Project Management

Project Nomenclature

- **Deliverables**: Units that are delivered
- **Activities**: Major work categories
- **Tasks**: Small units of work that make up activity

Activity: Major Unit of Work

Culminates in major project ***milestone***
(a scheduled event used to measure progress)

Examples of Activities

- Major Activities:
  - Planning
  - Requirements Elicitation
  - Requirements Analysis
  - System Design
  - Object Design
  - Implementation
  - System Testing
  - Delivery
- Sub-activities during requirements analysis:
  - Define scenarios
  - Define use-case model
  - Define object model
  - Define dynamic model
  - Design user interface

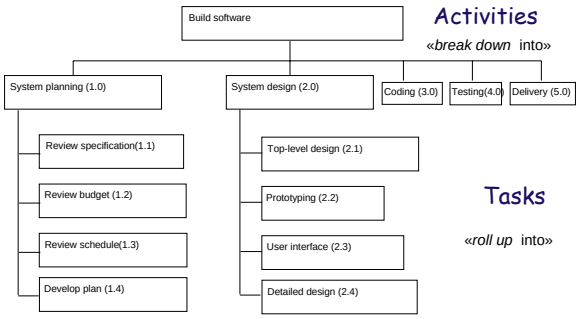
Big 3 Diagrams

- Work Breakdown Structure
- Precedence Diagram
- Schedule
  - with staff loading

Work Breakdown Structure (WBS)

- Break up project into **activities**, **sub-activities**, ... and **tasks**.
- The work breakdown structure does **not** attempt to show the interdependence or sequencing of the activities, only how they sub-divide and how much resources they are expected to require.

Work Breakdown Structure (WBS) Diagram



WBS Tradeoffs

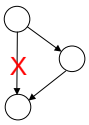
- Work breakdown structure allows **estimation of costs/time** required, by **"rolling up"** such quantities:

  Σ (children resources) = parent resource

- A WBS that is too coarse makes it difficult to assign tasks and utilize resources.

Precedence Diagram (PERT chart)

- List all the tasks.
- For each task:
  - List the tasks that must (immediately) precede that task
  - Remove *implied* dependencies (called the "**transitive reduction**" of the graph)

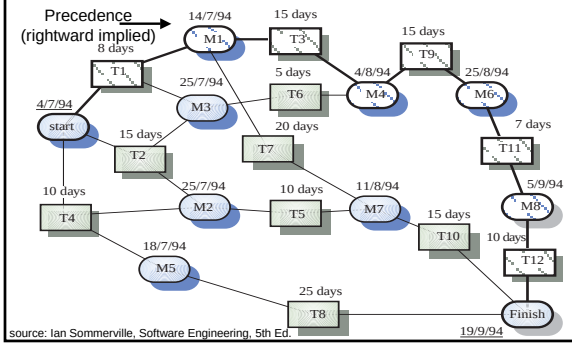


Task durations and dependencies

Task	Duration (days)	Dependencies
T1	8	
T2	15	
T3	15	T1
T4	10	
T5	10	T2, T4
T6	5	T1, T2
T7	20	T1
T8	25	T4
T9	15	T3, T6
T10	15	T5, T7
T11	7	T9
T12	10	T11

source: Ian Sommerville, Software Engineering, 5th Ed.

Precedence Chart with Milestones (ovals)



source: Ian Sommerville, Software Engineering, 5th Ed.

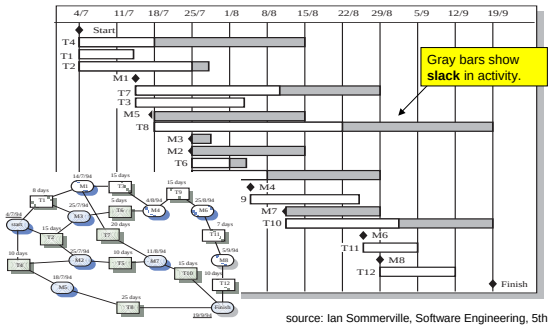
Critical Path

- A "critical path" on a Precedence chart is one in which the **sum of durations** of the tasks on the path equals the **shortest** overall project-completion time.
- The critical path is shaded on the preceding diagram.
- If your project's critical path is too long, what can you do to reduce it?

