is called a critical path drag, which is how much longer a project will take because of the task and constraint.

Counting the Cost:

For any new software project, it is necessary to know how much it will cost to develop and how much development time will it take. These estimates are needed before development is initiated, but how is this done? Several estimation procedures have been developed and are having the following attributes in common.

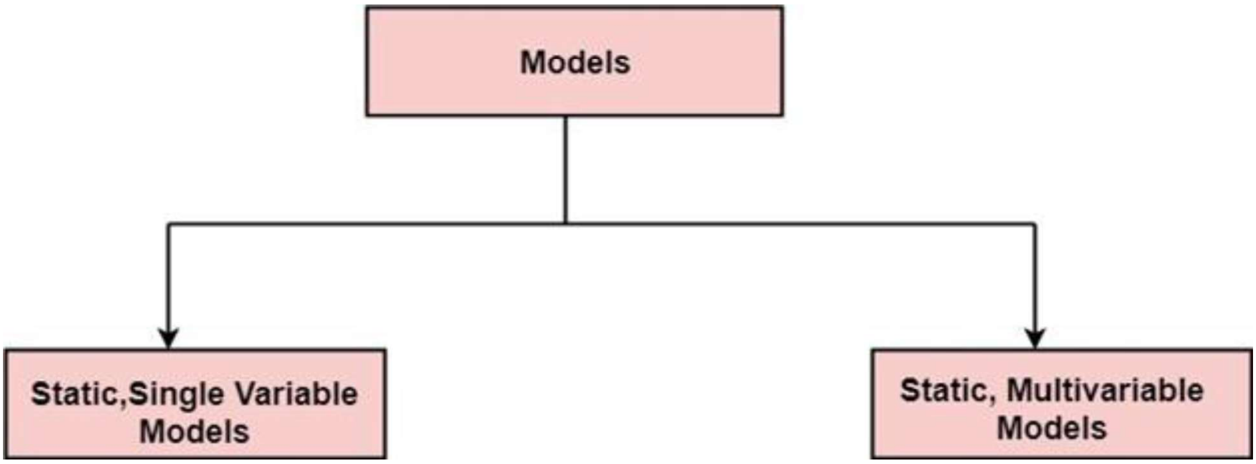
1. Project scope must be established in advanced.
2. Software metrics are used as a support from which evaluation is made.
3. The project is broken into small PCs which are estimated individually.
   To achieve true cost & schedule estimate, several option arise.
4. Delay estimation
5. Used symbol decomposition techniques to generate project cost and schedule estimates.
6. Acquire one or more automated estimation tools.

Uses of Cost Estimation

1. During the planning stage, one needs to choose how many engineers are required for the project and to develop a schedule.
2. In monitoring the project's progress, one needs to access whether the project is progressing according to the procedure and takes corrective action, if necessary.

Cost Estimation Models

A model may be static or dynamic. In a static model, a single variable is taken as a key element for calculating cost and time. In a dynamic model, all variable are interdependent, and there is no basic variable.



Static, Single Variable Models: When a model makes use of single variables to calculate desired values such as cost, time, efforts, etc. is said to be a single variable model. The most common equation is:

$$\mathbf{C=aL^b}$$

Where C = Costs
L= size
a and b are constants

The Software Engineering Laboratory established a model called SEL model, for estimating its software production. This model is an example of the static, single variable model.

$E=1.4L^{0.93}$
$DOC=30.4L^{0.90}$
$D=4.6L^{0.26}$

Where E= Efforts (Person Per Month)
DOC=Documentation (Number of Pages)
D = Duration (D, in months)
L = Number of Lines per code

Static, Multivariable Models: These models are based on method (1), they depend on several variables describing various aspects of the software development environment. In some model, several variables are needed to describe the software development process, and selected equation combined these variables to give the estimate of time & cost. These models are called multivariable models.

