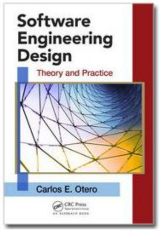
# CHAPTER 10: MANAGEMENT, LEADERSHIP, & ETHICS

-- Dr. L. Daniel Otero, Florida Institute of Technology --

## SESSION II: PLANNING, IMPLEMENTATION, AND TERMINATION

*Software Engineering Design: Theory and Practice*
by Carlos E. Otero



# SESSION'S AGENDA

- The Planning Stage
  - Scoping
    - Work breakdown structure
    - Budgeting
  - Organizing
    - Linear responsibility chart
    - Scheduling
  - Establish change control policy
- Implementation Stage
  - Earned value management
- Termination Stage
- What's next…

# THE PLANNING STAGE
# - SCOPING -

- The planning stage is critical to the success of any major project.
  - ✓ Planning exists to lay out a strong foundation for a successful project.
  - ✓ Planning establishes the directions to follow to complete the project and improve its probability of success.

- The key functions in the project planning stage are:
  - ✓ *Scoping*
  - ✓ *Organizing*

- Project scoping is the first key function of the planning stage.
  - ✓ It involves two main activities:
    - Identify the tasks
    - Develop the budget

- Two commonly used techniques for project scoping are:
  - ✓ The work breakdown structure (WBS)
  - ✓ Budgeting

# THE PLANNING STAGE
# - SCOPING: WORK BREAKDOWN STRUCTURE -

- The work breakdown structure (WBS) is a technique to represent, either graphically or in list format, a project modularized into task activities.
  - ✓ From the project management point of view, a project begins as a statement of work (SOW), which mainly consists as the set of main objectives to be achieved.
  - ✓ The SOW is decomposed into *tasks* where these tasks are decomposed into *subtasks*, and subtasks into *work packages*.
  - ✓ WBS represents these tasks.

- WBS is a simple but powerful technique that helps to plan, clearly define, and organize the activities related to reach specific milestones and complete a project.
  - ✓ Milestones are defined as specific events to be reached at specific points in time.

- For example, completion of the major design activities, e.g., architectural design, detailed design, and documentation can constitute the main project milestones of a design project.
  - ✓ Note that HCI design can be carried out as part of architectural or detailed design.
  - ✓ Construction design typically occurs during construction.

# THE PLANNING STAGE
# - SCOPING: WORK BREAKDOWN STRUCTURE -

- A very simple example of a WBS for the design of a software application

| Outline # | Task Name |
|---|---|
| 1 | Design Project |
| 1.1 | Architectural Design |
| 1.1.1 | Evaluate Alternative Designs |
| 1.1.1.1 | Prioritize Objectives |
| 1.1.1.2 | Formal Review of Evaluations |
| 1.1.2 | Select Architectural Design |
| 1.1.2.1 | Make Initial Selection |
| 1.1.2.2 | Formal Review to Select Final Design |
| 1.2 | Detailed Design |
| 1.2.1 | Evaluate Alternatives |
| 1.2.2 | Select Among Alternatives |
| 1.3 | Documentation |
| 1.3.1 | Write Architectural Design Documentation |
| 1.3.2 | Formal Review (Architecture Design) |
| 1.3.3 | Final Architectural Design Document |
| 1.3.4 | Write Detailed Design Documentation |
| 1.3.5 | Formal Review (Detailed Design) |
| 1.3.6 | Final Architectural Design Document |

# The Planning Stage
## - Scoping: Work Breakdown Structure -

- Evaluating the usefulness of a WBS is often more a matter of subjective assessments than not.
  - We can specify a criteria for this evaluation.
    - Scope accuracy
    - Completeness
    - Level of detail

- Scope accuracy
  - We need to agree that the WBS serves its overall objective of being a breakdown of the project into tasks, and nothing else.
  - A WBS does not contain information regarding tasks' relationships such as task precedence or task durations, and it does not include personnel skills required for the completion of tasks.
  - Instead, each element of a WBS represents an activity that consumes resources' time and effort.

- Completeness
  - Ensure that the WBS includes the complete set of tasks necessary to complete the project.

- Level of detail
  - Evaluates if the WBS breaks down the project's tasks into appropriate levels such that it facilitates the estimation of effort for each task.

# The Planning Stage
## - Scoping: Budgeting -

- Budgeting is the process of estimating the cost of a project.
  - It is basically a forecasting problem, since cost is a function of various parameters that are uncertain.

