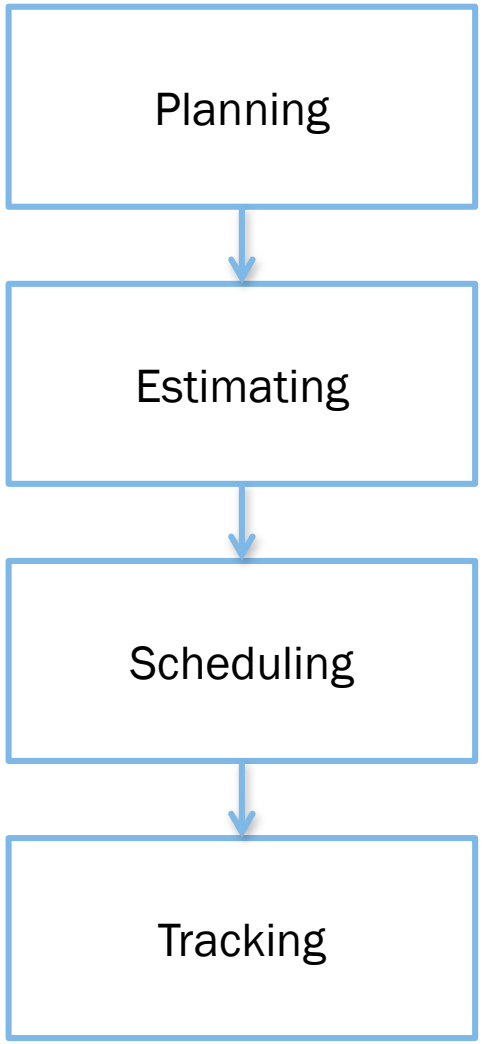
Project Scheduling and Tracking

SWEN 256 – Software Process & Project Management

Planning, Estimating, Scheduling, Tracking

- Plan: Identify activities. No specific start and end dates.
- Estimating: Determining the size & duration of activities.
- Schedule: Adds specific start and end dates, relationships, and resources.
- Track: Uses monitoring and tools to determine if plans, estimates, and schedules are accurate



How To Schedule

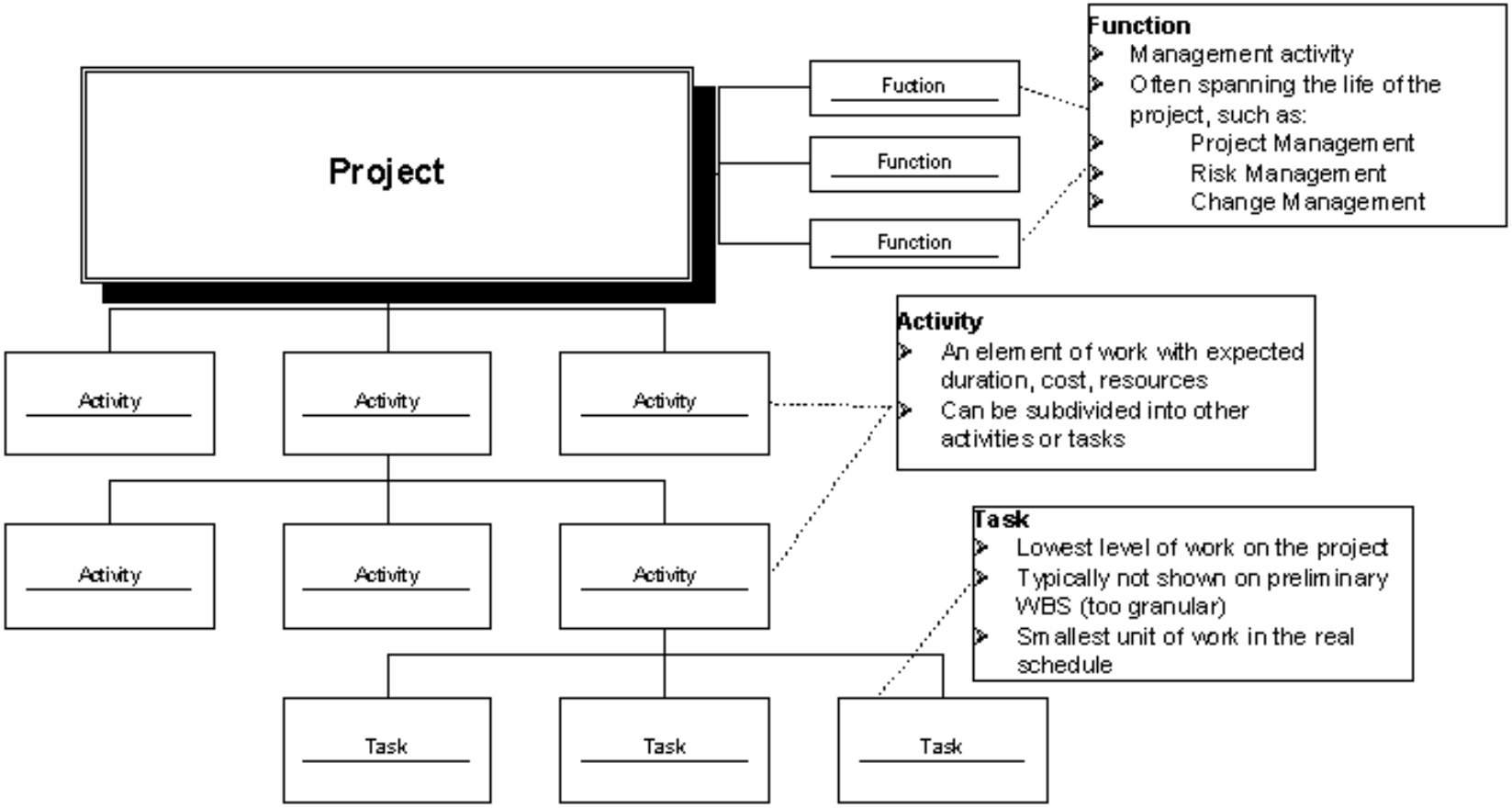
- 1. Identify “what” needs to be done
  - Work Breakdown Structure (WBS)
- 2. Identify “how much” (the size)
  - Size estimation techniques
- 3. Identify the dependency between tasks
  - Dependency graph, network diagram
- 4. Estimate total duration of the work to be done
  - The actual schedule

Partitioning Your Project

- You need to decompose your project into manageable chunks
- ALL projects need this step
- Divide & Conquer
- Two main causes of project failure
  - Forgetting something critical
  - Ballpark estimates become targets
- How does partitioning help this?