WALSTON and FELIX develop the models at IBM provide the following equation gives a relationship between lines of source code and effort:

$$\mathbf{E=5.2L^{0.91}}$$

In the same manner duration of development is given by

$$\mathbf{D=4.1L^{0.36}}$$

The productivity index uses 29 variables which are found to be highly correlated productivity as follows:

$$\mathbf{I = \sum_{i=1}^{29} W_i\, X_i}$$

Where $\mathbf{W_i}$ is the weight factor for the $\mathbf{i^{th}}$variable and $\mathbf{X_i=\{-1,0,+1\}}$ the estimator gives $\mathbf{X_i}$one of the values **-1, 0 or +1** depending on the variable decreases, has no effect or increases the productivity.

Example: Compare the Walston-Felix Model with the SEL model on a software development expected to involve 8 person-years of effort.

a. Calculate the number of lines of source code that can be produced.
b. Calculate the duration of the development.
c. Calculate the productivity in LOC/PY
d. Calculate the average manning

Solution:

The amount of manpower involved = 8PY=96persons-months

(a)Number of lines of source code can be obtained by reversing equation to give:

$$L = \left(\frac{E}{a}\right) 1/b$$

Then

L (SEL) = (96/1.4)1⁄0.93=94264 LOC
L (SEL) = (96/5.2)1⁄0.91=24632 LOC

(b)Duration in months can be calculated by means of equation

D (SEL) = 4.6 (L) 0.26
= 4.6 (94.264)0.26 = 15 months
D (W-F) = 4.1 $L^{0.36}$
= 4.1 (24.632)0.36 = 13 months

(c) Productivity is the lines of code produced per persons/month (year)

$$\text{P (SEL)} = \frac{94264}{8} = 11783\ \frac{\text{LOC}}{\text{Person}} - \text{Years}$$

$$\text{P (Years)} = \frac{24632}{8} = 3079\ \frac{\text{LOC}}{\text{Person}} - \text{Years}$$

(d)Average manning is the average number of persons required per month in the project

$$M\ (SEL) = \frac{96P-M}{15M} = 6.4\text{Persons}$$

$$M\ (W\text{-}F) = \frac{96P-M}{13M} = 7.4\text{Persons}$$

Being Specific:

Allocating resources and smoothing resource histograms is relatively straightforward where all resources of a given type can be considered more or less equivalent. When allocating labourers to activities in a large building project we need not distinguish among individuals - there are likely to be many labourers and they may be treated as equals so far as skills and productivity are concerned.

This is seldom the case with software projects. We saw in Chapter 5 that, because of the nature of software development, skill and experience play a significant part in determining the time taken and, potentially, the quality of the final product. With the exception of extremely large projects it makes sense to allocate individual members of staff to activities as early as possible, as this can lead us to revise our estimate of their duration.

In allocating individuals to tasks, a number of factors need to be taken into account.

• Availability We need to know whether a particular individual will be available when required. Reference to the departmental work plan determines this but the wise project manager w ill always investigate the risks that might be involved - earlier projects might, for example, over-run and affect the availability of an individual.

• Criticality Allocation of more experienced personnel to activities on the critical path often helps in shortening project durations or at least reduces the risk of overrun.

• Risk We saw how to undertake activity risk assessment in the previous chapter. Identifying those activities posing the greatest risk, and knowing the factors influencing them, helps to allocate staff. Allocating the most experienced staff to the highest risk activities is likely to have the greatest effect in reducing overall project uncertainties. More experienced staff are, however, usually more expensive.

• Training It will benefit the organization if positive steps are taken to allocate junior staff to appropriate non-critical activities where there will be sufficient

Reappraisal of the critical path and PERT or Monte Carlo risk analysis might need to be carried out in parallel with staff allocation.

slack for them to train and develop skills. There can even be direct benefits to the particular project since some costs may be allocated to the training budget.

• Team Building The selection of individuals must also take account of the final shape of the project team and the way they will work together. This and additional aspects of personnel management are discussed in Chapter 11.

Exercise 8.4 Amanda has decided that, where possible, whoever writes the specification for a module should also produce the design, as she believes this will improve the commitment and motivation of the three analyst-designers. Belinda. Tom and Daisy.

