Network Planning Models

Introduction:

- These project scheduling techniques model the project's activities and their relationships as a network. In the network, time flows from left to right.
- The two best known being CPM (Critical Path Method) and PERT (Program Evaluation Review Technique).
- Both of these techniques used an activity-on-arrow approach to visualizing the project as a network where activities are drawn as arrows joining circles, or nodes which represent the possible start and/or completion of an activity or set of activities.
- More recently a variation on these techniques, called precedence network, has become popular. This method uses activity-on-node networks where activities are represented as nodes and the links between nodes represent precedence (or sequencing) requirements.
- This latter approach avoids some of the problems inherent in the activity-on-arrow representation and provides more scope for easily representing certain situations. It is this method that is adopted in the majority of computer applications currently available. These three methods are very similar and it must be admitted that many people use the same name (particularly CPM) indiscriminately to refer to any or all of the methods.
- In the following sections of this chapter, we will look at the critical path method applied to precedence (activity-on-node) networks followed by a brief introduction to activity-on-arrow networks

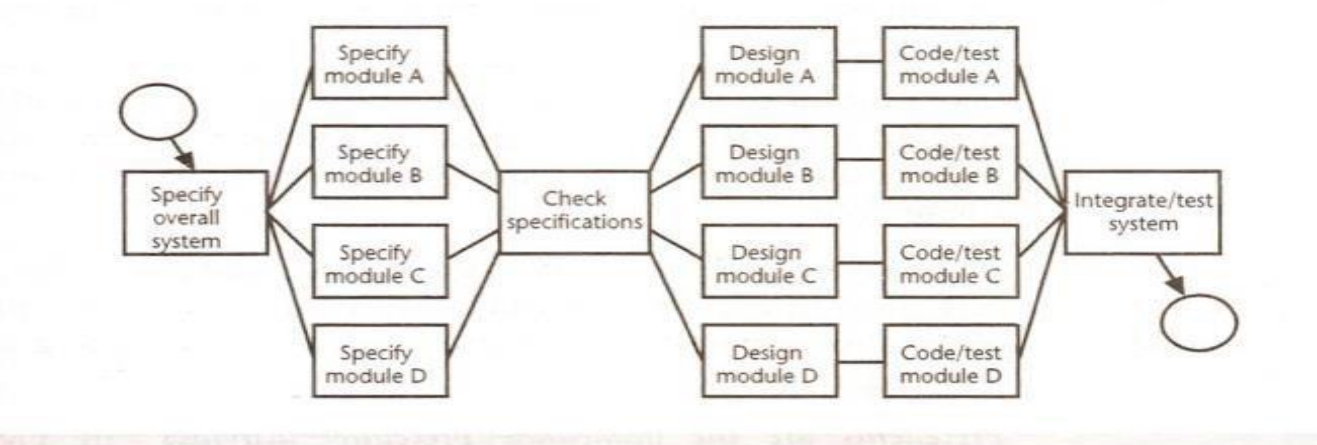

Fig: The IOE maintenance group accounts project activity network fragment with a check point activity added