Project Elements

- A Project: functions, activities, tasks



Work Breakdown Structures

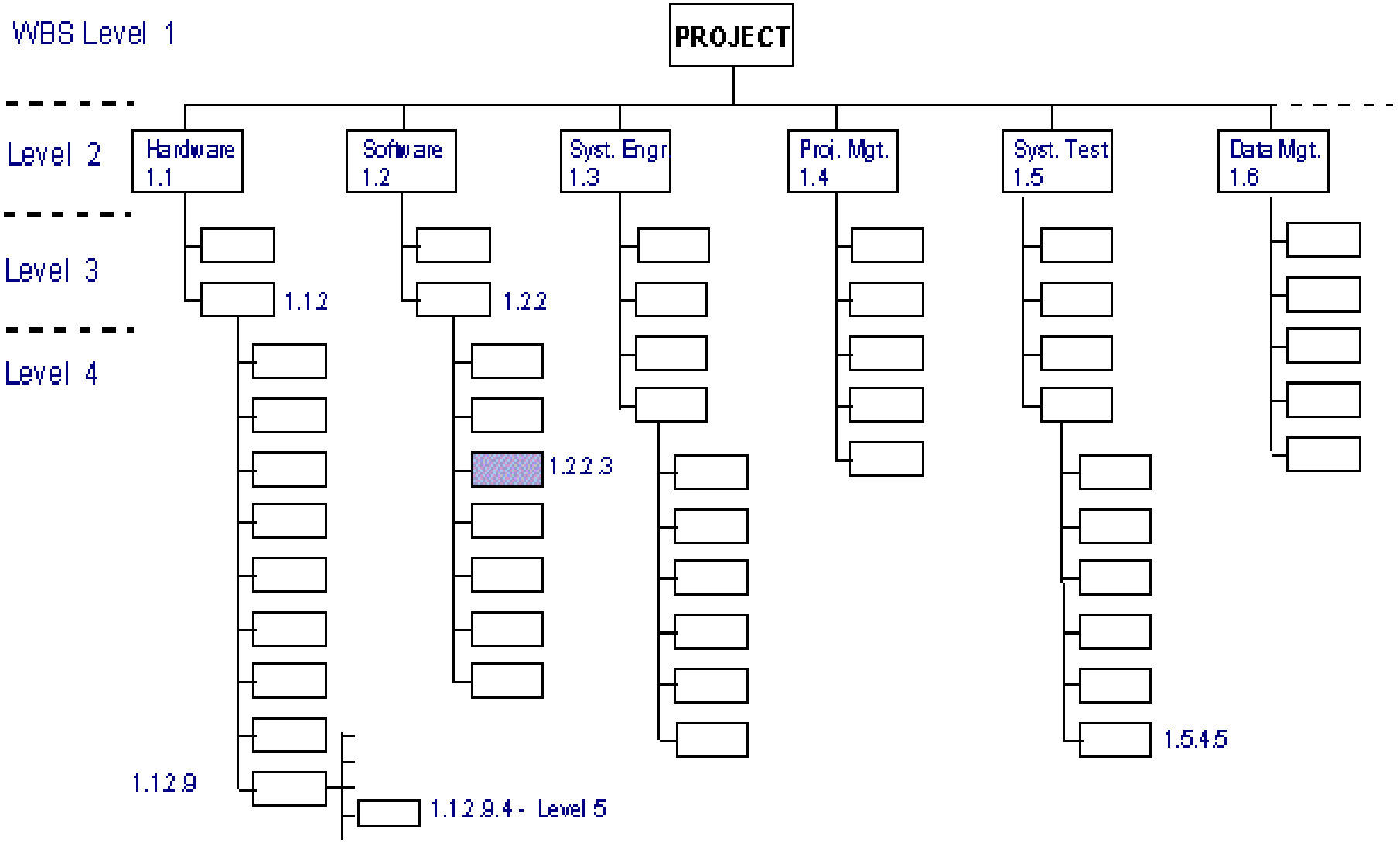
a.k.a. WBS

Work Breakdown Structure

- Work Break Down Structure (WBS): a check list of the work that must be accomplished to meet the project objectives.
- The WBS lists the major project outputs and those departments or individuals primarily responsible for their completion.

WBS Outline Example

0.0 Retail Web Site
1.0 Project Management
2.0 Requirements Gathering
3.0 Analysis & Design
4.0 Site Software Development
    4.1 HTML Design and Creation
    4.2 Backend Software
        4.2.1 Database Implementation
        4.2.2 Middleware Development
        4.2.3 Security Subsystems
        4.2.4 Catalog Engine
        4.2.5 Transaction Processing
    4.3 Graphics and Interface
    4.4 Content Creation
5.0 Testing and Production

