Project Management

Session 3- Project Planning and Network Diagrams
Andre Samuel
This Session

• Planning Process
• Task Dependency
• Network Analysis/Critical Path Analysis
‘The Plan is nothing, Planning is everything’

Lord Nelson and General Eisenhower
Planning

• It is about *sequencing* project activities and milestones into a sensible and logical order.

• It involves linking activities to form a project network to show how different activities are related to each other.

• Planning will establish the **timing** of the activities and resource requirements.
Purpose of Planning

• Its is used to find the expected **project duration and milestone dates**

• It is used to make sure that the project is completed by a target date e.g. opening of the Olympic Games

• To get stakeholder commitment
The Planning Process

- Identify Activities
- Determine Sequence of activities
- Prepare Estimates of Time and Resources
- Present Plan in a readily intelligible format
Identifying Activities

- Terms of reference
- Scope of works
- WBS
- Milestone Planning
Example WBS
Tasks Dependencies

• A relationship between two linked tasks
• Linked by a dependency between finish and start dates
• Defining task dependencies correctly results in a schedule that's easy to maintain regardless of what happens or how quickly
• There are four types of dependency: Finish to Start, Start to Start, Finish to Finish and Start to Finish
Chronology vs Control

• Although tasks are called either successors or predecessors, a dependency is not really about chronology, but about control.

• A task dependency specifies which of two tasks controls the scheduling of the other.

• The independent task (the predecessor) determines the scheduling of the dependent task (the successor).

• A predecessor can occur before, at the same time as, or after its successor.
Task dependencies vs Date Constraints

• Applying dependencies to tasks is preferable to setting hard dates in a schedule.
• Fixed dates applied to tasks, called date constraints, quickly increase the effort to maintain the schedule.
• If you use date constraints, you often must manually recalculate dates when the schedule changes.
• However, date constraints don't always cause schedule problems.
• For example, when tasks occur on specific dates, such as training classes or conferences, date constraints keep those events tied to the correct dates on the calendar.
Dependency Types

- Finish-to-Start (FS)
- Start-to-Start (SS)
- Finish-to-Finish (FF)
- Start-to-Finish (SF)
Finish-to-start (FS)

• This type of dependency is the most common, perhaps because control and chronology work in the same direction in this dependency. After the predecessor task finishes, the successor task starts.

• This means activity A must finish before activity B has permission to start.

• For example:
  – After members of a construction crew set up the concrete forms for a foundation, they start to pour the concrete into the forms.
Start-to-start (SS)

• This type of dependency indicates that the start of one task triggers the start of the second task. A lag time between tasks is often used with this type of dependency.

• This means that once activity A has started, activity B can also start.

• For example:
  – Members of a road crew start to place traffic cones to close a lane on the highway. Ten minutes (lag time), after they start the line-painting machine starts to paint lines.
Finish-to-finish (FF)

• This type of dependency includes one task that continues only as long as another task is in progress.
• This means activity A must finish before activity B can finish

• For example:
  – Traffic flaggers direct traffic until construction work is completed.
  – In a software project, two programmers may be working on two different modules. If module B makes use of module A, then module B cannot be completed until the first programmer has completed module A.
Start-to-finish (SF)

- This type of dependency turns most people's concept of predecessor and successor tasks upside down. The confusion occurs because in most start-to-finish cases, the predecessor occurs after the successor. And the start of the predecessor controls the finish of the successor.
- This means that as long as A has started, we can proceed to finish B.
- Activity A must start before Activity B can finish.
Examples of S-F

• I have to place the forms for a column after these are removed from another already poured and cured. I know what the starting date to start forming the new column is (fixed date) and need to install the rebar for that column previously. Therefore, I can link column formwork installation (B) and rebar installation (A) with a SF relationship meaning that activity B must start for activity A to be considered finished.

• Say you’re building a new gas pipeline. You would first finish construction and implementation of the new pipeline before you would begin shutting down and breaking down the old pipeline.
• **Lead Time**- is overlap between tasks that have a dependency.
  
  – For example, if a task can start when its predecessor is half finished, you can specify a finish-to-start dependency with a lead time of 50 percent for the successor task. You enter lead time as a **negative value** in MS Project.
• **Lag Time**- is a delay between tasks that have a dependency.
  
  – For example, if you need a two-day delay between the finish of one task and the start of another, you can establish a finish-to-start dependency and specify two days of lag time. You enter lag time as a **positive value** in MS Project.
Project Planning Tools

• Network Diagram (Critical Path Analysis)
• Gantt Charts
Network Diagrams

• Network diagram show the relationships between activities

• They are used to:
  – calculate the Project Duration
  – Reveal critical tasks which warrant greater management attention
Activity of Node Method

• In this approach each activity of the project is represented by a Node i.e. an activity box.