She has decided that she will use Tom, a trainee analyst-designer, for the specification and design of module D as both of these activ ities have a large float compared to their activity span (%t and V%y of their span respectively). Since the specification and design of module C are on the critical path, she decides to allocate both of these tasks to Belinda, a particularly experienced and capable member of staff.

Having made these decisions she has almost no flexibility in how she assigns the other specification and design activities. Work out from the activity bar chart produced as part of the solution to Exercise 8.2 (shown in Figure 8.6) whom she assigns to w hich of the remaining specification and design activities.

Publishing the Resource Schedule:

In allocating and scheduling resources we have used the activity plan (a precedence network in the case of the examples in this chapter), activity bar charts and resource histograms. Although good as planning tools, they are not the best way of publishing and communicating project schedules, For this we need some form of work plan. Work plans are commonly published either as lists or charts such as that illustrated in Figure 8.7. In this case Amanda has chosen not to include activity floats (which could be indicated by shaded bars) as she fears that one or two members of the team might work w ith less urgency if they are aware that their activities are not critical.

Notice that, somewhat unusually, it is assumed there are no public holidays or other non-productive periods during the 100 days of the project and that none of the team has holidays for the periods they are shown as working.

Amanda has also made no explicit allowance for staff taking sick leave.

Amanda now transfers some of the information from the work schedule to her precedence network. In particular, she amends the earliest start dates for activities and any other constraints (such as revised latest finish dates where resources need to be made available) that have been introduced. A copy of her revised precedence network is shown in Figure 8.8 - notice that she has highlighted all critical activities and paths.