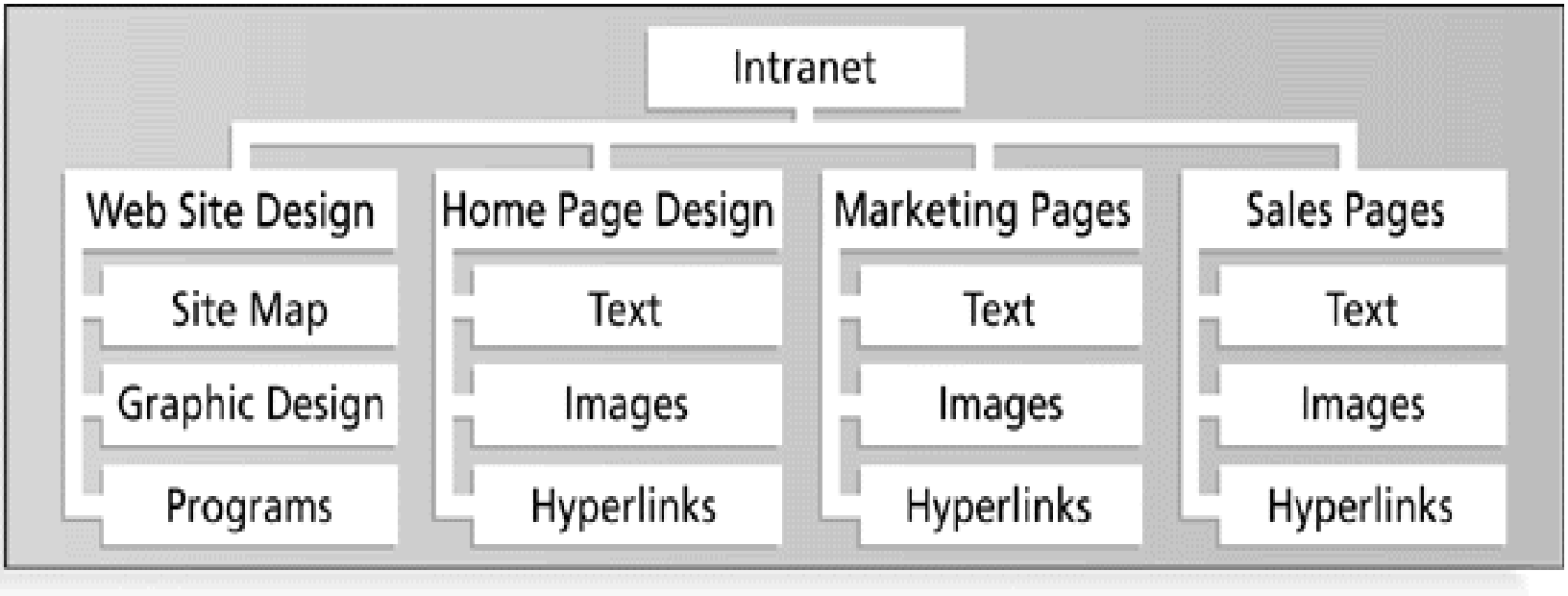


From: http://www.hyperthot.com/pm_wbs.htm

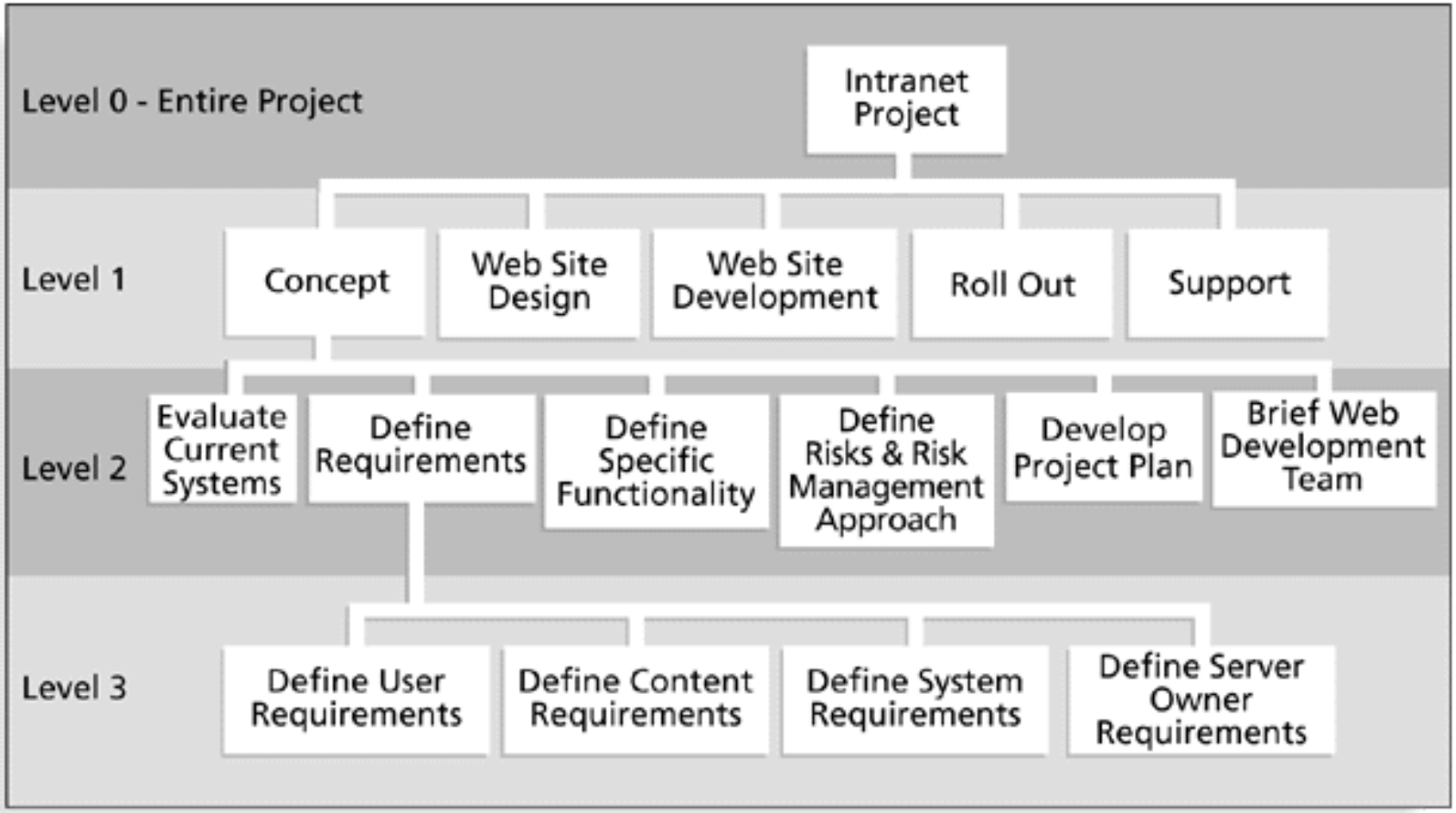
WBS Types

- Process WBS
  - a.k.a Activity-oriented
  - Ex: Requirements, Analysis, Design, Testing
  - Typically used by PM
- Product WBS
  - a.k.a. Entity-oriented
  - Ex: Financial engine, Interface system, DB
  - Typically used by engineering manager
- Hybrid WBS: both above
  - This is not unusual
  - Ex: Lifecycle phases at high level with component or feature-specifics within phases
  - Rationale: processes produce products

Product WBS


Intranet
Web Site Design
Site Map
Graphic Design
Programs
Home Page Design
Text
Images
Hyperlinks
Marketing Pages
Text
Images
Hyperlinks
Sales Pages
Text
Images
Hyperlinks

Process WBS


Level 0 - Entire Project
Intranet Project
Level 1
Concept
Web Site Design
Web Site Development
Roll Out
Support
Level 2
Evaluate Current Systems
Define Requirements
Define Specific Functionality
Define Risks & Risk Management Approach
Develop Project Plan
Brief Web Development Team
Level 3
Define User Requirements
Define Content Requirements
Define System Requirements
Define Server Owner Requirements

Process WBS

- List of Activities, not Things
- List of items can come from many sources
  - SOW, Proposal, brainstorming, stakeholders, team
- Describe activities using “bullet language”
  - Meaningful but terse labels
- All WBS paths do not have to go to the same level
- Do not plan more detail than you can manage

Work Packages (Tasks)

- Generic term for discrete **tasks** with definable end results
- The “one-to-two” rule
  - Often at: 1 or 2 persons for 1 or 2 weeks
- Basis for monitoring and reporting progress
  - Can be tied to budget items (charge numbers)
  - Resources (personnel) assigned
- Ideally shorter rather than longer
  - Longer makes in-progress estimates needed
  - These are more subjective than “done”
  - “4/40” or “8/80” rule (shortest/longest duration)
  - Not so small as to micro-manage

WBS & Methodology

- PM must map activities to chosen lifecycle
- Each lifecycle has different sets of activities
- Integral process activities occur for all
  - Planning, configuration, testing
- Operations and maintenance phases are not normally in plan (considered post-project)
- Some models are “straightened” for WBS
  - Spiral and other iterative models
  - Linear sequence several times
- Deliverables of tasks vary by methodology

WBS Techniques

- Top-Down
- Bottom-Up
- Analogy
- Rolling Wave
  - 1st pass: go 1-3 levels deep
  - Gather more requirements or data
  - Add more detail later
- Post-its on a wall

All WBS Techniques rely upon **Expert Judgment!**

WBS Techniques

- Top-down
  - Start at highest level
  - Systematically develop increasing level of detail
  - Best if
    - The problem is well understood
    - Technology and methodology are not new
    - This is similar to an earlier project or problem
  - But is also applied in majority of situations

WBS Techniques

- Bottom-up
  - Start at lowest level tasks
  - Aggregate into summaries and higher levels
  - Cons
    - Time consuming
    - Needs more requirements complete
  - Pros
    - Detailed

WBS Techniques

- Analogy
  - Base WBS upon that of a “similar” project
  - Use a template
  - Analogy also can be estimation basis
  - Pros
    - Based on past actual experience
  - Cons
    - Needs comparable project

WBS Techniques

- Brainstorming
  - Generate all activities you can think of that need to be done
  - Group them into categories
- Both Top-down and Brainstorming can be used on the same WBS
- Remember to get the people who will be doing the work involved (buy-in matters!)