- Conceptually, a common strategy to develop a budget is to simply cost each element in the WBS.
  - To accomplish this, elements are associated with *direct* and *indirect costs*.

- Direct costs are those that can be directly tied to the development of the design.
  - The most often used direct costs are labor and equipment (i.e., hardware/software equipment).

- Indirect costs, on the other hand, include costs such as fringe benefits and administrative expenses.

## THE PLANNING STAGE
## - SCOPING: BUDGETING -

- One of the most significant and hard to estimate direct costs for each task is the cost of the staff that will be directly working on the completion of the task.
  - Given that software projects are typically custom-made solutions to particular problems, tasks are considered non-repetitive and often involve *learning rates*.
    - This is particularly true for tasks that involve new technologies or the use of skills that the staff is unfamiliar with.
    - Therefore, it is important to account for learning rates in the budgeting process.

- It is important to understand that developing accurate budgets for software projects is often very difficult due to the complexity of the projects.
  - Data from previous similar projects, as well as input from experienced personnel, are used as resources to prepare budgets.

- After the budget for a project is created and approved, it becomes more of a control mechanism for managers because it establishes the cost threshold for the entire project.

# THE PLANNING STAGE
# - ORGANIZING -

- Organizing is the second key function of the planning stage.
  - It involves two main activities:
    - Responsibility assignments
    - Scheduling

- That is, after identifying and organizing the tasks in the scoping function with the WBS, the organizing function:
  - Conducts task assignments
  - Establishes task durations
    - Task durations are critical because they are integral part of the monitoring and controlling mechanisms in the implementation stage.
  - Identifies any predecessors
    - For each task A, predecessor is a task that must be completed before another task can begin.

- Project management tools that are applicable for organizing include:
  - Linear Responsibility Chart
  - Scheduling with Gantt Charts and Network Diagrams

# THE PLANNING STAGE
# - LINEAR RESPONSIBILITY CHART -

- An important outcome of the planning stage is to have a clear understanding of the roles that each staff member plays in the process of completing each element of the WBS.
  - ✓ This can be viewed as a plan for task assignments.
  - ✓ The goal here is to clearly identify the staff members that will be mainly responsible for each task, and those that will serve as reviewers and/or support the task in any way.
  - ✓ A project management tool that allows such outcome is called a *Linear Responsibility Chart (LRC)*.

- An LRC is a tabular representation of task assignments.
  - ✓ Let's look at an example in the next slide…

# THE PLANNING STAGE
# - LINEAR RESPONSIBILITY CHART -

| Outline # | Task Name | Staff_ 1 | Staff_ 2 | Staff_ 3 | Staff_ 4 | Staff_ 5 |
|---|---|---|---|---|---|---|
| 1 | Design Project | | | | | |
| 1.1 | Architectural Design | 1 | 2 | | | 3 |
| 1.1.1 | Evaluate Alternative Designs | 1 | 2 | | | |
| 1.1.1.1 | Prioritize Objectives | 1 | 2 | | | |
| 1.1.1.2 | Formal Review of Evaluations | | | 1 | 3 | 3 |
| 1.1.2 | Select Architectural Design | 1 | | | | |
| 1.1.2.1 | Make Initial Selection | 1 | | | | |
| 1.1.2.2 | Formal Review to Select Final Design | 1 | | 2 | 3 | 3 |
| 1.2 | Detailed Design | | | | 1 | 3 |
| 1.2.1 | Evaluate Alternatives | | | 2 | 1 | |
| 1.2.2 | Select Among Alternatives | | | 2 | 1 | 3 |
| 1.3 | Documentation | | | | | |
| 1.3.1 | Write Architectural Design Documentation | 1 | 2 | | | |
| 1.3.2 | Formal Review (Architecture Design) | 1 | | | 3 | 3 |
| 1.3.3 | Final Architectural Design Document | 1 | | 3 | | |
| 1.3.4 | Write Detailed Design Documentation | 2 | | | 1 | 3 |
| 1.3.5 | Formal Review (Detailed Design) | 2 | | | 1 | 3 |
| 1.3.6 | Final Detailed Design Document | | | 3 | 1 | 3 |
| | | | | | | |
| **Legend:** | | | | | | |
| 1 = Primary responsibility | | | | | | |
| 2 = Support development role | | | | | | |
| 3 = Review role | | | | | | |

# The Planning Stage
## - Scheduling -

- After task assignments, a key activity in the planning stage is *scheduling.*
  - ✓ Formally defined as the process of establishing task durations in order to convert the WBS into an operating timetable.