Belinda

Daisy

Gavin

Purdy

Specify module B Speafy module C Design module C System integration

Specify module D Design module D Design module B

Specify module A Design module A

Code S test module A Code & test module B

Justin

Code 4 test module C

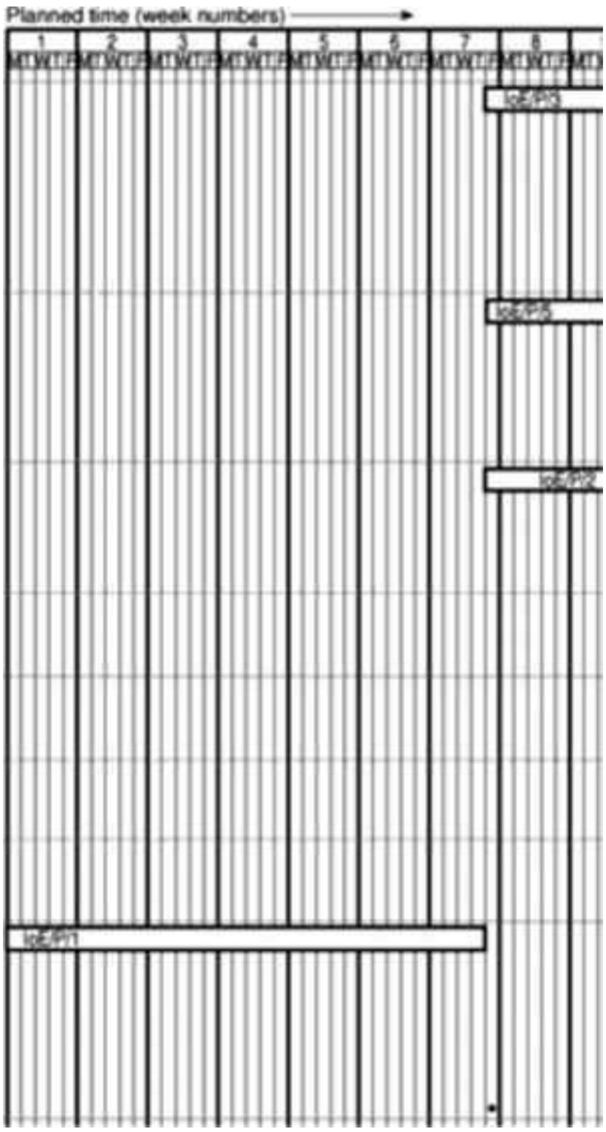
Spencer

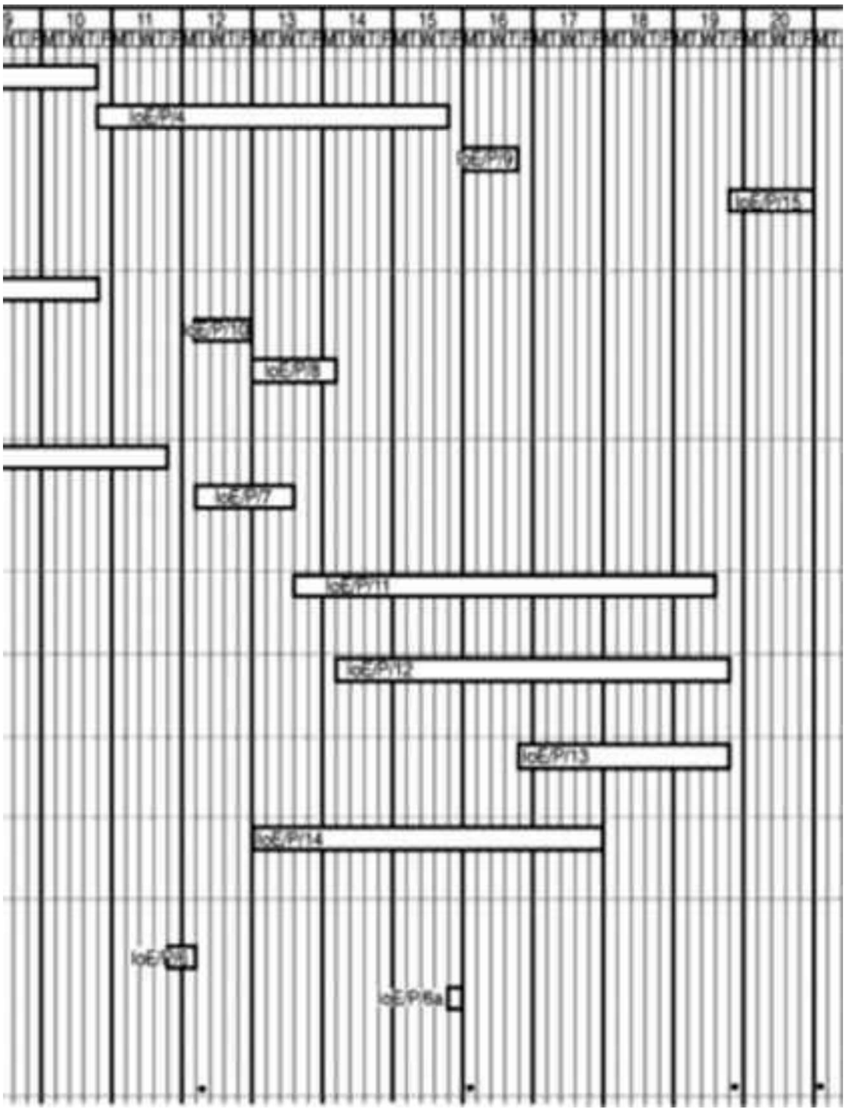
Code & test module D

Amanda

Specify overall sysiem Check specifications Check module C spec

Review meetings


Planned time (week numbers)
1
2
3
4
5
6
7
8