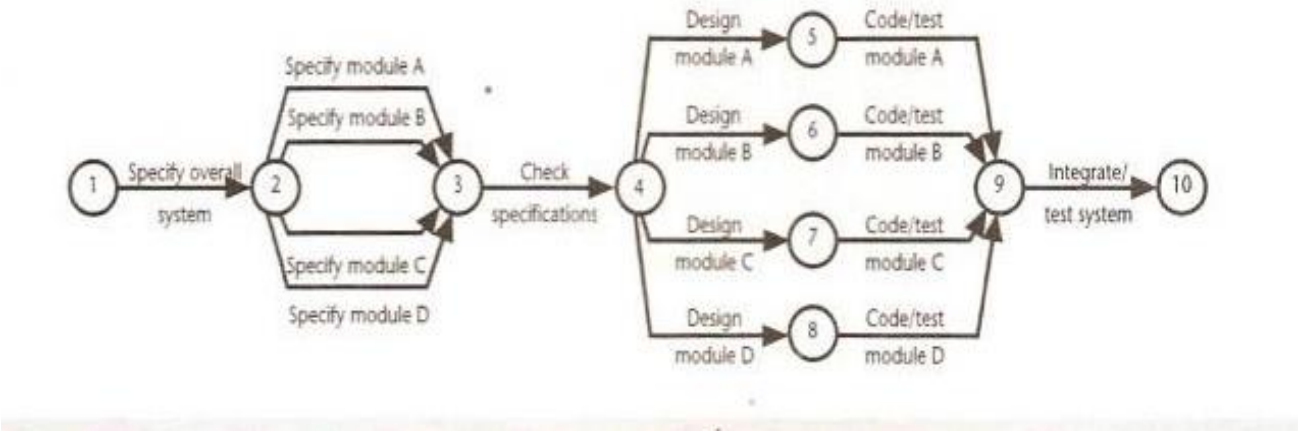

Fig: The IOE maintenance group accounts project activity network fragment represented as a CPM network

:

Formulating a network model

- The first stage in creating a network model is to represent the activities and their interrelationships as a graph. In activity-on-node we do this by representing activities as links (arrowed lines) in the graph — the nodes (circles) representing the events of activities starting and finishing.

Constructing precedence networks

- A project network should have only one start node
- A project network should have only one end node
- A node has duration
- A node represents an activity and, in general, activities take time to execute.
- Links normally have no duration
- precedents are the immediate preceding activities In Figure, the activity 'Program test' cannot start until both 'Code' and 'Data take-on' have been completed and activity 'Install' cannot start until 'Program test' has finished. 'Code' and Data take-on' can therefore be said to be precedents of 'Program test', and 'Program test' is a precedent of 'Install'. Note that we do not speak of 'Code' and 'Data take-on' as precedents of 'Install' - that relationship is implicit inthe previous statement. Time moves front left to right

Formulating a network model

- The first stage in creating a network model is to represent the activities and their interrelationships as a graph. In activity-on-node we do this by representing activities as links (arrowed lines) in the graph — the nodes (circles) representing the events of activities

starting and finishing.

Constructing precedence networks

- A project network should have only one start node
- A project network should have only one end node
- A node has duration
- A node represents an activity and, in general, activities take time to execute.
- Links normally have no duration
- precedents are the immediate preceding activities In Figure, the activity 'Program test' cannot start until both 'Code' and 'Data take-on' have been completed and activity 'Install' cannot start until 'Program test' has finished. 'Code' and Data take-on' can therefore be said to be precedents of 'Program test', and 'Program test' is a precedent of 'Install'. Note that we do not speak of 'Code' and 'Data take-on' as precedents of 'Install' - that relationship is implicit inthe previous statement.
- Time moves front left to right

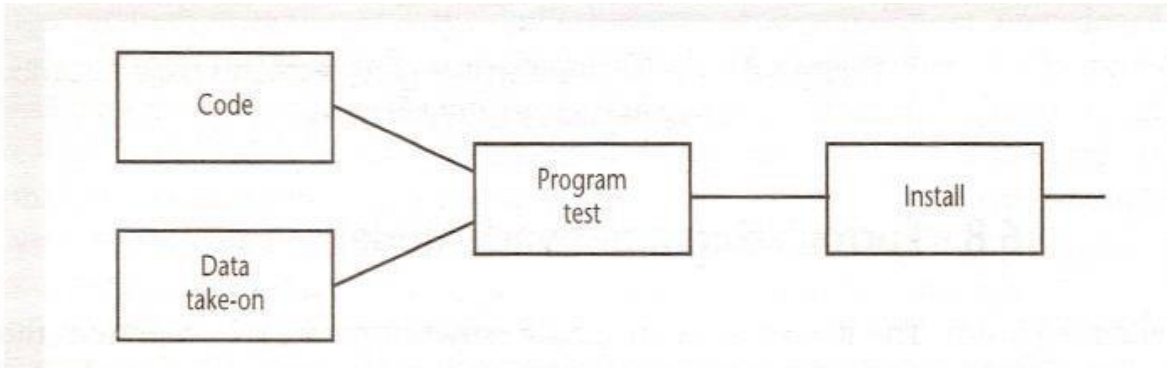

Fig : Fragment of a precedence network

- A network may not contain loops, Figure demonstrates a loop in a network. A loop is anerror in that it represents a situation that cannot occur in practice. While loops, in the sense of iteration, may occur in practice, they cannot be directly represented in a project network.