- Scheduling helps to identify the critical project activities that cannot be delayed without delaying the overall project duration.
  - ✓ We can also identify those activities that will not affect the overall project duration if delayed.
  - ✓ Scheduling allows tasks to be ordered so that for each task, preceding and subsequent tasks are easily identified.

- There are various project management techniques that are used in scheduling. Two of the most common ones are:
  - ✓ Gantt charts
  - ✓ Network diagrams

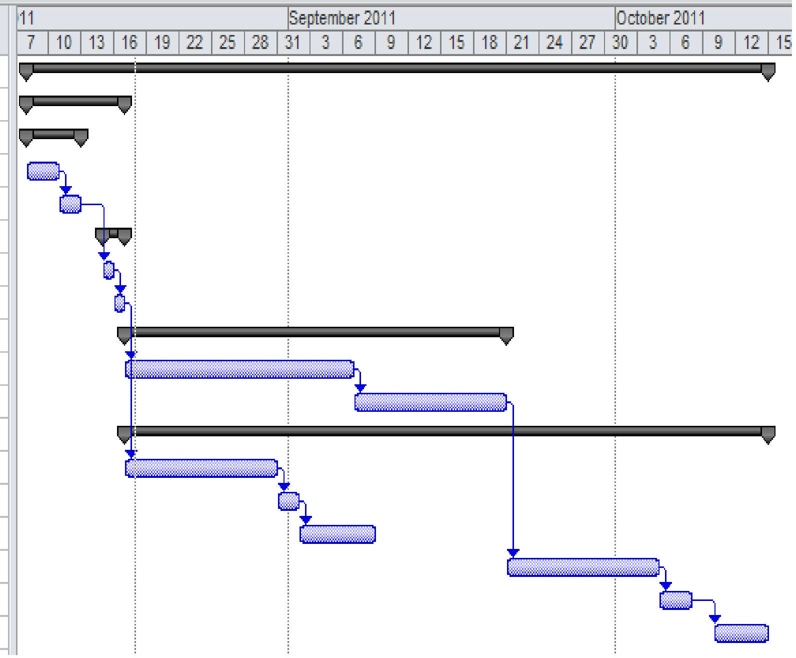
# THE PLANNING STAGE
# - THE GANTT CHART -

- The Gantt Chart is a graphical representation of the logical flow between tasks (i.e., activities) and their durations.
  - ✓ Very simple to follow.
  - ✓ Often used to show schedule and progress, as well as to serve as a control/monitoring mechanism.
  - ✓ Requires three basic parameters:
    - Tasks
    - Durations
    - Predecessors and successors

- These parameters are used to develop a graph that depicts the durations of tasks and their relationship.
  - ✓ Let’s see an example in  the next slide…

# THE PLANNING STAGE
# - THE GANTT CHART -

| | Outline # | Task Name | Duration | Start | Finish | Predecessors |
|---|---|---|---|---|---|---|
| 1 | **1** | **Design Project** | **50 days?** | **Mon 8/8/11** | **Fri 10/14/11** | |
| 2 | **1.1** | **Architectural Design** | **7 days?** | **Mon 8/8/11** | **Tue 8/16/11** | |
| 3 | **1.1.1** | **Evaluate Alternative Designs** | **5 days** | **Mon 8/8/11** | **Fri 8/12/11** | |
| 4 | 1.1.1.1 | Prioritize Objectives | 3 days | Mon 8/8/11 | Wed 8/10/11 | |
| 5 | 1.1.1.2 | Formal Review of Evaluations | 2 days | Thu 8/11/11 | Fri 8/12/11 | 4 |
| 6 | **1.1.2** | **Select Architectural Design** | **2 days?** | **Mon 8/15/11** | **Tue 8/16/11** | |
| 7 | 1.1.2.1 | Make Initial Selection | 1 day? | Mon 8/15/11 | Mon 8/15/11 | 5 |
| 8 | 1.1.2.2 | Formal Review to Select Final Design | 1 day? | Tue 8/16/11 | Tue 8/16/11 | 7 |
| 9 | **1.2** | **Detailed Design** | **25 days** | **Wed 8/17/11** | **Tue 9/20/11** | |
| 10 | 1.2.1 | Evaluate Alternatives | 15 days | Wed 8/17/11 | Tue 9/6/11 | 8 |
| 11 | 1.2.2 | Select Among Alternatives | 10 days | Wed 9/7/11 | Tue 9/20/11 | 10 |
| 12 | **1.3** | **Documentation** | **43 days** | **Wed 8/17/11** | **Fri 10/14/11** | |
| 13 | 1.3.1 | Write Architectural Design Documentation | 10 days | Wed 8/17/11 | Tue 8/30/11 | 8 |
| 14 | 1.3.2 | Formal Review (Architecture Design) | 2 days | Wed 8/31/11 | Thu 9/1/11 | 13 |
| 15 | 1.3.3 | Final Architectural Design Document | 5 days | Fri 9/2/11 | Thu 9/8/11 | 14 |
| 16 | 1.3.4 | Write Detailed Design Documentation | 10 days | Wed 9/21/11 | Tue 10/4/11 | 11 |
| 17 | 1.3.5 | Formal Review (Detailed Design) | 3 days | Wed 10/5/11 | Fri 10/7/11 | 16 |
| 18 | 1.3.6 | Final Architectural Design Document | 5 days | Mon 10/10/11 | Fri 10/14/11 | 17 |