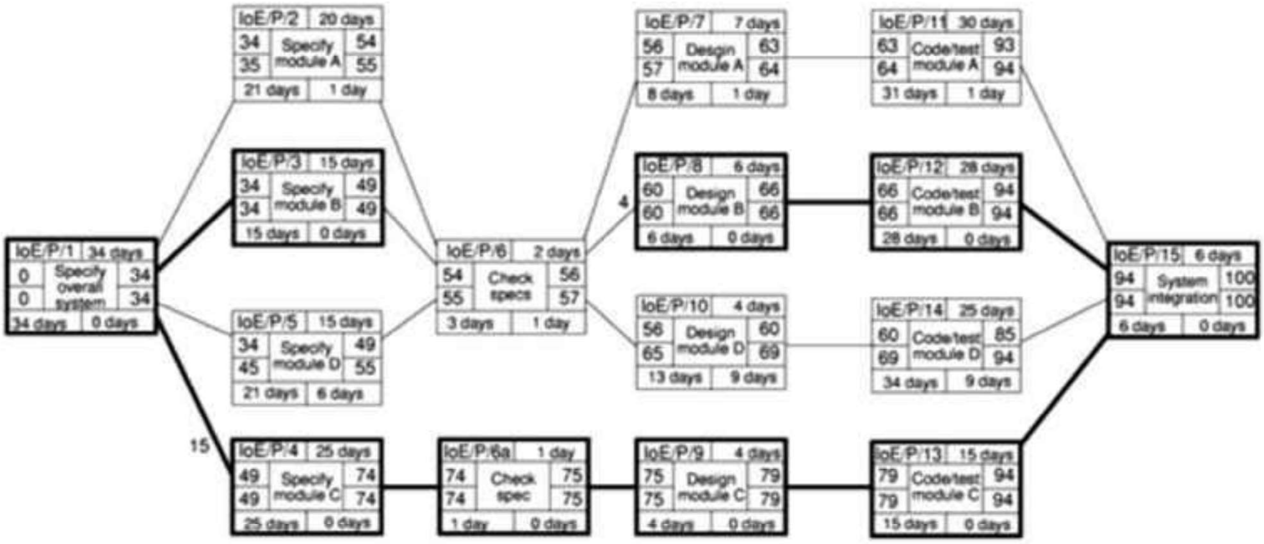


Figure 8.8 Amanda \v revised precedence network showing scheduled start and completion dates.

Cost Schedule:

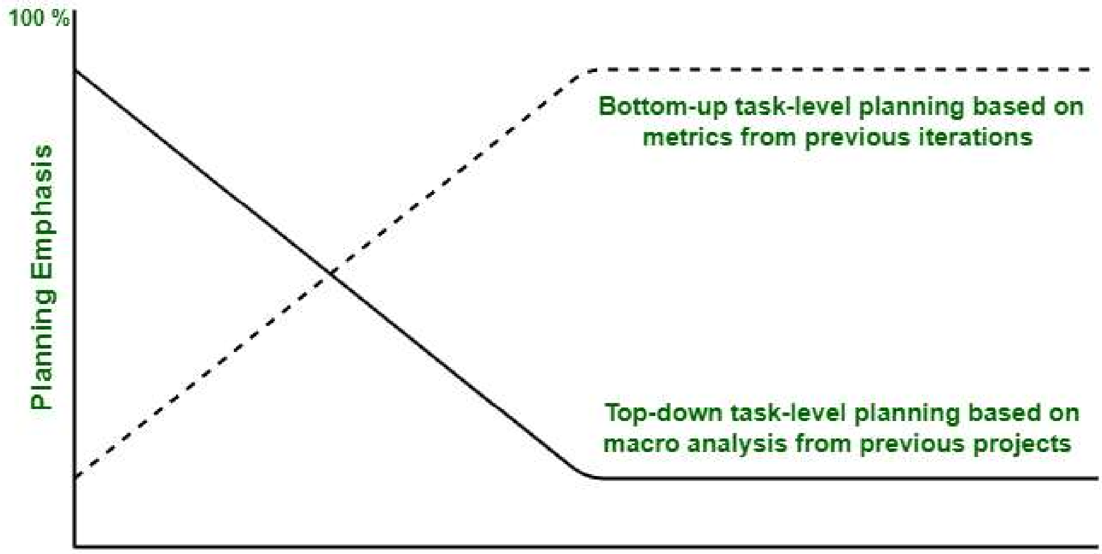
The cost and schedule estimation process helps in determining number of resources to complete all project activities. It generally involves approximation and development of costing alternatives to plan, perform or work, deliver, or give project. A good estimation is very much essential for keeping a project under budget.

