SOFTWARE PROJECT MANAGEMENT

TIME MANAGEMENT

OBJECTIVES

- To
  - Get a (very general) overview of project management concepts
  - To understand how to manage time in a software project

1.1 Introduction

What is a project?

- A **project** is a temporary endeavor undertaken to achieve particular goals and objectives.

What is project management?

- **Project management** is the use of skills and knowledge to successfully plan, manage and complete a project.
- Project management skills are industry independent.

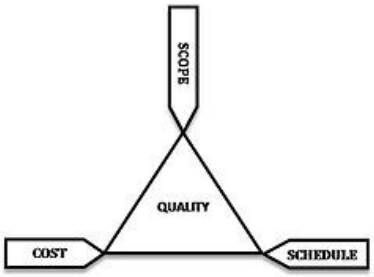
1.2 Who are stakeholders?

- Stakeholders are individuals and organisations
  - actively involved in the project,
  - who are affected by the outcome of the project and
  - who may have influence over the project.
- They include:
  - Project team
  - Clients/Customers
  - Executive sponsor
  - Public agencies

1.3 What Are The Constraints On A Project?

- Projects are constrained by four factors:
  1. **Cost (budget)**
  2. **Schedule (time)**,
  3. **Scope**
  4. **Quality**
     - CSSQ
- Pressure on schedule, scope and budget each affect the **quality** of the project outcome

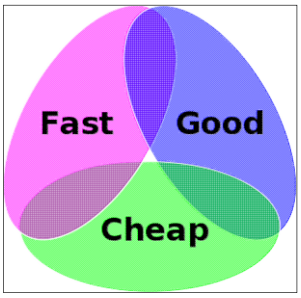
1.4 The Project Triangle



1.5 Euler vs Venn Diagrams

- A **Venn** diagram is a mathematical illustration that shows all of the possible mathematical or logical relationships between sets.
- A **Euler diagram** resembles a Venn diagram, but **does not necessarily show all possible intersections of the sets.**
- A **Euler diagram** is often more useful for showing real world data, because not all sets partially overlap with all other sets.

1.6 Euler diagram



No Overlap of all three

1.7 POSSIBLE OUTCOMES

•Design something quickly and to a high standard, but then it will **not be cheap**.

•Design something quickly and cheaply, but it will **not** be of **high quality**.

•Design something with high quality and cheaply, but it will **take a long time**.

1.8 Skills To Manage Constraints

- The discipline of Project Management is about providing the tools and techniques that enable the project team (not just the project manager) to organise their work to meet these constraints.

1.9 What Is The Critical Path?

1. Critical path has been defined as the series of activities from project start to project finish that defines the total duration of the project.
2. A critical path is the sequence of project network activities which add up to the **longest** overall duration.
   - Microsoft Office Project defines a task as critical path if **changes to the finish date of the task affect the finish date of the project**.

1.10 What Are Project Phases?

- Project phases are the stages that every project passes through
- For example, a **controlling phase** may include filing a change of scope request to track changes made to the scope document for the project.

1.11 Project Phases Include:

PHASE	ACTIVITIES IN THE PHASE	OUTPUT
Initiation	Defining the project, creating a scope document, identifying stakeholders and building a project team	Project Charter, key project team members identified.
Planning	Budgeting, scheduling and planning activities.	Scope Document, Budget, Microsoft Project plan (if using MS Project).
Execution	Executing the plan and adapting to stakeholders' expectations.	Reports from project
Monitoring and Controlling	Monitoring progress, balancing demands of time, scope and quality, tracking corrective actions, and reporting progress to stakeholders.	Reports from project
Closure	Handing off to end users, closing down operations, and reporting outcomes.	Reports from project

1.12 Criteria For Success

- For the purposes of this course, a software project will be considered successful if:
  1. The software is delivered **on schedule**
  2. Development costs were **within budget**
  3. The software **meets the needs of users** (in scope and quality)

(Felici, 2011)

Configuration management

- Configuration management is the name given to the general process of managing a changing software system.
- The aim of configuration management is to support the system integration process so that all developers can access the project code and documents in a controlled way, find out what changes have been made, and compile and link components to create a system.
- Configuration management activities include:
  - Version management
  - System integration
  - Problem tracking

Configuration management activities

- **Version management**, where support is provided to keep track of the different versions of software components. Version management systems include facilities to coordinate development by several programmers.
- **System integration**, where support is provided to help developers define what versions of components are used to create each version of a system. This description is then used to build a system automatically by compiling and linking the required components.
- **Problem tracking**, where support is provided to allow users to report bugs and other problems, and to allow all developers to see who is working on these problems and when they are fixed.

Configuration management tools

- http://code.google.com/edu/tools101/scm.html
- https://developers.google.com/university/tools/

17

PM Knowledge Areas

18

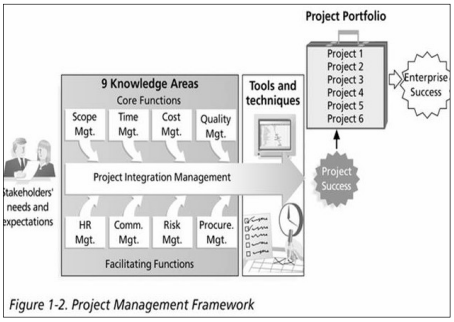


Figure 1-2. Project Management Framework

p. 8, Schwalbe (2009) https://www.e-education.psu.edu/geog584/11_p6.html

Project Planning

- Involves devising and maintaining a workable scheme to ensure that the project addresses the organisation's needs.
- Planning is crucial in information technology projects because once a project team implements a new system, it takes a considerable amount of effort to change the system

Project Planning

- Research suggests that companies working to implement best practices should spend at least 20% of project time in initiating and planning *(Schwalbe p.81)* .