# The Planning Stage
# - Network Diagrams -

- Similar to Gantt charts, network diagrams are also graphical representations of logical flow between tasks, describe task durations, and need the same three basic parameters:
  - ✓ Tasks,
  - ✓ Tasks durations, and
  - ✓ Predecessors and successors

- Estimating accurate task durations is often a very complex endeavor.
  - ✓ One common approach is to estimate minimum ($a$), maximum ($b$), and most likely ($m$) duration times.
  - ✓ With these three parameters we can calculate the expected time (ET) of each task using a probabilistic approach.
  - ✓ Assuming a beta probability distribution for the time estimates, the expected time for each activity can be approximated using the following:
    - Expected Time = $(a + 4m + b) / 6$

- A key benefit for having three estimates for task durations is that we can use them to develop variance calculations, which can then be used to estimate the probability of completing the project within a specific time.
  - ✓ The variance of the duration of a task is calculated using:
    - Variance = $((b-a)/6)^2$

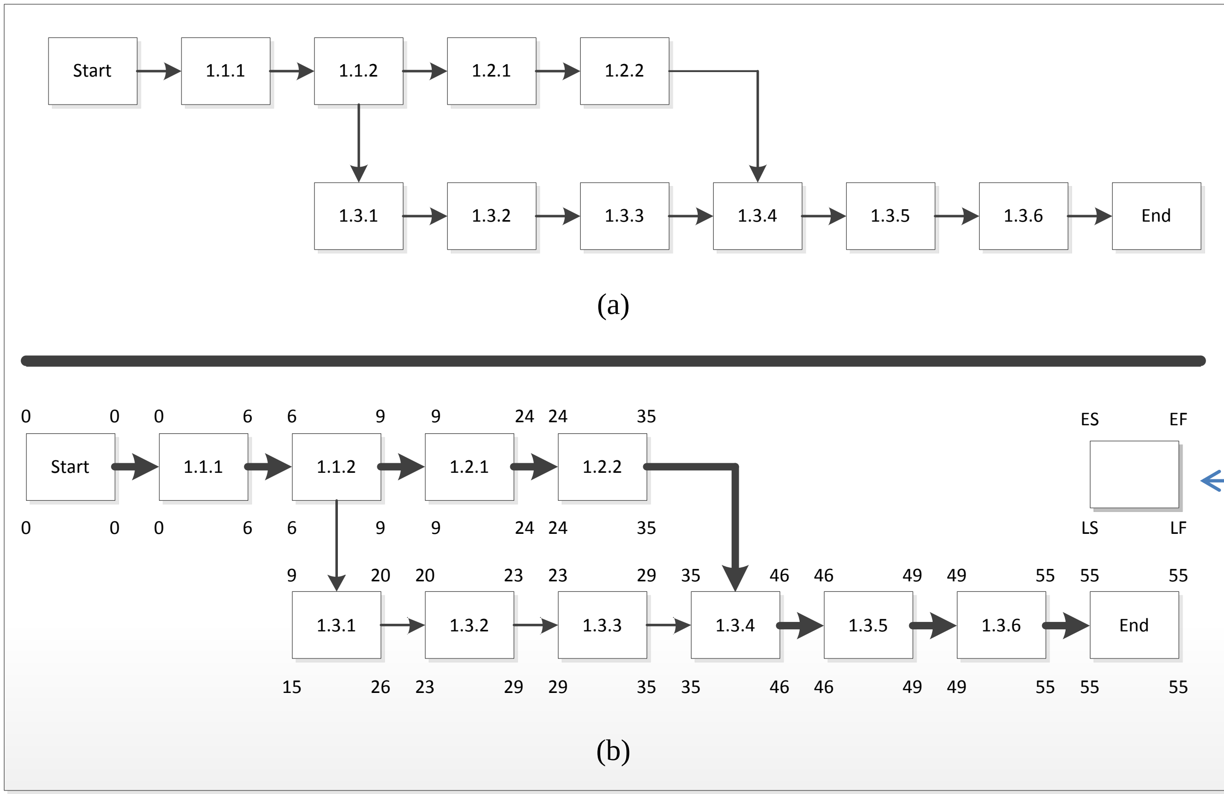
# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