Two perspectives are generally required to derive project plans. These perspectives are given below :

1. **Forward-Looking :**
    - The Forward-Looking approach is also known as Top-Down approach. This approach generally starts with describing and explaining various project tasks that involve starting with project aim or end deliverable and breaking it all down into smaller planning chunks.
    - Top-down budgeting also refers to a method of budgeting where project managers prepare a high-level budget for organization.
    - These project managers or senior management develops and creates a characterization of overall size, process, environment, people, and quality that is essential for software project. In this approach, duration of deliverable's is estimated.
    - It generally takes less time and effort than bottom-up estimate. With help of software cost estimation model, an estimation of overall effort and schedule is done. The project manager generally divides estimation of overall effort into a top-level of WBS (Work Breakdown Structure).
    - They also divide schedule into major milestones dates. At this stage, sub-project managers are simply given responsibility for decomposing every element of WBS into lower levels with help of various allocations of top-level, staffing profile, and, major milestones dates as constraints.
    - The main benefit of this approach is use of holistic data from earlier projects or products, along with unmitigated risks, and scope creeps. This also helps in reducing risk of overlooked work activities or costs.
2. **Backward-Looking :**
    - Backward-Looking approach is also known as Bottom-up approach.
    - In this approach, project team breaks requirements of clients down, determining lowest level appropriate to develop a range of estimates, covering overall scope of project based on available definition of task.
    - Overall elements of lowest level WBS are generally explained into detailed tasks, for which WBS element manager is responsible for estimating budget and schedule.
    - All of these estimates are joined and integrated into higher-level WBS budgets and milestones.

Milestone scheduling also called budget allocation with help of top-down approach results in a highly optimistic plan. Whereas, bottom-up approach results in a highly pessimistic plan. Iteration is very much needed and important, using results of one approach to validate and even check results of other approach. Both of approaches should be used together, in balance, throughout life-cycle of project as shown below.

Below is diagram showing planning balance through life cycle.



Engineering Stage		Production Stage	
Inception	Elaboration	Construction	Transition

Engineering stage planning emphasis on following points :

- Macro-level task estimation for engineering artifacts.
- Macro-level task estimation for production stage artifacts.
- Stakeholder concurrence.
- Coarse-grained variance analysis of actual vs. planned expenditures.
- Tuning top-down project-independent planning guidelines into project-specific planning guidelines.
- WBS definition and elaboration.

Production stage planning emphasis on following points :

- Macro-level task estimation for production stage artifacts.
- Macro-level task estimation for maintenance of engineering artifacts.
- Stakeholder concurrence.
- Coarse-grained variance analysis of actual vs. planned expenditures.

Top-down perspective generally dominates during engineering stage. This is because there is no enough depth or details of understanding not even stability in sequences of detailed task to perform planning of bottom-up approach. On other hand, there is enough prior experience and planning fidelity that bottom-up planning perspective dominates during production stage.

The Scheduling Sequence:

(ioing from an ideal activity plan to a costcd schedule can be represented as a sequence of steps, rather like the classic waterfall life-cycle model. In the ideal world, we would start with the activity plan and use this as the basis for our risk assessment. The activity plan and risk assessment would provide the basis for our resource allocation and schedule from which we would produce cost schedules.

In practice, as we have seen by looking at Amanda's project, successful resource allocation often necessitates revisions to the activity plan, which, in turn, will affect our risk assessment. Similarly, the cosi schedule might indicate the need or