Importance of Project Schedules

- Managers often cite delivering projects on time as one of their biggest challenges
- Time has the least amount of flexibility; it passes no matter what happens on a project
- Schedule issues are the main reasons for conflicts on projects, especially during the second half of projects

Individual Work Styles and Cultural Differences Cause Schedule Conflicts

- Peoples' attitudes toward structure and deadline differ
  - Some people prefer to follow schedules, meet deadlines and have closure.
  - Others prefer to keep things open and flexible; deadlines are a signal to start rather than to complete a project
- Different cultures and even entire countries have different attitudes about schedules

Project Time Management Processes

1. **Activity definition**:
   - identifying the specific activities/tasks that the project team members and stakeholders must perform to produce the project deliverables
2. **Activity sequencing**:
   - identifying and documenting the relationships between project activities
3. **Activity resource estimating**:
   - estimating how many **resources** a project team should use to perform project activities

Project Time Management Processes

4. **Activity duration estimating**:
   - estimating the number of work periods that are needed to complete individual activities
5. **Schedule development**:
   - analysing activity sequences, activity resource estimates, and activity duration estimates to create the project schedule
6. **Schedule control**:
   - controlling and managing changes to the project schedule

Activity Definition

Identifying the specific activities/tasks that the project team members and stakeholders must perform to produce the project deliverables

- Project schedules are built from the
  - **Project charter**
    - the document that formally authorises a project
    - Assigns a project manager and provides him/her and the Project Team the authority to use resources for the purpose of undertaking the project
    - may contain brief general information about the project; or may contain specific details
    - documents initial requirements and expectations
  - **Scope statement** (what is and what is not in a project – products/deliverables and the processes used to create them) and **WBS** (major work packages)
- Activity definition involves developing a more detailed WBS and supporting explanations to understand all the work to be done so you can develop realistic cost and duration estimates

Activity Definition

Identifying the specific activities/tasks that the project team members and stakeholders must perform to produce the project deliverables

- The basis for creating a project schedule is derived from four project time management processes
  - Activity definition – further defining the scope
  - Activity sequencing – further defining the time
  - Activity resource and activity duration estimation (further defining the time and cost)

Milestones

- A **milestone** is a **significant** event that normally has no duration
  - Not every deliverable or output created for a project is a milestone
- It often takes several activities and a lot of work to complete a milestone
- Useful for setting schedule goals and monitoring progress
- **Examples**
  - obtaining customer sign-off on key documents or
  - completion of specific products such as software modules or
  - the installation of new hardware

Activity Sequencing

Identifying and documenting the relationships between project activities

- After defining project activities, determine the relationships between activities
- A **dependency** or **relationship** is the sequencing of project activities or tasks
- Dependencies must be determined if one is to use **critical path analysis**

Aside: Planning, Estimating, Scheduling

- What's the difference?
  - **Plan**:
    - Identify activities. No specific start and end dates (WBS).
  - **Estimating**:
    - Determining the size & duration of activities.
  - **Schedule**:
    - Adds specific start and end dates, relationships, and resources. (**Gantt**)

Aside: How To Schedule

1. Identify "**what**" needs to be done
   - Work Breakdown Structure (**WBS**)
2. Identify "**how much**" (the size)
   - Size estimation techniques
3. Identify the **dependency** between tasks
   - Dependency graph, network diagram
4. Estimate **total duration** of the work to be done
   - The actual schedule

Creating a Task List/How to

- Building a house
  - Main task/objective (Node)
  - Sub-tasks
  - Work-packages
- Brainstorming
  - Generate all activities you can think of that need to be done
  - Group them into categories
- Remember to get the people who will be doing the work involved (buy-in matters!)

Task List – Basis of Many Things

- Work Breakdown Structure
- Network scheduling
- Costing
- Risk analysis
- Organisational structure
- Control
- Measurement

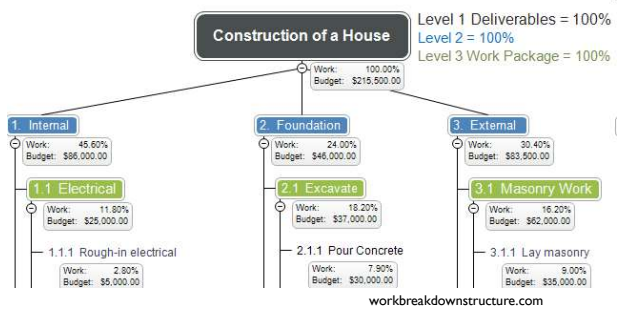
Create a Work Breakdown Structure

- A work breakdown structure (WBS) is a project management tool designed to capture project tasks in a visual, organised manner.
- It is a deliverable oriented decomposition of a project into smaller components.
- A well defined WBS will provide the PM, the project team and stakeholders with a clear vision about the work needed to complete the project.
- Work breakdown structures are widely used for projects of all types
- It can be likened to a **graphical to-do-list**
  - A simple, task-oriented WBS resembles a graphical To Do list.
  - … but in the corporate world there is more to developing and using a WBS than simply creating an expanded To Do list (Gordon, A, 2011).