- Example:
  - ✓ Task durations for the three-level tasks described in the previous WBS

| Outline # | Time (days) | | | | | |
|---|---|---|---|---|---|---|
| | Minimum (a) | Most Likely (m) | Maximum (b) | Immediate Predecessors | Expected Time (ET) | Variance ($\sigma^2$) |
| 1.1.1 | 3 | 5 | 13 | - | 6 | 2.78 |
| 1.1.2 | 1 | 2 | 9 | 1.1.1 | 3 | 1.78 |
| 1.2.1 | 8 | 15 | 22 | 1.1.2 | 15 | 5.44 |
| 1.2.2 | 5 | 10 | 21 | 1.2.1 | 11 | 7.11 |
| 1.3.1 | 5 | 10 | 21 | 1.1.2 | 11 | 7.11 |
| 1.3.2 | 1 | 2 | 9 | 1.3.1 | 3 | 1.78 |
| 1.3.3 | 3 | 5 | 13 | 1.3.2 | 6 | 2.78 |
| 1.3.4 | 5 | 10 | 21 | 1.2.2; 1.3.3 | 11 | 7.11 |
| 1.3.5 | 1 | 2 | 9 | 1.3.4 | 3 | 1.78 |
| 1.3.6 | 3 | 5 | 13 | 1.3.5 | 6 | 2.78 |

# The Planning Stage
# - Network Diagrams -

- Resulting AON network diagram.

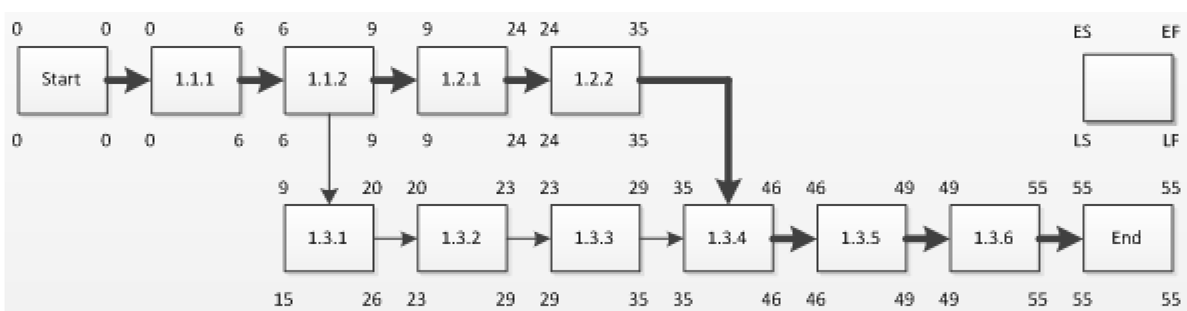



- ES = Earliest Time; EF = Earliest Finish
- LS = Latest Start, LF = Latest Finish

# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

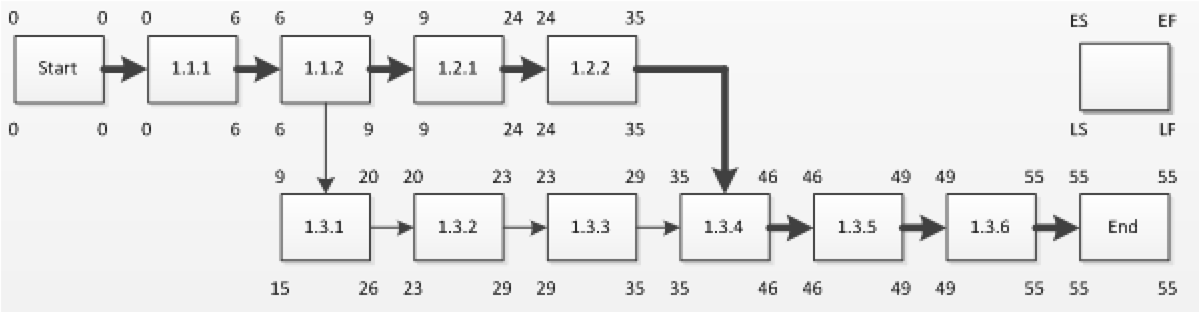
- Network diagrams are used to show additional important information such as:
  - ✓ Critical paths
  - ✓ Earliest and latest completion times
  - ✓ Slack times