<table>
<thead>
<tr>
<th></th>
<th>EST</th>
<th>DUR</th>
<th>EFT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task Name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LST</td>
<td>FLOAT</td>
<td>LFT</td>
<td></td>
</tr>
</tbody>
</table>

• Dur = Duration in time units
• EST = Earliest Start Time
• EFT = Earliest Finish Time
• LST = Latest Start Time
• LFT = Latest Finish Time
Labels for Activity Box

- **EST** - The earliest point in time when the schedule activity can begin
- **EFT** - The earliest point in time when the schedule activity can be complete
- **LST** - The latest point in time when the schedule activity can begin so as not to delay the project completion date or any constraint.
- **LFT** - The latest point in time when the schedule activity can finish so as not to delay the project completion date or any constraint.
## Dependency Table for a Software Project

<table>
<thead>
<tr>
<th>ID</th>
<th>TASK NAME</th>
<th>DURATION</th>
<th>PREDECESSORS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A1 CLIENT WORKSHOP</td>
<td>10</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>A2 CASE MODEL</td>
<td>15</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>A3 USER REVIEW</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>B1 Db DEFINITION</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>B2 FORMS DESIGN</td>
<td>20</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>B3 REPORT DESIGN</td>
<td>5</td>
<td>3,5</td>
</tr>
<tr>
<td>7</td>
<td>C1 SYSTEM TESTING</td>
<td>20</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>C2 HANDOVER TO CLIENT</td>
<td>5</td>
<td>6,7</td>
</tr>
</tbody>
</table>
Two Parts to the Analysis

• Forward Pass
  – Calculates the Duration of the Project
• Backward Pass
  – Calculates the slack/float for each task and shows the critical path
**Forward Pass** to calculate EST and EFT

- The EST for the first activity is zero
- EFT for an activity is always found by adding its duration to its EST, i.e. $EFT = EST + \text{Duration}$
- The EST for all remaining activities is the same as the EFT of its immediate predecessor, i.e. $EST = EFT$ of preceding activity
- In the case of convergence, the EST is taken from the path having the highest EFT
- The EFT of the last activity is the duration of the project
Forward Pass
Reverse Pass to calculate LST and LFT

- The LFT for the last activity is the same as its EFT
- The LST for an activity is always found by subtracting its duration from its LFT i.e.
  \[ \text{LST} = \text{LFT} - \text{Duration} \]
- The LFT for all remaining activities is given by the LST of its immediate successor i.e.
  \[ \text{LFT} = \text{LST of the successor activity} \text{ (moving from start to finish)} \]
- In case of convergence, the LFT is taken from the path having the lowest LST
Float/Slack

• This is the spare time for an activity
• This means that a certain amount of delay can be tolerated without affecting the expected project duration
Types of Float/Slack

- **Total Float** - Total float is the period by which an activity can shift in its timing without affecting the relevant schedule completion date (either a constraint, or the project completion).

- **Free Float** - Free float is the period by which an activity can shift in its timing without affecting the Early Start of any succeeding activity or event.
Calculating Float

• **Float/Slack** = LST – EST
  
  **OR**  = LFT - EFT
Float/Slack
Defining a Critical Task

• A critical task is one where EFT = LFT. Thinking this through it means that the activity MUST start on that date, the EST or the entire project duration is affected

• **Simply put an activity is Critical when the Float = 0**

• A non-critical task is where EFT < LFT. This means that although the activity could start as early as the EST, providing it finishes by the LFT the project could still finish on schedule
Defining the Critical Path

• Critical Path-
  – The critical path is the longest sequence of activities from commencement to completion of the project
  – The critical path is the series of tasks (or even a single task) that dictates the calculated finish date of the project.
  – That is, when the last task in the critical path is completed, the project is completed.
  – It is identified by the series of task where the Float or Slack is ‘Zero’
Importance Network Analysis

• **Critical activities** must stay on schedule as they are most likely to cause delays
• Project managers must concentrate their time on these tasks
• Conversely, knowing the float for non-critical activities will let the project manager know how long they can be delayed for before impacting the entire project.
Project Crashing

• This means adding resources to one or more critical path activities

• In the short term this may actually reduce productivity because of the learning curve effect

• Crashing incurs additional cost, but must be weighed up against finishing late
Fast Tracking

• Shortening the project duration by overlapping project activities or phases
• Perform critical path tasks in parallel that were originally planned sequentially
Optimizing through Resource Levelling and Smoothing

• When you assign resources (staff) to a project it can often lead to over allocation or staff conflict

• For example resource A is assigned to an activity on Project XYZ on a day, for 8 hours. This resource is also allocated to Project LMN for 8 hours in a day

• Or resource A is assigned to work on multiple activities on a project occurring at the same time

• How do we resolve this?
Resource Levelling

• A technique in which start and finish dates are adjusted based on resource constraints with the goal of balancing demand for resources with the available supply

• This involves removing the over allocation of resources. This can be done by:
  – Delay certain task
  – Assign a different resource
  – Change task dependencies
Resource Smoothing

• Resource smoothing means the uniform distribution of resource allocation over the project.

• Let's say that you have a resource working 30 hours this week, 10 hours next week, and 20 hours the week after next week. When you apply resource smoothing, the resource will work 20 hours this week, 20 hours next week, and 20 hours the week after. You see, the total amount of hours remains 60 hours, but is more uniformly distributed over the weeks.
Levelling and Smoothing Scenario
How to shorten project duration?

- Project crashing
- Fast Tracking
- Resource Levelling and Smoothing
- Re-scoping - reduce scope to reduce time
- Increased procurement - outsource work
- Lean thinking – redesign work processes
- Renegotiate target dates if possible