Work Breakdown Structure: WBS

- Hierarchical list of project's work activities
- Uses a decimal numbering system
  - E.g. 3.1.5
- Does **not** show dependencies or durations
- For further info:
  - ***Tips for Building a Work Breakdown Structure***
  - *http://www.brighthub.com/office/project-management/articles/2940.aspx*
  - *http://en.wikipedia.org/wiki/Work_breakdown_structure*
  - *http://www.project-blog.com/?page_id=64*

WBS EXAMPLE



Why do a WBS?

- **Scope.** The WBS is very good to use when determining, defining and communicating the scope and deliverables of the project.
- **Manageable pieces.** WBS enables break down of tasks into smaller pieces making it easier to follow up and manage the work.
- **Reporting and follow up.** The structure of the WBS will create a framework for reporting and follow up.
- **Foundation.** The information in the WBS will act as input to other Project Management Processes, such as planning and risk management.
- **Communication.** The WBS is a communication tool both internally and externally, creating a common understanding between PM, stakeholders and project members.
- **Success rate.** A project without a proper WBS lacks the correct pre-condition to succeed.

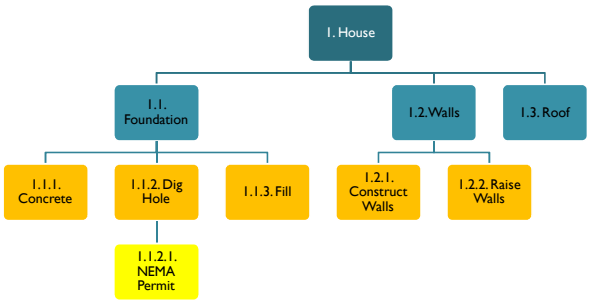
How to Create a WBS

- Creating a work breakdown structure (WBS) helps you be both comprehensive and specific when managing a project. It helps you identify major parts of the project
1. Determine the major deliverables or products to be produced.
2. Divide each of these major deliverables into its component deliverables in the same manner.
3. Divide each of these work pieces into its component parts.
- You should focus on capturing the deliverables in the project.
- A Deliverable is a **unique product or result** that must be completed to be able to perform the project. A deliverable can be internal or external.

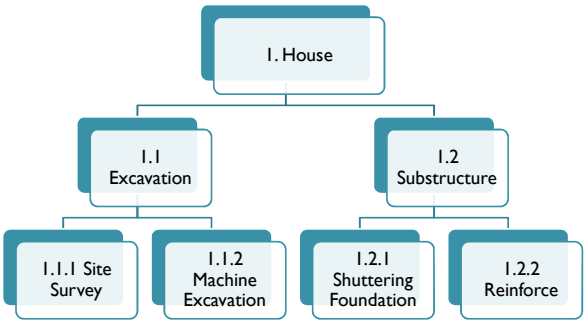
Level of breakdown

- Each project is different and you need to find a suitable level of decomposition for your specific project
- There are no hard rules for how many levels you should have.
- A large and complex project will likely have more levels than a small and straight forward project.

WBS Example



WBS Example



Schedule Development

analysing activity sequences, activity resource estimates, and activity duration estimates to create the project schedule

- Once tasks (from the WBS) and size/effort (from estimation) are known, then **schedule**
- **Primary objectives**
  - Best time
  - Least cost
  - Least risk
- **Secondary objectives**
  - Evaluation of schedule alternatives
  - Effective use of resources
  - Communications

Terminology

- **Precedence**:
  - A task that must occur before another is said to have precedence over the other
- **Concurrence**:
  - Concurrent tasks are those that can occur at the same time (in parallel)
- **Leads & Lag Time**
  - Delays between activities
  - Time required before or after a given task
- **Float & Slack (synonymous terms)**
  - The time an activity has before it delays next task
    - *(the amount of time you can delay an activity without delaying the earliest start (Early Start/ES) of any succeeding activities.)*

Terminology

- **Leads vs Lag Time**
  - Lead and lag are two terms associated with the relationships that may occur between multiple schedule activities.
  - Lead is the **acceleration of a successor activity**.
    - In other words, the second activity can begin (and be conducted in parallel) as the first activity.
    - Lead is only found activities with finish-to-start relationships: A must finish before B can start.
    - In order to leverage a lead, which will compress the total combined duration of both activities, the dependency must be discretionary, meaning that **there is no physical limitation on completing A before B begins**.

Terminology

- **Lag**
  - Lag is the delay of a successor activity and represents time that must pass before the second activity can begin.
  - There are no resources associated with a lag.
  - Lag may be found in activities with all relationship types:
    - finish-to-start,
    - start-to-start,
    - finish-to-finish, and
    - start-to-finish.

Terminology

- **Example**
  - The photo shoot will take four days and the photo editing will take six days.
  - Instead of waiting until the end of the 4-day photo shoot to begin editing the pictures, we start editing after the first day of shooting.
  - If you do a start-finish, what will be the total duration?
  - What about if you leverage the lead?
  - The total duration will be brought down from ten days to seven days by leveraging the **lead**.
- The photo proofs are sent to the customer upon completion of the shoot, however, there is a 15-day **lag** associated with the ***customer review*** before the ***printing of the photos*** can begin.

Terminology

- **Summary**
- Lead and lag are both used in the development of the project schedule.
- Lead
  - is an acceleration of the successor activity and
  - can be used only on finish-to-start activity relationships.
- Lag
  - is a delay in the successor activity and
  - can be found on all activity relationship types.