- After estimating task durations and variances, the next step is to calculate the critical path.
  - ✓ A path is any combination of nodes from the start node to the last node.
  - ✓ For example, our network diagram shows the following two paths:
    - Path #1: (1.1.1) – (1.1.2) – (1.2.1) – (1.2.2) – (1.3.4) – (1.3.5) – (1.3.6)
    - Path #2: (1.1.1) – (1.1.2) – (1.3.1) – (1.3.2) – (1.3.3) – (1.3.4) – (1.3.5) – (1.3.6)



# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

- The *critical path* is defined as the path with the longest duration.
  - ✓ Using the expected times computed:
    - The expected duration of Path#1 is 55 days.
    - The expected duration of Path#2 is 49 days.
  - ✓ Therefore, the critical path for this problem is Path#1.
  - ✓ The *critical path duration* is 55 days.




- The activities/tasks along the critical path are called critical activities.
  - ✓ If any of the critical activities gets delayed, the entire project will be delayed!

# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

- The next step is to calculate earliest and latest times for activities to start and finish.
  - ✓ That is, we need to calculate the earliest time that we can begin each activity, and the earliest time that the activity can be completed.
  - ✓ We denote these terms as *earliest start (ES)* and *earliest finish (EF)* times.

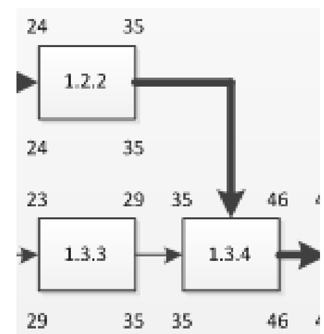
- Similarly, we calculate the latest time that we can begin each activity, and the latest time that the activity can be completed.
  - ✓ We denote these terms as *latest start (LS)* and *latest finish (LF)* times.
  - ✓ The difference between the LS and ES of an activity is called the *slack time* for the activity.
  - ✓ Because activities along the critical path cannot be delayed without delaying the entire project's duration, *only non-critical path activities can have slack times*.

# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

- To calculate earliest times, move from the start node to the other nodes.
  - ✓ Since activity 1.1.1 begins at time 0,
    - Its ES is 0
    - Its EF time is its expected time duration (i.e., 6 days)
  - ✓ Any immediate preceding activity can start as early as activity 1.1.1 ends.
    - Therefore, the ES of activity 1.1.2 is 6 days, and its EF is 9 days, which is the sum of its ES and expected time duration.
    - Similarly, the ES of activity 1.3.1 is 9 days, and its EF is 20 days.
  - ✓ In cases where an activity has more than one predecessor, the largest EF time of the predecessors become the ES time of the activity.
    - For example, activity 1.3.4 has two predecessors: 1.2.2 (with an EF time of 35 days) and 1.3.3 (with an EF time of 29 days).
      - Since activity 1.3.4 can be started only when both of its predecessors complete, its earliest possible start time is 35 days.

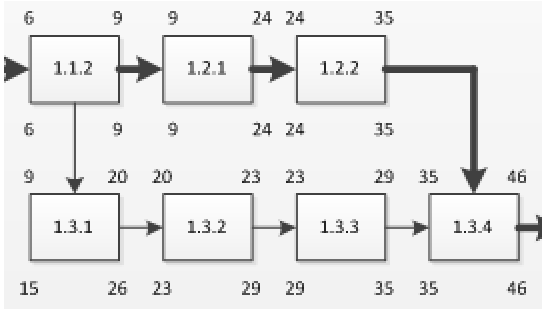
# The Planning Stage
# - Network Diagrams -

- To calculate latest times, we move from the last node to the start node.
  - ✓ For critical activities:
    - LF = EF and
    - LS = ES
  - ✓ This is not the case for non-critical activities. For example, consider 1.3.3:
    - If this activity finishes at its EF time of 29 days, activity 1.3.4 (i.e., its successor) will still be unable to start because it needs to wait for activity 1.2.2 to finish at day 35.
      - Thus, activity 1.3.3 can actually finish as late as day 35 and not affect the critical path duration of the project.
      - Consequently, the LF time for activity 1.3.3 is 35 days, and its LS time is 29 days
        - » 29 days = its LF minus its expected time duration of 6 days.