WBS – Basis of Many Things

- Network scheduling
- Costing
- Risk analysis
- Organizational structure
- Control
- Measurement

WBS Guidelines

- Should be easy to understand
- Some companies have corporate standards for these schemes
- Some top-level items, like Project Mgmt. are in WBS for each project
  - Others vary by project
- What often hurts most is what's missing
- Break down until you can generate accurate time & cost estimates
- Ensure each element corresponds to a deliverable

Sequencing Tasks & Activities

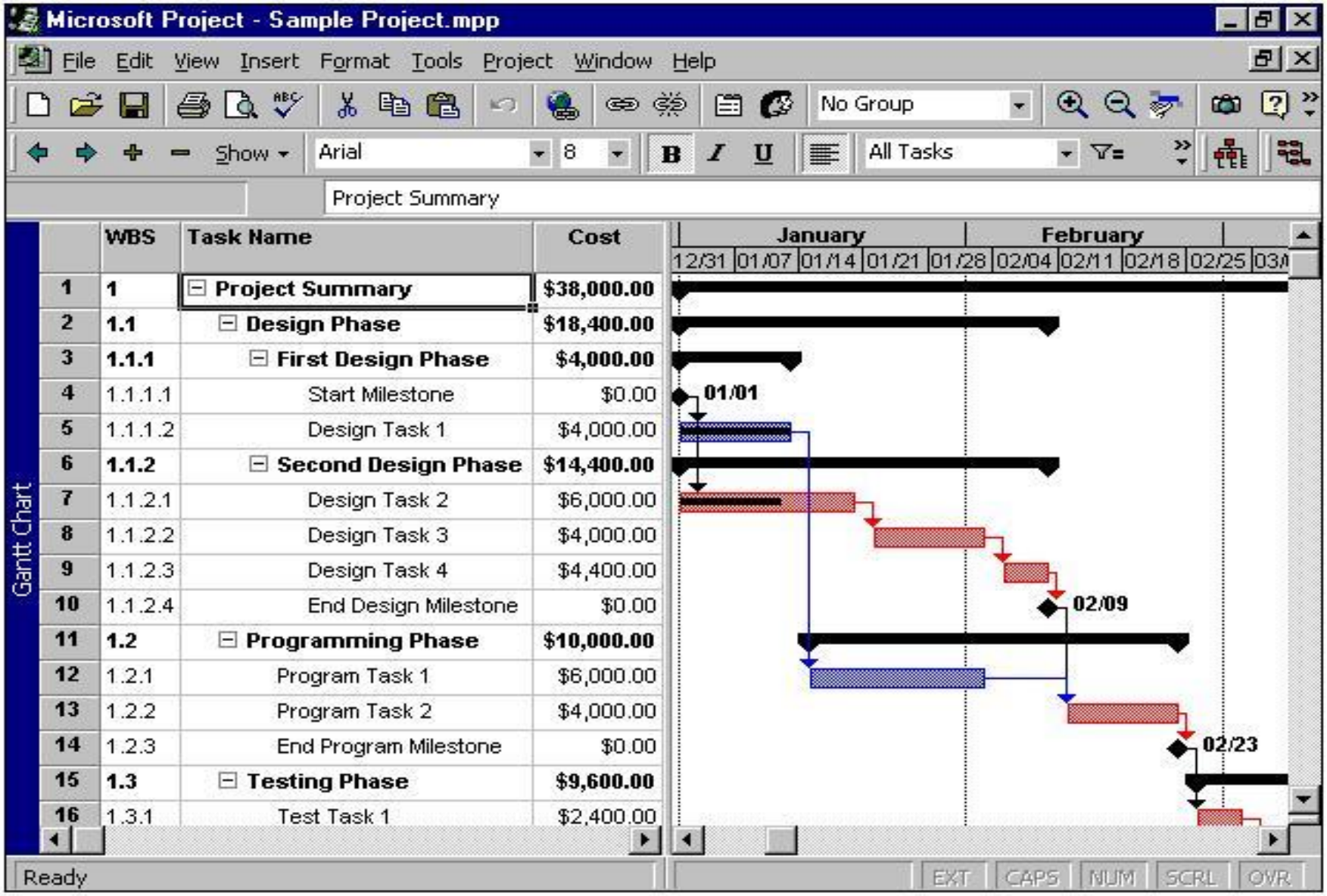
Tools and Techniques

Sequence the Work Activities

- Milestone Chart
- Gantt chart
- Network Techniques
  - CPM (Critical Path Method)
  - PERT (Program Evaluation and Review Technique)

Gantt Chart

- Gantt chart is a means of displaying simple activities or events plotted against time or dollars
- Most commonly used for exhibiting program progress or for defining specific work required to reach an objective
- Gantt charts may include listing of activities, activity duration, scheduled dates, and progress-to-date


Microsoft Project - Sample Project.mpp
File Edit View Insert Format Tools Project Window Help
No Group
Show
Arial
8
All Tasks
Project Summary
Gantt Chart
WBS
Task Name
Cost
1 1 Project Summary $38,000.00
2 1.1 Design Phase $18,400.00
3 1.1.1 First Design Phase $4,000.00
4 1.1.1.1 Start Milestone $0.00
5 1.1.1.2 Design Task 1 $4,000.00
6 1.1.2 Second Design Phase $14,400.00
7 1.1.2.1 Design Task 2 $6,000.00
8 1.1.2.2 Design Task 3 $4,000.00
9 1.1.2.3 Design Task 4 $4,400.00
10 1.1.2.4 End Design Milestone $0.00
11 1.2 Programming Phase $10,000.00
12 1.2.1 Program Task 1 $6,000.00
13 1.2.2 Program Task 2 $4,000.00
14 1.2.3 End Program Milestone $0.00
15 1.3 Testing Phase $9,600.00
16 1.3.1 Test Task 1 $2,400.00
January
February
12/31 01/07 01/14 01/21 01/28 02/04 02/11 02/18 02/25
01/01
02/09
02/23
Ready
EXT CAPS NUM SCRL OVR

Gantt Chart

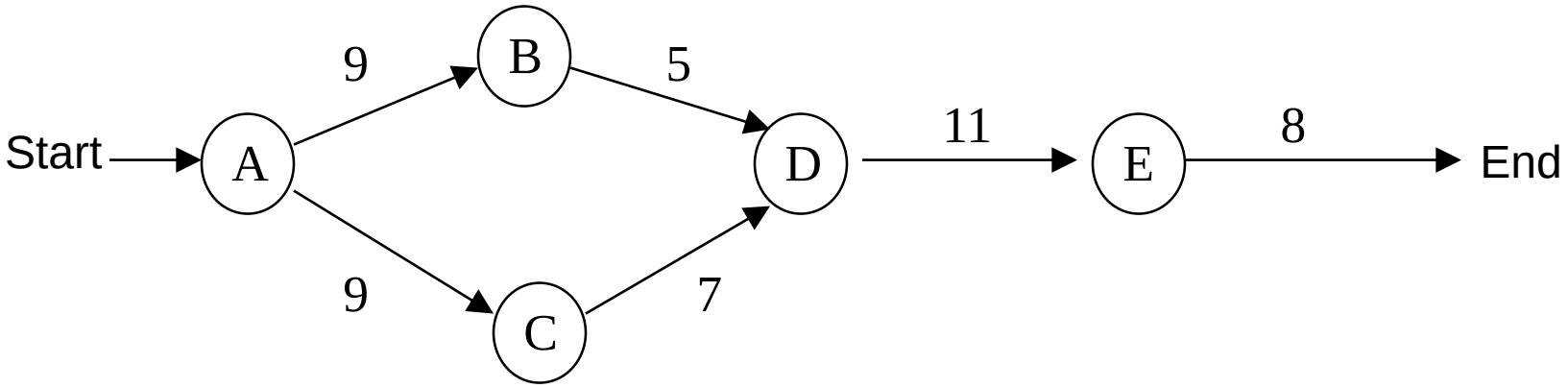
- Advantages:
  - Easy to understand
  - Easy to change
- Disadvantages:
  - Only a vague description of the project
  - Does not always show interdependency of activities
  - May not show results of an early or late start of an activity

Network Techniques

- A precedence network diagram is a graphic model portraying the sequential relationship between key events in a project.
- Initial development of the network requires that the project be defined and thought out.
- The network diagram clearly and precisely communicates the plan of action to the project team and the client.