Gantt chart (schedule)

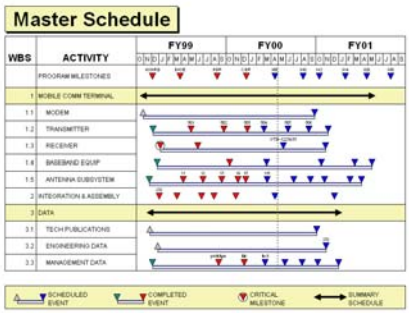
- **Schedule** (or Gantt chart) shows a **particular scheduling** of the tasks to time lines, subject to **all** of the constraints (not just precedence)

Gantt from Precedence (inset)



source: Ian Sommerville, Software Engineering, 5th Ed.

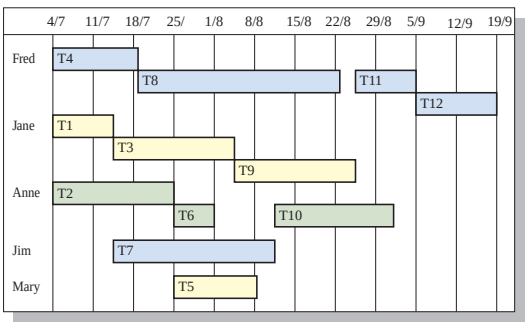
Example of Schedule Symbology (not standardized, so give a legend)



Staff Loading & Resources

- Staff loading and resource constraints are two aspects of scheduling not represented directly on Gantt chart.
- Generally they have the effect of providing **added sequencing due to resource contention**, and therefore lengthening overall project time

One Way to Show Staff Loading

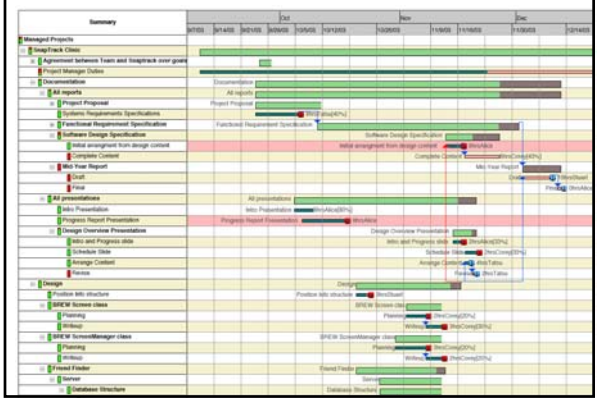


source: Ian Sommerville, Software Engineering, 5th Ed.

3 Diagrams in 1

- If you choose the right tool, you can avoid duplicating work in creating diagrams.
- Some tools, such as Microsoft Project, allow you to show the work breakdown and precedence on directly on the schedule.
- You enter the information just once.

Portion of Schedule from a 2003 Project



Some Tools

- Microsoft Project
  - Handles WBS
  - Handles PERT to Gantt
  - Sort of handles staff loading
  - Does earned value tracking (to be described)
- Planner (http://www.imendio.com/projects/planner/) open source tool, for the Gnome Linux & Unix desktop
- TopDown (Mac) or Visio (PC) can be used for drawing any kind of diagram.
- PowerPoint is often useful for diagrams.
- **Post pdf's of diagrams to your Wiki.**

Tracking How Well We Are Doing

Troubles Shown by Gantt Chart

- Mostly long bars, and few of them
  - Difficult to track project status, since the longer the bar, the more likely there is error in the estimate
  - Solution: Break long bars into to smaller ones and reassign

Troubles Shown by Gantt Chart

- Long bars not **in parallel** with each other:
  - Is the **whole team** really doing that one task?
  - Or are a couple doing it while the others are idle?
  - Is it really not possible to describe in any finer detail?

Troubles Shown by Gantt Chart

- Long chains of short bars with one person assigned to each bar
  - Not enough parallelism among tasks
  - Solution: Reduce dependence between bars as much as possible.

Troubles Shown by Gantt Chart

- No specific staff assignment to bars
  - Difficult to know who on the team is actually doing the work
  - How do you know who to ask for status?

Think Carefully about Staff Assignments

- Idle staff now may mean that everyone pays later to make up for the lost resource.