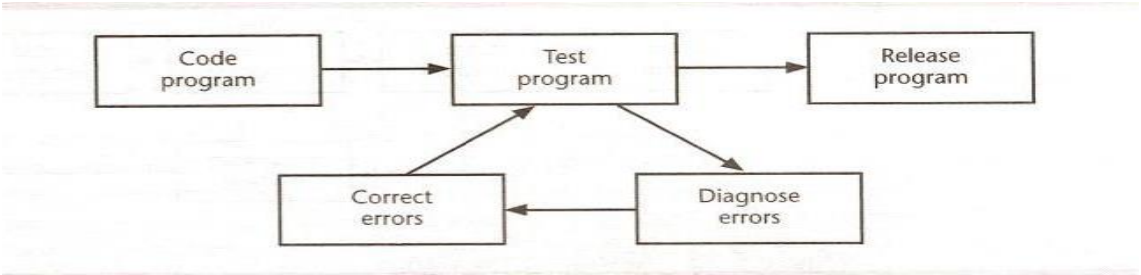

Fig: A Loop representing an impossible sequence

- A network should not contain dangles. A dangling activity such as 'Write user manual' in Figure :should not exist as it is likely to lead to errors in subsequent analysis.
- Redraw the network with a final completion activity — which, at least in this case, is probably a more accurate

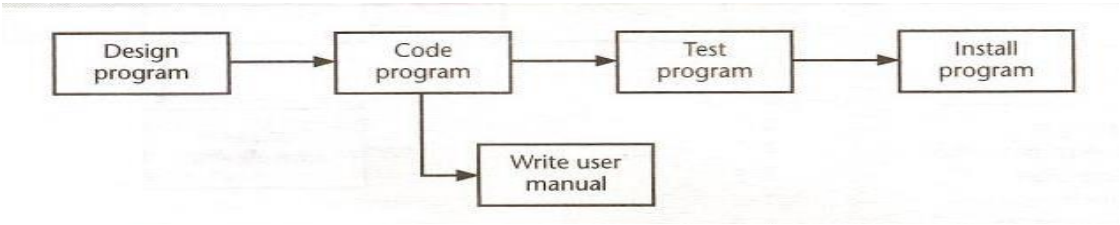

Fig: A Dangle

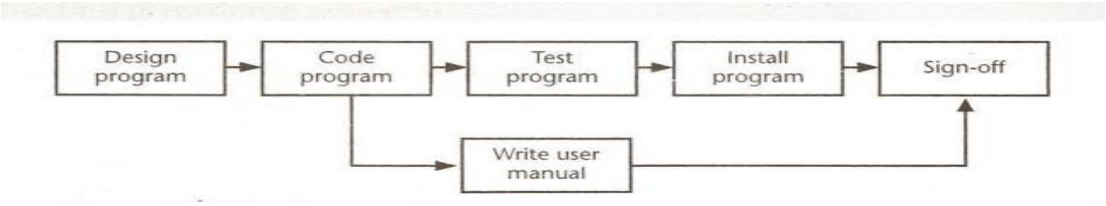

Fig: Resolving the dangle

Representing lagged activities

- We might come across situations where we wished to undertake two activities in parallel so long as there is a lag between the two. We might wish to document amendments to a program as it was being tested - particularly if evaluating a prototype.
- Where activities can occur in parallel with a time lag between them we represent the lag with a duration on the linking arrow as shown in Figure 6.13. This indicates that documenting amendments can start one day after the start of prototype testing and will be completed two days after prototype testing is completed.

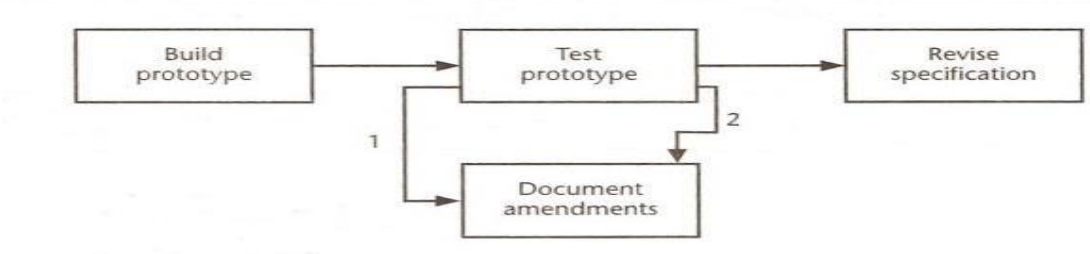

Fig: Indicating lags