Precedence Network Diagram

Task	Duration	Dependencies
A - Architecture & design strategy	9	start
B - Decide on number of releases	5	A
C - Develop acceptance test plan	7	A
D - Develop customer support plan	11	B,C
E - Final sizing & costing	8	D



Critical Path Method

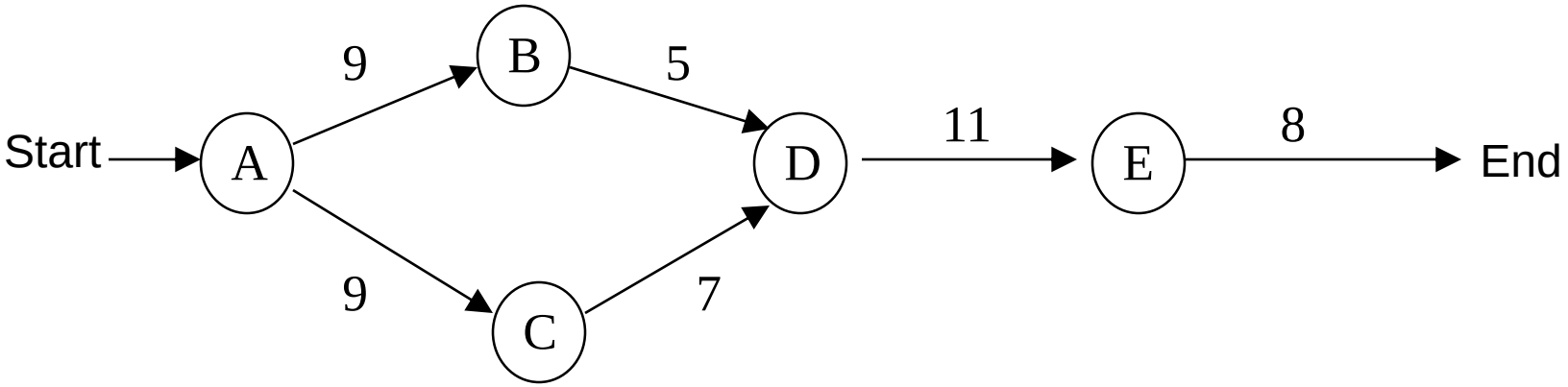
Critical Path Method (CPM) tries to answer the following questions:

- What is the duration of the project?
- By how much (if at all) will the project be delayed if any one of the activities takes N days longer?
- How long can certain activities be postponed without increasing the total project duration?

Critical Path

- Sequence of activities that have to be executed one after another
- Duration times of these activities will determine the overall project time, because there is no slack/float time for these activities
- If any of the activities on the critical path takes longer than projected, the entire project will be delayed by that same amount
- Critical path = Longest path in the precedence network (generally, the longest in time)

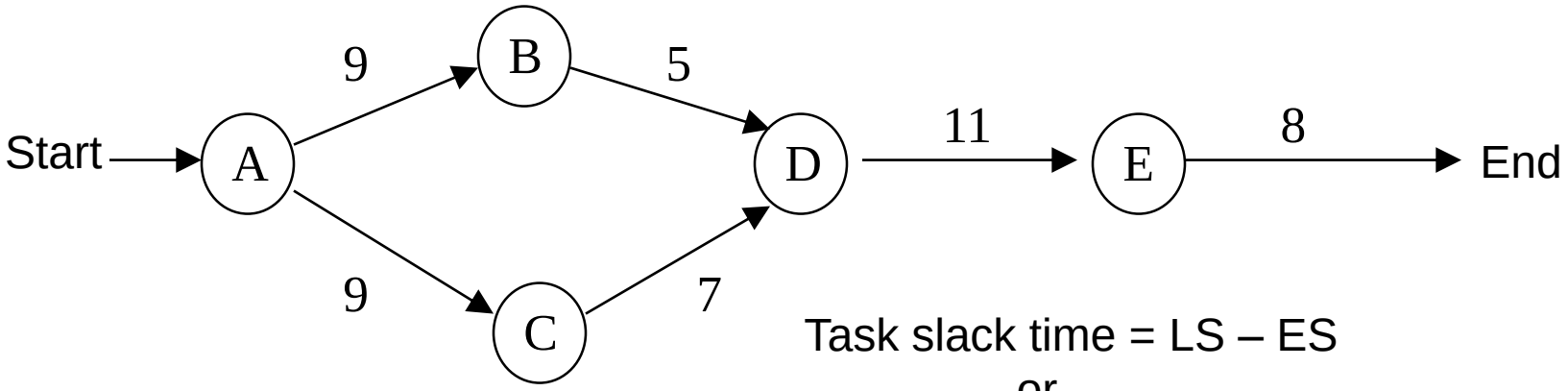
Critical Path Example



- Critical Path = A – C – D – E (35 time units)
- Critical Tasks = A,C,D,E
- Non-Critical Path = A-B-D-E
- Non-Critical Tasks = B (only)

Critical Path Example (Continued)

Task	Duration	Depend	Earliest Start	Earliest Finish	Latest Start	Latest Finish
A	9	none	0	9	0	9
B	5	A	9	**14**	**11**	16
C	7	A	9	16	9	16
D	11	B,C	16	27	16	27
E	8	D	27	35	27	35



Slack time – maximum allowable delay for a non-critical activity.