Terminology

- When it comes to each activity in the project, there are four parameters for each related to the timelines:
- **Earliest start time (ES)**
  - The earliest time an activity can start once the previous dependent activities are over.
- **Earliest finish time (EF)**
  - ES + activity duration.
- **Latest finish time (LF)**
  - The latest time an activity can finish without delaying the project.
- **Latest start time (LS)**
  - LF - activity duration.

Terminology

- **Float & Slack (synonymous terms)**
  - The time an activity has before it delays next task
    - *(the amount of time you can delay an activity without delaying the earliest start (Early Start/ES) of any succeeding activities.)*
- **Slack Time $T_S = T_L - T_E$**
  - T_E = earliest time an event can take place
  - T_L = latest date it can occur w/o extending project's completion date
- A **forward pass** through a network diagram determines the early start and early finish dates
- A **backward pass** determines the late start and late finish dates

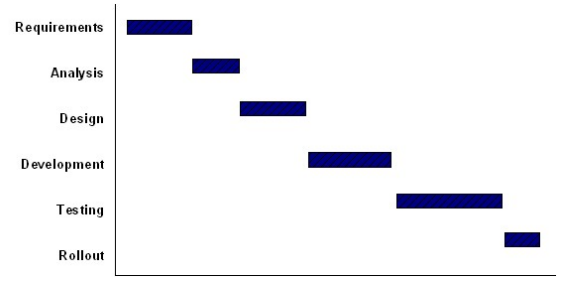
Terminology

- Milestones
  - Have a duration of zero
  - Identify critical points in your schedule
  - Shown as inverted triangle or a diamond
  - Often used at "review" or "delivery" times
  - Or at end or beginning of phases
    - E.g. Software Requirements Review (SRR)
    - E.g. User Sign-off
  - Can be tied to contract terms

Scheduling Techniques

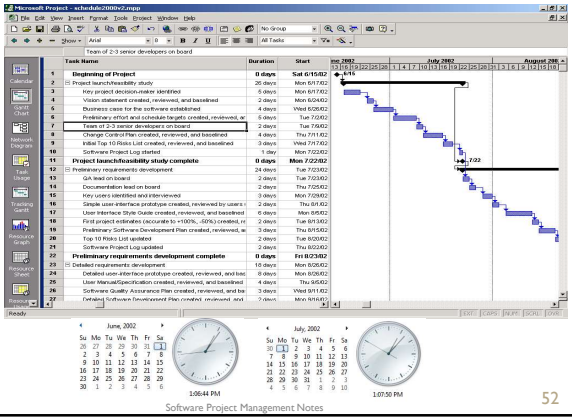
- Mathematical Analysis
  - Network Diagrams
    - PERT
    - CPM
- Bar Charts
  - Milestone Chart
  - Gantt Chart

Bar Chart
Simple Gant Chart



Either showing just highest summary bars
Or milestones only

Gantt Chart



Gantt Chart

- Disadvantages
  - Does not incorporate uncertainty of a given activity (as does PERT)
    - *Program Evaluation and Review Technique*
- Advantages
  - Easily understood
  - Easily created and maintained

Network Diagrams

- A **network diagram** is a schematic display of the logical relationships among, or sequencing of, project activities
  - It is a technique for showing activity sequencing.
- Two main formats are the arrow and precedence diagramming methods

Network Diagrams

- Developed in the 1950's
- A graphical representation of the tasks necessary to complete a project
- Visualizes the flow of tasks & relationships

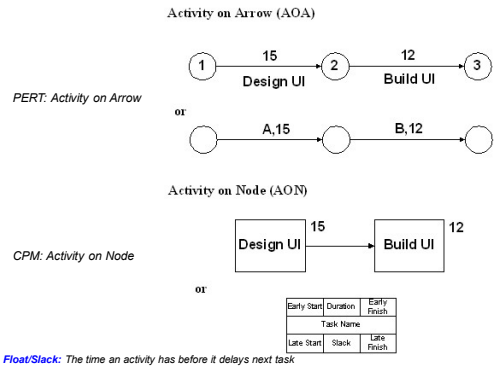
Network Diagrams

- PERT
  - Program Evaluation and Review Technique
- CPM
  - Critical Path Method
- Sometimes treated synonymously
- All are models using network diagrams

Network Diagrams

- Two classic formats
  - AOA: Activity on Arrow
  - AON: Activity on Node
- Each task labeled with
  - Identifier (usually a letter/code)
  - Duration (in a std. unit like "days")
- There are other variations of labeling
- There is one start & one end event
- Time goes from left to right

Node Formats



Float/Slack: The time an activity has before it delays next task

Network Diagrams

- AOA consists of
  - Circles representing Events
    - Such as 'start' or 'end' of a given task
  - Lines representing Tasks
    - Thing being done e.g. 'Build UI'
- AON
  - Tasks on Nodes
    - Nodes can be circles or rectangles (usually latter)
    - Task information written on node
  - Arrows are dependencies between tasks

CRITICAL PATH METHOD

- It is an algorithm (step-by-step procedure for calculations) for scheduling a set of project activities
- To use CPM to organise a project

1. Make a list of all activities required to complete the project (work breakdown structure),
2. Estimate the time that each activity will take to completion, and
3. The dependencies between the activities.
4. Create a network diagram by hand or by using diagram software (**Activity on Node**)

CPM

- **CPM** calculates
  - the **longest** path of planned activities to the end of the project, and
  - the earliest and latest that each activity can start and finish without making the project longer
- A critical path is **the sequence of project network activities which add up to the longest overall duration**

CPM

Benefits

- Helps PM develop and test plan to ensure that it is robust.
  - **Critical Path Analysis** formally identifies tasks which must be completed on time for the whole project to be completed on time.
- Helps identify the minimum length of time needed to complete a project
- If resources need to be reallocated to catch up on missed or overrunning tasks, it identifies which tasks can be delayed .