7000 -r

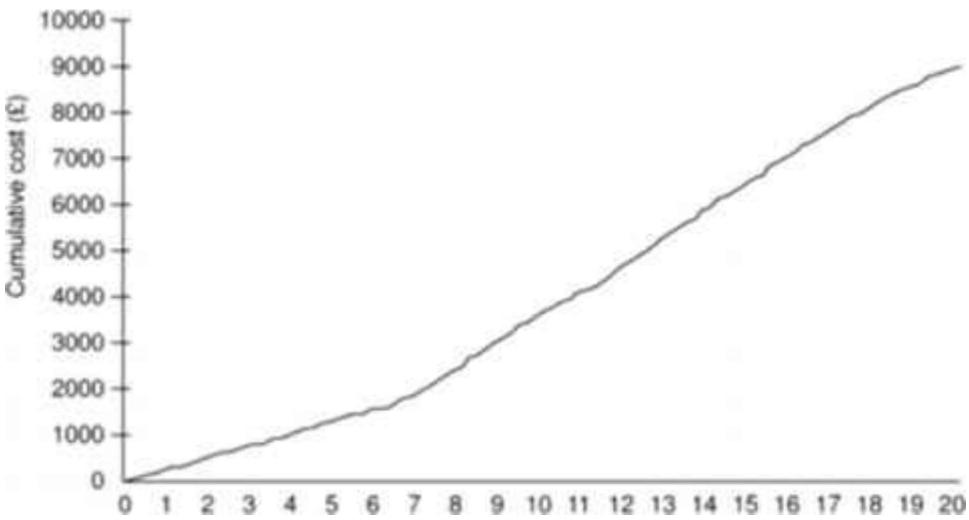
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Week number

Figure 8.9 Weekly project costs for the IOE project.

2 3000-fl

Overheads



Week number

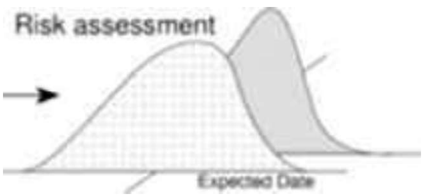
Figure 8.10 Cumulative project costs for the IOE project.

Week number

Figure 8.10 Cumulative project costs for the IOE project.

desirability to reallocate resources or revise activity plans - particularly where that schedule indicates a higher overall project cost than originally anticipated.

The interplay between the plans and schedules is complex - any change to any one will affect each of the others. Some factors can be directly compared in terms of money - the cost of hiring additional staff can be balanced against the costs of delaying the project's end date. Some factors, however, are difficult to express in money terms (the cost of an increased risk, for example) and will include an element of subjectivity.



Cod* Wodu*

Oocu^wtC

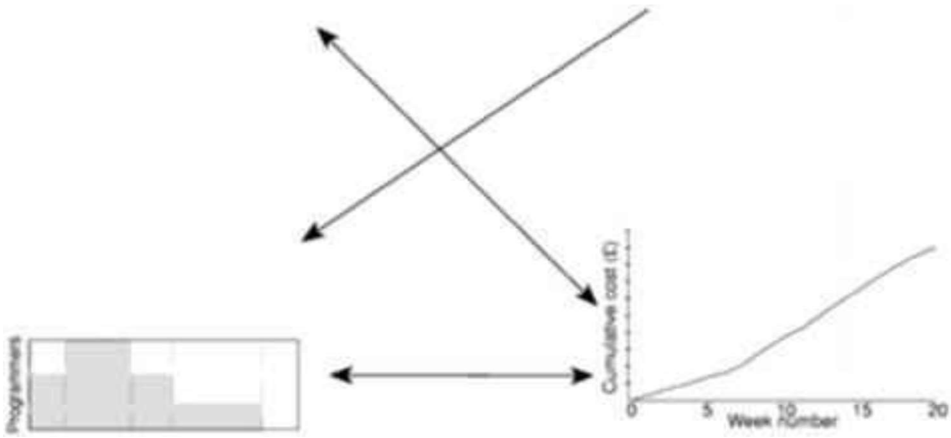


Figure 8.11 Successful project scheduling is not o simple sequence.

Cost schedule

Resource allocation

Figure 8.11 Successful project scheduling is not o simple sequence.

Cost schedule

Cod* Wodu*

Oocu^wtC

Resource allocation

While good project planning software will assist greatly in demonstrating the consequences of change and keeping the planning synchronized, successful project scheduling is largely dependant upon the skill and experience of the project manager in juggling the many factors involved