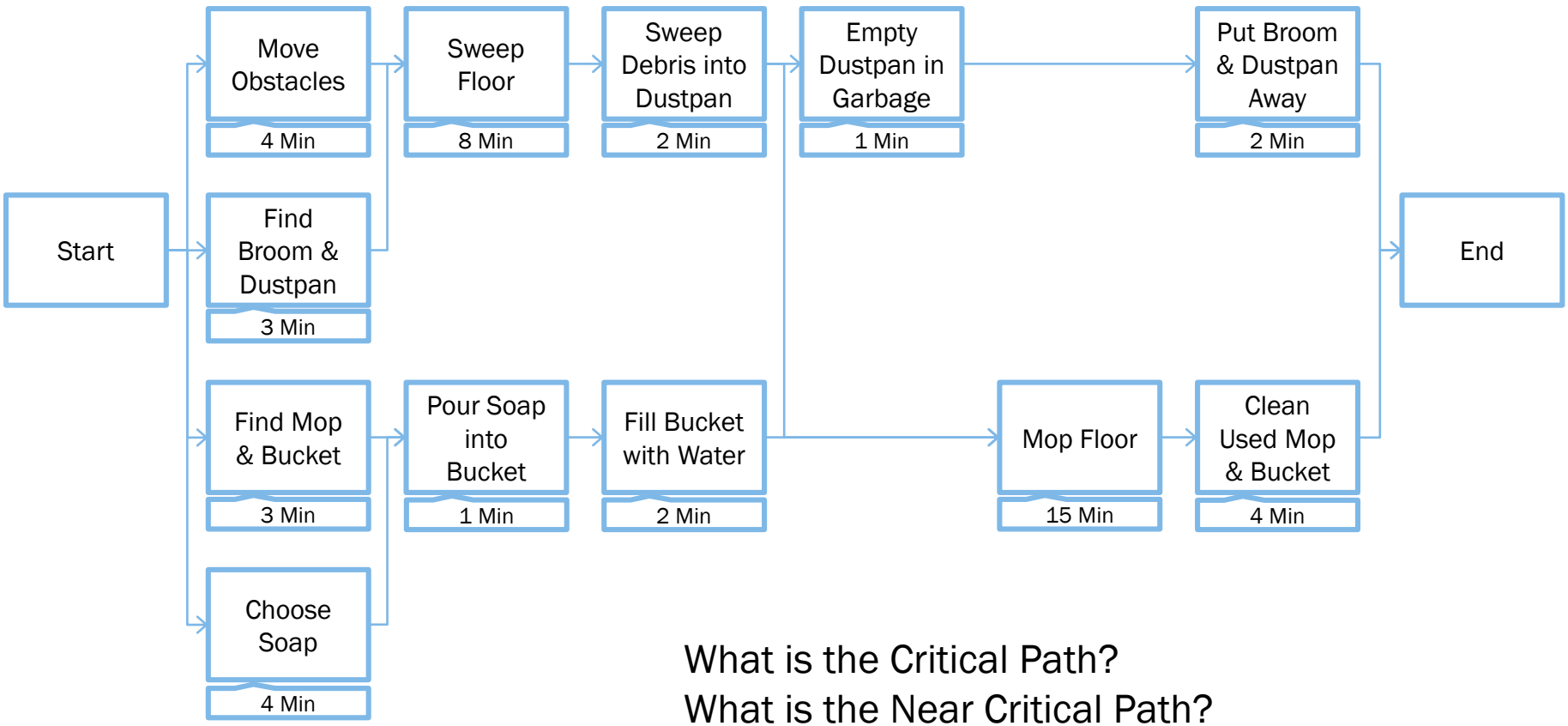
Task slack time = LS – ES

- or -

Task slack time = LF - EF

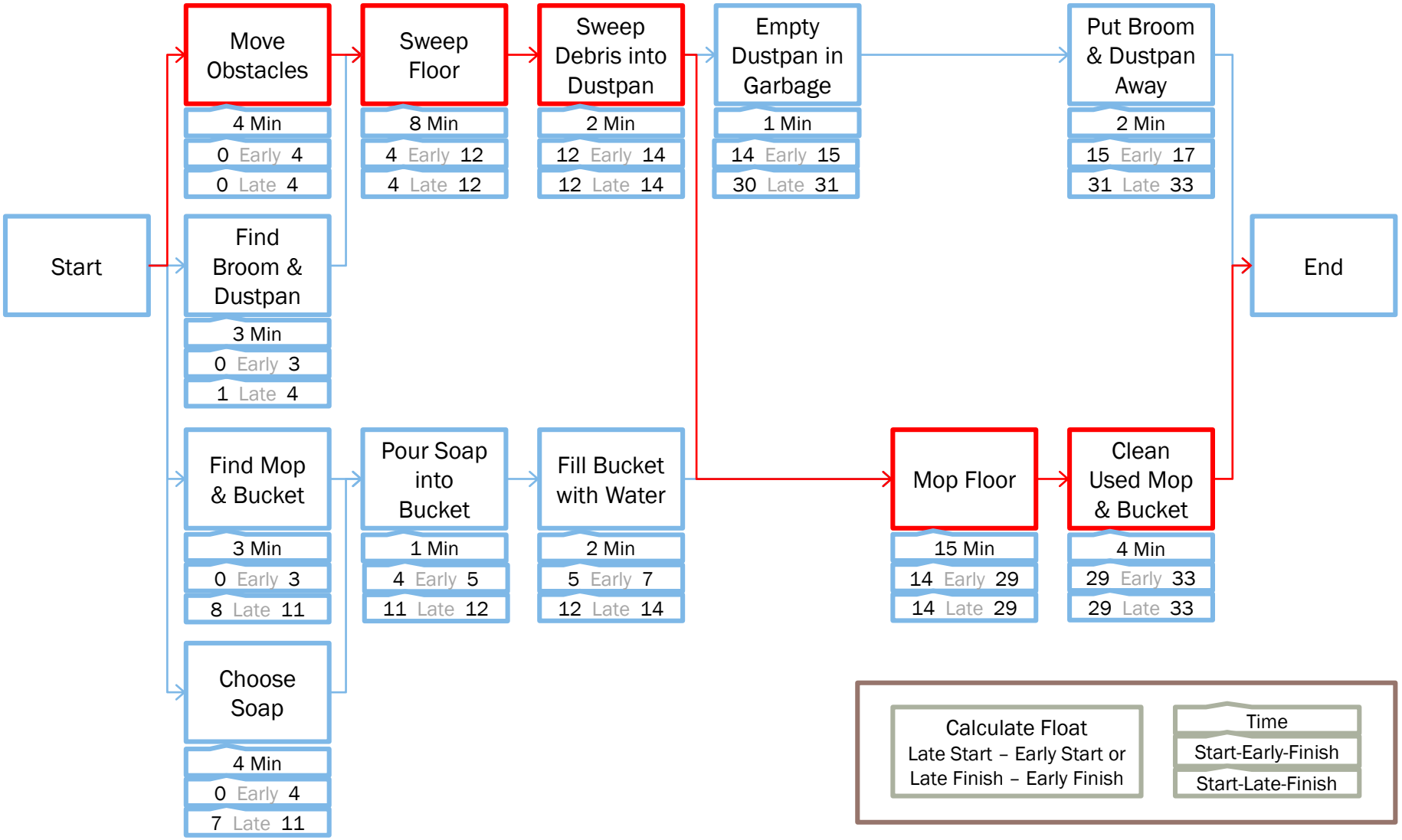
Task B has 2 time units of ***slack time***

Critical Path Example



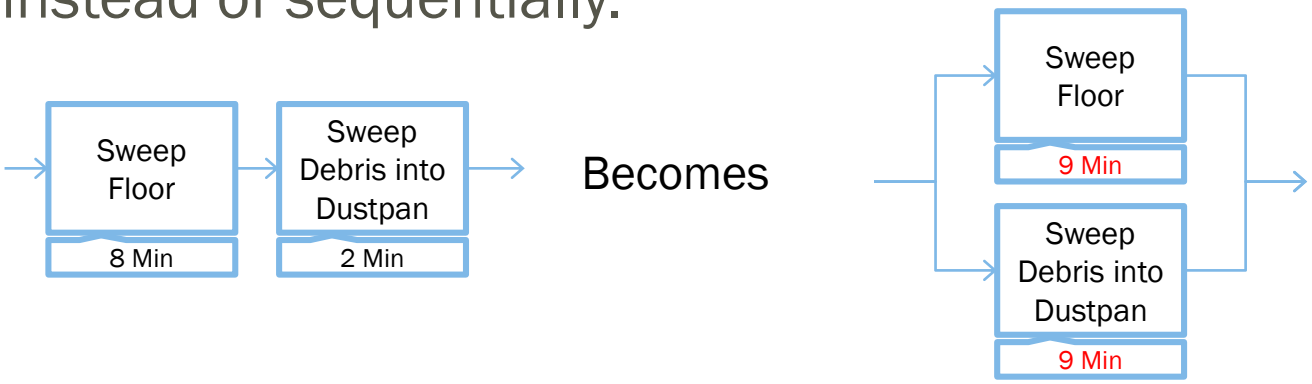
What is the Critical Path?
What is the Near Critical Path?
What is the Float/Slack of “Choose Soap”?
What is the float of “Mop Floor”?