Disadvantages

- difficult to understand

  the relation of tasks to time is not as immediately obvious as with Gantt Charts

•As with Gantt Charts, in practice project managers use software tools like Microsoft Project to create CPA Charts. These SW tools:

- •make them easier to draw,
- •make modification of plans easier
- •provide facilities for monitoring progress against plans.

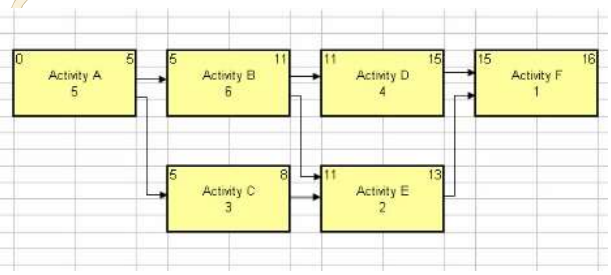
Calculating Early Start and Early Finish

- When you have a long critical path, but the other paths in your network diagram are short, then you have a lot of freedom in when you can start and finish each of the activities that are not on the critical path.
- You can use **early start** and **early finish** to know exactly how much freedom you have in your schedule to move the start dates for those activities around without causing problems.

Calculating Early Start and Early Finish Forward Pass

- The first step in the calculation process is the ***Forward Pass***.
- Here, the Early Start and Early Finish values for each activity, along with the overall Project Duration, are calculated.
- Start by placing a zero in the Early Start (**ES**) position of the first activity.
- Continue by using the following formulas:
  - **Early Start = Maximum (or Highest) EF value from immediate Predecessor(s)**
  - **Early Finish = ES + Duration**

Calculating Early Start and Early Finish AON Forward Pass



65

Calculating Early Start and Early Finish Forward Pass

- The early start value for Activity E is based on the maximum early finish from its two predecessors (Activity B and Activity C).
  - Based on the Finish-to-Start relationships shown, both Activity B and Activity C must both be finished before Activity E can start.
  - In other words, Activity E is waiting on whichever predecessor finishes the latest (in this case, Activity B).
  - The same situation exists for Activity F.
- The duration of the schedule (project duration) is shown in the **EF of the latest activity** in the network diagram i.e. 16.
  - assuming that the time-unit of the example schedule is based on days, the schedule duration is sixteen days.

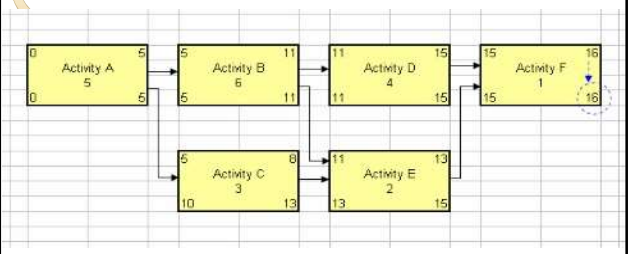
Calculating Late Start and Late Finish

- It's also important to know how late any activity can run before it delays the project.
- Late start and late finish allow you to work out how late you can start a certain task and how much it can slip before it delays your project.
- An activity with a large late start or late finish means you have more options.

Calculating Late Start and Late Finish Backward Pass

- Backward Pass is second step in the calculation.
  - The Late Start and Late Finish values are calculated.
- The formulas for the backward pass are:
- **Late Finish (LF) = Minimum (or Lowest) LS value from immediate Successor(s)**
- **Late Start = LF – Duration**
- You start at the last activity in the schedule and proceeds backward through the schedule until the Late Start value is computed for the schedule's beginning activity.

Calculating Late Start and Late Finish AON Backward Pass



Calculating Late Start and Late Finish Backward Pass

- To start the backward pass calculation, go to the last activity and copy its EF value to its LF value.
- Then apply the backward pass formulas for late start and late finish.
- Note:
  - the LF value for Activity B is based on the lowest LS shown by its two successors (Activity D and Activity E).

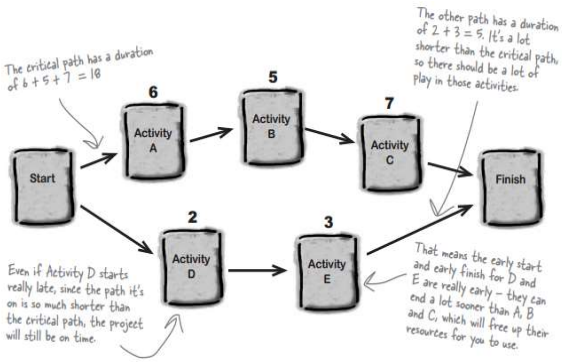
ES, EF, LS, and LF Overview

- Calculating the ES, EF, LS, and LF may seem complicated, but it only takes a little practice to get the hang of it.
- Once you walk through it step by step, you'll see that it's actually pretty easy!
- You won't have to do this kind of thing on the job... that's what computers are for!
- Project management software like **Microsoft Project** will do these calculations for you.
- But you need to know how to do it yourself, because when the computer is doing critical path analysis, this is exactly how it figures it out!