Critical Path



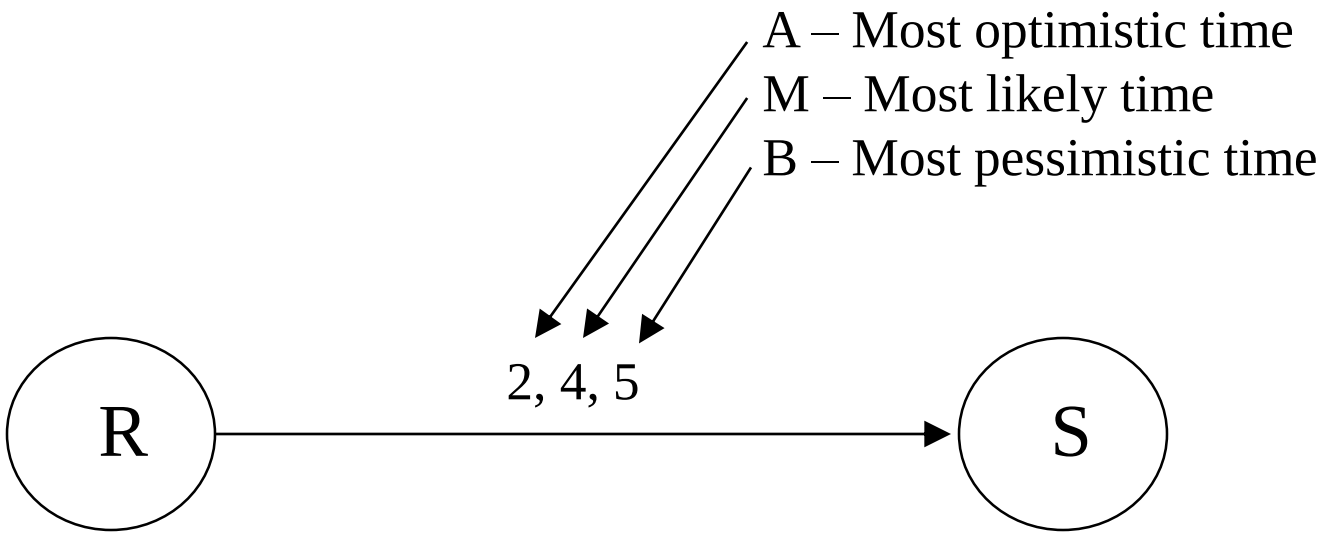
Schedule Compression

- Fast Tracking – Work critical path activities in parallel, instead of sequentially.



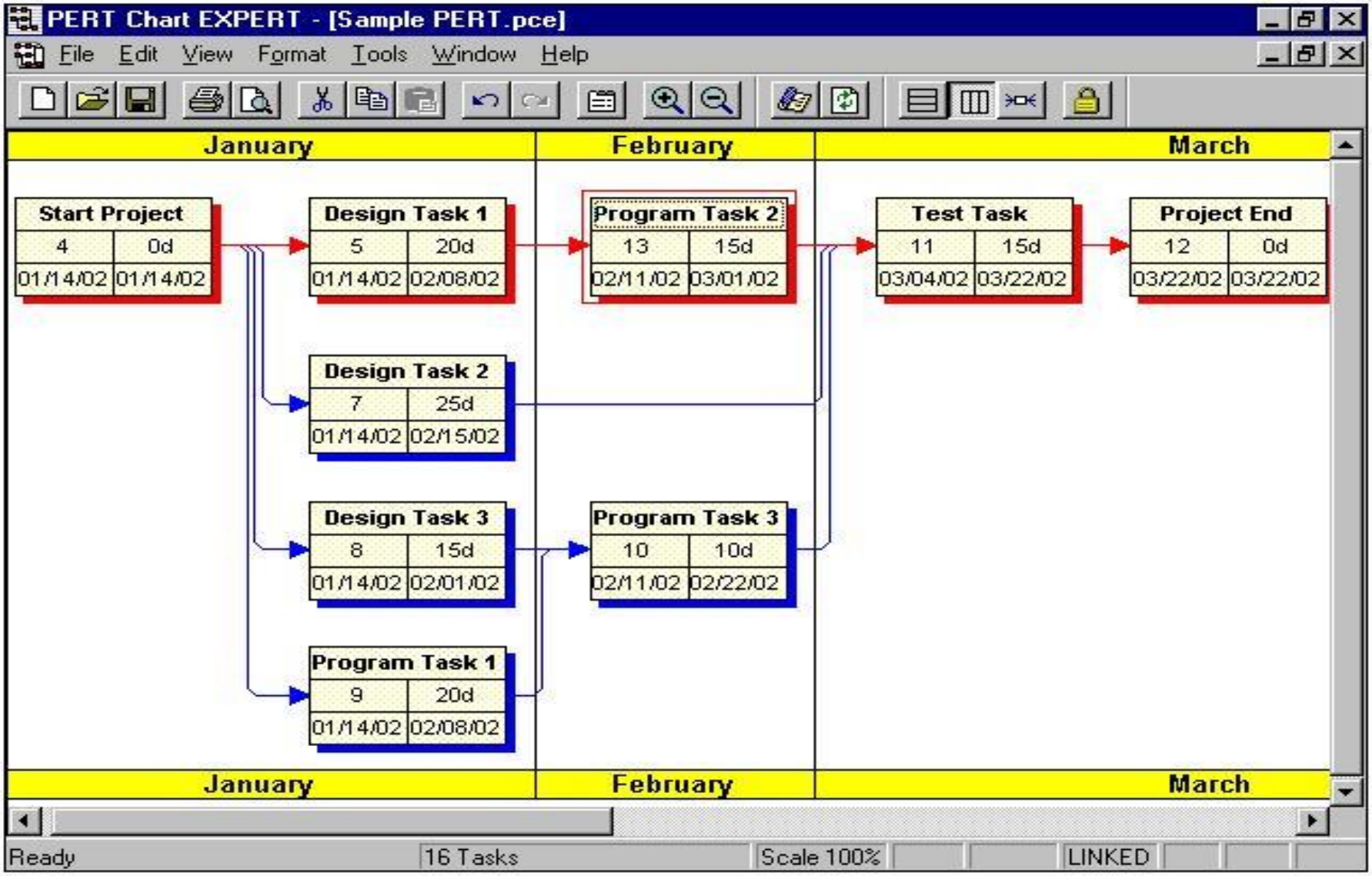
- Crashing – Cost and schedule tradeoffs (Time ≈ Money) with the end result of reducing overall time to completion
- Other Ways (Less preferred but sometimes needed)
  - Reduce Scope/Quality of product
  - Increase resources, Reduce risks
  - Say no: Sometimes schedule compression just isn't an option