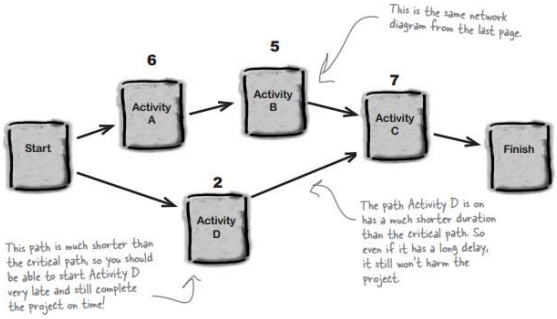
Calculating Early Start and Early Finish Using 1 as the First Number

- Some software give you the option to start the first number of the earliest EF (i.e. of the first activity) from either 0 or 1.
  - Some even have their default setting as starting with the number 1.
- You can then move forward with the calculations.
  - See the following slides.

Calculating Early Start and Early Finish

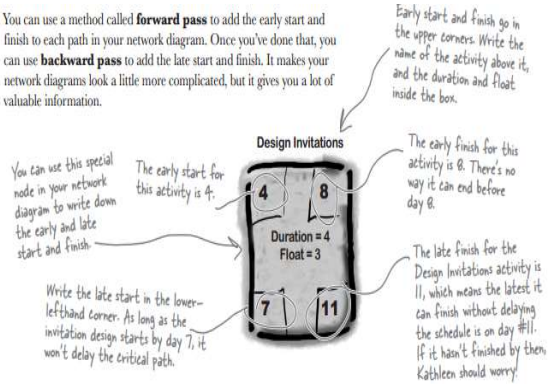


Calculating Late Start and Late Finish



Forward & Backward Pass

You can use a method called **forward pass** to add the early start and finish to each path in your network diagram. Once you've done that, you can use **backward pass** to add the late start and finish. It makes your network diagrams look a little more complicated, but it gives you a lot of valuable information.

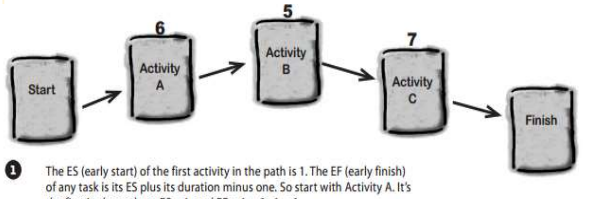


Float/Slack: *The time an activity has before it delays next task*

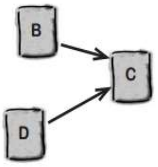
Forward Pass
Calculating the Early Start and Early Finish

- To calculate the **early start** and **early finish** start at the beginning of the critical path and move forward through each activity following these three steps:
  1. The ES (early start) of the first activity in the path is 1 (one). The EF (early finish) of any task is its ES plus its duration minus one.
  2. Now move forward to the next activity in the path. To figure out ES, take the EF of the previous task and add one.
  3. If an activity has two predecessors, to calculate its EF use the predecessor with the latest EF.

Forward Pass



1. The ES (early start) of the first activity in the path is 1. The EF (early finish) of any task is its ES plus its duration minus one. So start with Activity A. It's the first in the path, so ES = 1, and EF = 1 + 6 - 1 = 6.
2. Now move forward to the next activity in the path, which is Activity B in this diagram. To figure out ES, take the EF of the previous task and add one. So for Activity B, you can calculate ES = 6 + 1 = 7, and EF = 7 + 5 - 1 = 11.
3. **Uh-oh! Activity C has two predecessors.** Which one do you use to calculate EF? Since C can't start until both B and D are done, use **the one with the latest EF.** That means you need to figure out the EF of activity D (its ES is 1, so its EF is 1 + 2 = 3). Now you can move forward to Activity C and calculate its EF. The EF of Activity D is 2, which is smaller than B's EF of 11, so for Activity C the ES = 11 + 1 = 12, and EF = 12 + 7 - 1 = 18.

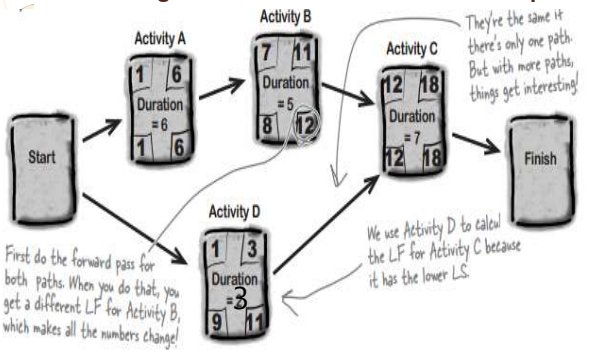


Backward Pass
Calculating the Late Start and Late Finish

- Start at the end of the path you just took a pass through and work your way backward to figure out the late start and finish.

1. Start at the end of the path, with Activity C. The LF (late finish) of the last activity is the same as the EF. Calculate its LS (late start) by subtracting its duration from the LF and adding one.
2. Now move backwards to the previous activity in the path—its LF is the LS of the activity after it minus one.
3. Do the same for all activities. LF is the LS for the activity after the current activity minus one.
4. Now you can move onto the next-longest path. If there are more paths, you'd then move on to the next-longest one, etc., filling in LF and LS for any nodes that haven't already been filled in.

Backward Pass
Calculating the Late Start and Late Finish Example



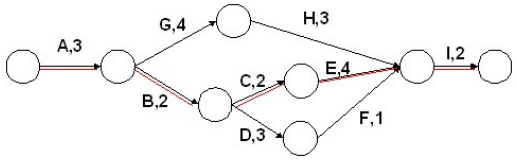
Backward Pass
Calculating the Late Start and Late Finish Example

1. Start at the end of the path, with Activity C. The LF (late finish) of the last activity is the same as the EF. Calculate its LS (late start) by subtracting its duration from the LF and adding one. LS = 18 - 7 + 1 = 12.
2. Now move backwards to the previous activity in the path—in this case, Activity B. Its LF is the LS of Activity C minus one, so LF = 12 - 1 = 11. Calculate its LS in the same way as step 1: LS = 11 - 5 + 1 = 7.
3. Now do the same for Activity A. LF is the LS for Activity B minus one, so LF = 7 - 1 = 6. And LS is LF minus duration plus one, so LF = 6 - 6 + 1 = 1.
4. Now you can move onto the next-longest path, Start-D-C-Finish. If there were more paths, you'd then move on to the next-longest one, etc., filling in LF and LS for any nodes that **haven't already been filled in.**

Critical Path

- "The specific set of sequential tasks upon which the project completion date depends"
  - **or "the longest full path"**
- All projects have a **Critical Path**
- Accelerating non-critical tasks do not directly shorten the schedule

Critical Path Example



Task Dependency Types

- **External** Dependencies
  - Outside of the project itself
    - E.g. Release of 3rd party product; contract signoff
      - Release of 3rd party product e.g. the release of the newest version of an OS
    - E.g. stakeholders, suppliers, outside contractors, year-end
- **Resource** Dependencies
  - Two tasks rely on the same resource
    - E.g. You have only one DBA but multiple DB tasks

PERT

- A **PERT** (Program Evaluation and Review Tool) chart is a (network diagram) tool that facilitates decision making.
- Two consecutive events in a PERT chart are linked by activities, which are conventionally represented as arrows (***Activity-on-arrow***)
- The events are presented in a logical sequence and no activity can commence until its immediately preceding event is completed.
- PERT is valuable to manage where multiple tasks are occurring simultaneously to reduce redundancy

PERT

- Similar to CPM – but **activity on arrow**
- PERT is a variation on Critical Path Analysis that takes a slightly more skeptical view of time estimates made for each project stage.
- To use it, estimate the shortest possible time each activity will take, the most likely length of time, and the longest time that might be taken if the activity takes longer than expected.
- Use the following formula to calculate the time to use for each project stage:

$$\frac{\text{shortest time} + 4 \times \text{likely time} + \text{longest time}}{6}$$

http://www.mindtools.com/critpath.html

PERT

- **Advantages**
  - Accounts for uncertainty
- **Disadvantages**
  - Time and labor intensive
  - Assumption of unlimited resources is big issue
  - Mostly only used on large, complex projects
- It is better to get PERT software to calculate it

PERT is seldom used today (PMBOK).

CPM vs. PERT

- Both use Network Diagrams
  - CPM: deterministic
    - Activity times are assumed to be known or predictable
  - PERT: probabilistic
    - Activity times are assumed to be random, with assumed probability distribution
- CPM: one estimate, PERT, three estimates
  - CPM: the longest path of planned activities to the end of the project
  - PERT: shortest time, likely time, longest time

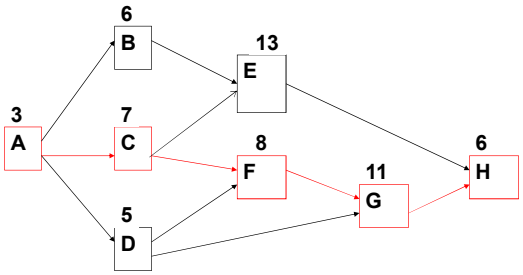
ASSIGNMENT

- Study the following table:

Activity	Completion time (days)	Immediate predecessor activities
A	3	-
B	6	A
C	7	A
D	5	A
E	13	B, C
F	8	C, D
G	11	D, F
H	6	G,E

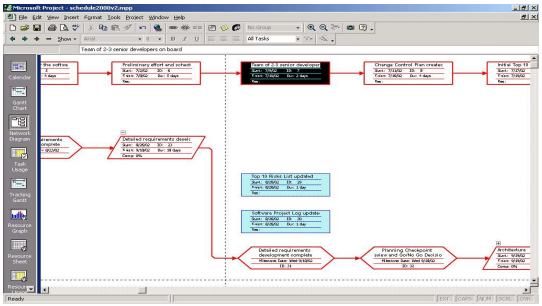
Required:
Construct a **network diagram** that satisfies the scheduling requirements shown:

CPM

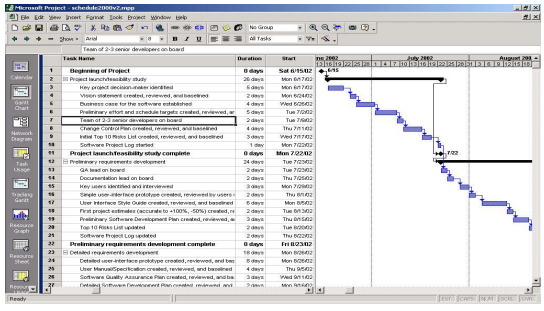


http://project.mvps.org/networkanalysis.htm on step by step

MS-Project Example



5.0. Gantt Chart



Features of a PMIS

1. List phases, tasks and sub-tasks
2. Create a Work Breakdown Structure (WBS)
3. Create network diagrams – Critical Path Analysis (CPA), Project Evaluation and Review Technique (PERT)
4. Produce varied reports for different stakeholders