PERT



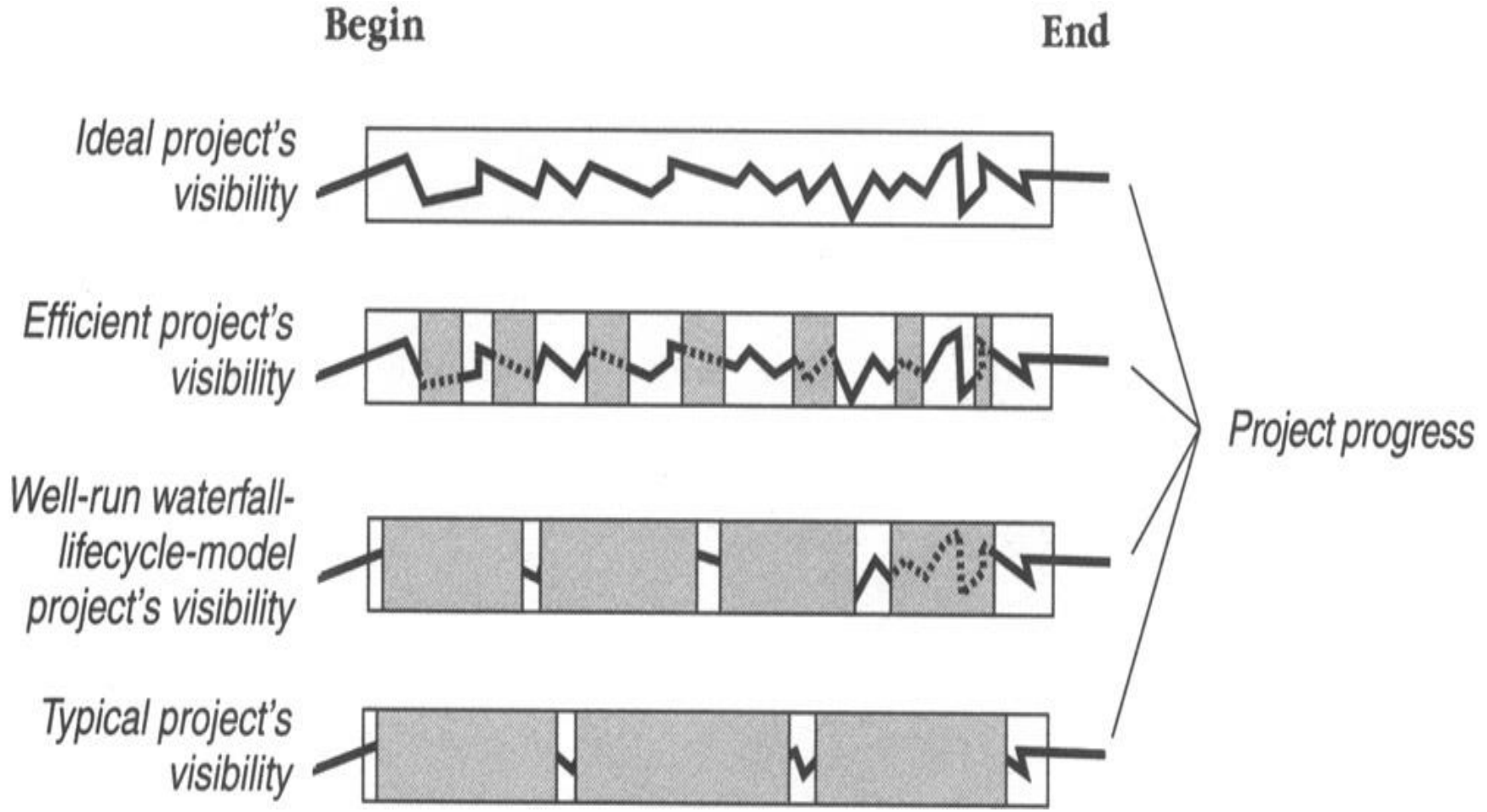
Expected Time = (a + 4m + b)/6

Expected Time = 3.8


PERT Chart EXPERT - [Sample PERT.pce]
File Edit View Format Tools Window Help
January
February
March
Start Project
4 0d
01/14/02 01/14/02
Design Task 1
5 20d
01/14/02 02/08/02
Design Task 2
7 25d
01/14/02 02/15/02
Design Task 3
8 15d
01/14/02 02/01/02
Program Task 1
9 20d
01/14/02 02/08/02
Program Task 2
13 15d
02/11/02 03/01/02
Program Task 3
10 10d
02/11/02 02/22/02
Test Task
11 15d
03/04/02 03/22/02
Project End
12 0d
03/22/02 03/22/02
January
February
March
Ready
16 Tasks
Scale 100%
LINKED

Project Tracking

Tracking and Visibility



Percent Complete

Task	Complete?
Conceptual Design	Complete
Program Specification	Complete
Coding	In Progress
Documentation	In Progress
User Manual Production	Not Started
Testing	Not Started

Percent Complete

Task	How Complete?
Conceptual Design	200/200
Program Specification	300/300
Coding	150/600
Documentation	10/100
User Manual Production	0/400
Testing	0/500

660 / 2100 * 100 = 31.4% complete

Earned Value

- Earned Value (EV) is a methodology used to control a project
- It provides a uniform measure for project progress for the entire project or any sub-element
- Provides a consistent method of project progress and performance
- Provides a basis for cost performance analysis of a project

To Use Earned Value Tracking

1. Establish a WBS to divide the project into manageable parts
2. Identify the activities required for the current project
3. Allocate the effort required for each activity
4. Schedule the activities over time and resources
5. Analyze/review the schedule

(continued)

To Use Earned Value Tracking

6. Update the schedule by reporting activity progress
7. Enter the actual cost on the activities
8. Execute the Earned Value calculations
9. Analyze the data and make course corrections as necessary

Earned Value

- Establish a common value scale for every task, regardless of the type of work involved (software projects use effort)
- Total effort for the entire project is estimated
- Every task is given a planned value based on its estimated percentage of the total project effort
- Completion of a task results in a credit, or an earned value, of the value allocated to the task

Earned Value Example

Total Project Effort: 1000 person hours

Task A Estimate: 15 person hours

Planned Value: 1.5

Completing task A contributes 1.5 to the cumulative earned value total for the project

Earned Value Tracking

- Earned value credit is given only when the task is 100% complete
- Partially completed tasks are NOT given partial credit (in most software projects)
- Large tasks can/must be broken into subtasks
- Size tasks up to 80 person-hours; aim for 2 to 4 task completions per week

Summary

- Planning, Estimating, Scheduling, and Tracking are a continuum
- Projects need to be partitioned for manageability
  - Work Breakdown Structures are a great way to do this
- Sequencing Tasks & Activities is vital
  - Gantt Charts allow quick reference
  - Network Techniques such as Precedence Network Diagrams, the Critical Path Method, and PERT Charts are useful tools
- Project Tracking is important for project visibility
  - The Earned Value Technique is a key tool in this

Questions/Discussion