Other requirements of a PMIS

5. Allow different team members to create and update items – collaboration
6. Budget, plans, schedules, calendars, email/chat
7. Dependent on organisation, budget, complexity of project, experience and capacity of the project manager and project team

Reducing Project Duration

- How can you shorten the schedule?
- Via
  - Reducing scope (or quality)
  - Adding resources
  - Concurrency (perform tasks in parallel)
  - Substitution of activities

Compression Techniques

- Shorten the overall duration of the project
- Crashing
  - Looks at cost and schedule tradeoffs
    - Gain greatest compression with least cost
  1. Add resources to critical path tasks
  2. Limit or reduce requirements (scope)
  3. Changing the sequence of tasks
- Fast Tracking
  - Overlapping of phases, activities or tasks that would otherwise be sequential
  - Involves some risk
  - May cause rework

What Microsoft Project Cannot Do

- The machine cannot do four things:
  - it cannot create the **tasks** for you
  - it cannot create the **logic** relationships
  - it does not know the **duration** of each task
  - it cannot possibly know what **resources** you have to apply to the tasks
- Thus, the drawing up of a network chart showing the logic is an essential part of the paperwork and that must be done before putting information into the computer.
- By having the paperwork as complete as you can get it, the entry of data into the computer becomes a simple routine.

What Microsoft Project Can Do

- It does not do project management for you; neither does ownership of the product make you a project manager!
- However:
  - It can do the calculations accurately for you.
  - It makes visible the parameters it needs (have I forgotten something?) and any problems (like unacceptable circular logic).
  - It allows "what-ifs?" to make changes to the project and see the effect of those changes before finalizing your plan and committing it to work.

What Microsoft Project Can Do

- Once your plan is in action, it allows progress to be tracked so that you can make adjustments to keep on target.
- It is a tremendous aid to communication:
  - There are built-in reports to print.
  - You can export to PowerPoint for presentations, to Word for detailed reports, to Excel to do intricate cost analyses, and to Access for manipulation of project data.
  - You can send information by e-mail.
  - You can pass information over networks and the Internet.

The MS-Project Process

- Move WBS into a Project outline (in Task Sheet)
- Add resources (team members or roles)
- Add costs for resources
- Assign resources to tasks
- Establish dependencies
- Refine and optimize
- Create baseline
- Track progress (enter actuals, etc.)

Create Your Project

- File/New
- Setup start date
- Setup calendar
  - Menu: Project/Project Information
  - Often left with default settings
  - Hours, holidays

Enter WBS

- Outlining
- Sub-tasks and summary tasks
- Do not enter start/end dates for each
- Just start with Task Name and Duration for each
- Use Indent/Outdent buttons to define summary tasks and subtasks
- You can enter specific Start/End dates but don't most of the time

Establish Durations

- Know the abbreviations
  - h/d/w/m
  - d is default
- Can use partial
  - .5d is a half-day task
- Elapsed durations
- Estimated durations
  - Put a '?' after duration

Add Resources

- Work Resources
  - People
- Material Resources
  - Things
  - Can be used to track costs
    - Ex: amount of equipment purchased
  - Not used as often in typical software project

Resource Sheet

- Can add new resources here
  - Or directly in the task entry sheet
    - Beware of misspellings (Project will create near-duplicates)
- Setup costs
  - Such as annual salary (put 'yr' after 'Std. Rate')

Effort-Driven Scheduling

- MS-Project default
- Duration * Units = Work
  - Duration = Work / Units (D = W/U)
  - Work = Duration * Units (W = D*U)
  - Units = Work / Duration (U = W/D)
- Adding more resources to a task shortens duration
- Can be changed on a per-task basis
  - In the advanced tab of Task Information dialog box
  - Task Type setting
- Beware the Mythical Man-month
  - Good for laying bricks, not always so for software development

Link Tasks

- On toolbar: Link & Unlink buttons
  - Good for many at once
- Or via Gantt chart
  - Drag from one task to another

Milestones

- Zero duration tasks
- Insert task 'normally' but put 0 in duration

Make Assignments

- Approach 1. Using Task Sheet
  - Using Resource Names column
  - You can create new ones by just typing-in here
- 2. Using Assign Resources dialog box
  - Good for multiple resources
  - Highlight task, Tools/Resources or toolbar button
- 3. Using Task Information dialog
  - Resources tab
- 4. Task Entry view
  - View/More Views/Task Entry
  - Or Task Entry view on Resource Mgmt. toolbar

Save Baseline

- Saves all current information about your project
  - Dates, resource assignments, durations, costs

Fine Tune

- Then is used later as basis for comparing against "actuals"
- Menu: Tools/Tracking/Save Baseline

References

- Westland, Jason. 2003. **Project Management Guidebook**
- **Work breakdown structure.** http://en.wikipedia.org/wiki/Work_breakdown_structure
- **Course Q7503: Principles of Software Project Management,** lectures by, Instructor: John Musser jm2217@columbia.edu. See http://www.projectreference.com/
- **Microsoft Project: I – Introduction** by Mike Glen, MVP. http://pubs.logicalexpressions.com/Pub0009/LPMArticle.asp?ID=141
- **Leads, Lags and Floats.** http://www.tutorialspoint.com/management_concepts/leads_lags_floats.htm