    - Activity 1.3.2 can finish as late as the LF time of its successor 1.3.3.
      - Therefore, its LF time is 29 days, and its LS time 26 days.

# THE PLANNING STAGE
# - NETWORK DIAGRAMS -




- ✓ When the earliest and latest times for each activity are completed, it can be easily seen that only the non-critical activities (i.e., 1.3.1, 1.3.2, and 1.3.3) have slack times.
  - The total allowable slack time for these activities is 6 days.
    - E.g., if a slack time of 5 days is used in activity 1.3.1, then only one of the remaining non-critical activities can be delayed by 1 day.
    - E.g., if a slack time of 6 days is used in activity 1.3.1, then the network will have two critical paths.
      - » This means that in this case, non-critical activities 1.3.2 and 1.3.3 will become critical activities (i.e., no slack times).

# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

- Probability of Time to Completion
  - In the previous example, expected task durations were calculated based on estimates of minimum, average, and maximum durations using the beta distribution.
  - The other obvious option is to overlook these three parameters and estimate a single average task duration instead.
    - The drawback of this option is that it excludes a range of possibilities (e.g., max/min values) and accurate estimates of task times with a single parameter seems unlikely.
  - Benefits of using min, max and avg. parameters
    - Setting values for these three parameters –instead of one estimate—alleviates the estimation process for decision makers.
    - Estimating these three parameters allows us to develop variance calculations, and then use the variances to estimate the probability of completing the project within a specific time.

# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

- ✓ Estimating the probability of completing a project by a particular time period requires using the *cumulative standard normal distribution* to obtain Z-values.
  - The standard normal distribution is a normal distribution with a mean equal to zero and a standard deviation equal to one.
  - A *Z-value* represents the number of standard deviations from the mean (either to the right or to the left of the mean) of the standard normal distribution.

- ✓ Use the expected completion times of the critical activities and their variances to determine the probability of completing the critical path activities by a deadline.
  - First, transform (a.k.a. "normalize") the estimated project duration time into a *Z*-value using the following equation:

$$Z = \frac{D - EPT}{\sqrt{\sigma^2_{EPT}}}$$

    – Z = the number of standard deviations of a standard normal distribution
    – D = the desired completion time (usually in days) of a project
    – EPT =expected project completion time (this is the critical path duration)
    – $\sigma^2_{EPT}$ = the variance of the critical path

# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

- Example of the cumulative normal distribution table

| Z | 0.00 | 0.01 | 0.02 | 0.03 | 0.04 | 0.05 | 0.06 | 0.07 | 0.08 | 0.09 |
|---|---|---|---|---|---|---|---|---|---|---|
| **0.0** | 0.5000 | 0.5040 | 0.5080 | 0.5120 | 0.5160 | 0.5199 | 0.5239 | 0.5279 | 0.5319 | 0.5359 |
| **0.1** | 0.5398 | 0.5438 | 0.5478 | 0.5517 | 0.5557 | 0.5596 | 0.5636 | 0.5675 | 0.5714 | 0.5754 |
| **0.2** | 0.5793 | 0.5832 | 0.5871 | 0.5910 | 0.5948 | 0.5987 | 0.6026 | 0.6064 | 0.6103 | 0.6141 |
| **0.3** | 0.6179 | 0.6217 | 0.6255 | 0.6293 | 0.6331 | 0.6368 | 0.6406 | 0.6443 | 0.6480 | 0.6517 |
| **0.4** | 0.6554 | 0.6591 | 0.6628 | 0.6664 | 0.6700 | 0.6736 | 0.6772 | 0.6808 | 0.6844 | 0.6879 |
| **0.5** | 0.6915 | 0.6950 | 0.6985 | 0.7019 | 0.7054 | 0.7088 | 0.7123 | 0.7157 | 0.7190 | 0.7224 |
| **0.6** | 0.7258 | 0.7291 | 0.7324 | 0.7357 | 0.7389 | 0.7422 | 0.7454 | 0.7486 | 0.7518 | 0.7549 |
| **0.7** | 0.7580 | 0.7612 | 0.7642 | 0.7673 | 0.7704 | 0.7734 | 0.7764 | 0.7794 | 0.7823 | 0.7852 |
| **0.8** | 0.7881 | 0.7910 | 0.7939 | 0.7967 | 0.7996 | 0.8023 | 0.8051 | 0.8079 | 0.8106 | 0.8133 |
| **0.9** | 0.8159 | 0.8186 | 0.8212 | 0.8238 | 0.8264 | 0.8289 | 0.8315 | 0.8340 | 0.8365 | 0.8389 |
| **1.0** | 0.8413 | 0.8438 | 0.8461 | 0.8485 | 0.8508 | 0.8531 | 0.8554 | 0.8577 | 0.8599 | 0.8621 |
| **1.1** | 0.8643 | 0.8665 | 0.8686 | 0.8708 | 0.8729 | 0.8749 | 0.8770 | 0.8790 | 0.8810 | 0.8830 |
| **1.2** | 0.8849 | 0.8869 | 0.8888 | 0.8907 | 0.8925 | 0.8944 | 0.8962 | 0.8980 | 0.8997 | 0.9015 |
| **1.3** | 0.9032 | 0.9049 | 0.9066 | 0.9082 | 0.9099 | 0.9115 | 0.9131 | 0.9147 | 0.9162 | 0.9177 |
| **1.4** | 0.9192 | 0.9207 | 0.9222 | 0.9236 | 0.9251 | 0.9265 | 0.9279 | 0.9292 | 0.9306 | 0.9319 |
| **1.5** | 0.9332 | 0.9345 | 0.9357 | 0.9370 | 0.9382 | 0.9394 | 0.9406 | 0.9418 | 0.9430 | 0.9441 |
| **1.6** | 0.9452 | 0.9463 | 0.9474 | 0.9485 | 0.9495 | 0.9505 | 0.9515 | 0.9525 | 0.9535 | 0.9545 |
| **1.7** | 0.9554 | 0.9564 | 0.9573 | 0.9582 | 0.9591 | 0.9599 | 0.9608 | 0.9616 | 0.9625 | 0.9633 |
| **1.8** | 0.9641 | 0.9649 | 0.9656 | 0.9664 | 0.9671 | 0.9678 | 0.9686 | 0.9693 | 0.9700 | 0.9706 |
| **1.9** | 0.9713 | 0.9719 | 0.9726 | 0.9732 | 0.9738 | 0.9744 | 0.9750 | 0.9756 | 0.9762 | 0.9767 |
| **2.0** | 0.9773 | 0.9778 | 0.9783 | 0.9788 | 0.9793 | 0.9798 | 0.9803 | 0.9808 | 0.9812 | 0.9817 |
| **2.1** | 0.9821 | 0.9826 | 0.9830 | 0.9834 | 0.9838 | 0.9842 | 0.9846 | 0.9850 | 0.9854 | 0.9857 |
| **2.2** | 0.9861 | 0.9865 | 0.9868 | 0.9871 | 0.9875 | 0.9878 | 0.9881 | 0.9884 | 0.9887 | 0.9890 |
| **2.3** | 0.9893 | 0.9896 | 0.9898 | 0.9901 | 0.9904 | 0.9906 | 0.9909 | 0.9911 | 0.9913 | 0.9916 |
| **2.4** | 0.9918 | 0.9920 | 0.9922 | 0.9925 | 0.9927 | 0.9929 | 0.9931 | 0.9932 | 0.9934 | 0.9936 |
| **2.5** | 0.9938 | 0.9940 | 0.9941 | 0.9943 | 0.9945 | 0.9946 | 0.9948 | 0.9949 | 0.9951 | 0.9952 |
| **2.6** | 0.9953 | 0.9955 | 0.9956 | 0.9957 | 0.9959 | 0.9960 | 0.9961 | 0.9962 | 0.9963 | 0.9964 |
| **2.7** | 0.9965 | 0.9966 | 0.9967 | 0.9968 | 0.9969 | 0.9970 | 0.9971 | 0.9972 | 0.9973 | 0.9974 |
| **2.8** | 0.9974 | 0.9975 | 0.9976 | 0.9977 | 0.9977 | 0.9978 | 0.9979 | 0.9980 | 0.9980 | 0.9981 |
| **2.9** | 0.9981 | 0.9982 | 0.9983 | 0.9983 | 0.9984 | 0.9984 | 0.9985 | 0.9985 | 0.9986 | 0.9986 |
| **3.0** | 0.9987 | 0.9987 | 0.9987 | 0.9988 | 0.9988 | 0.9989 | 0.9989 | 0.9989 | 0.9990 | 0.9990 |
| **3.1** | 0.9990 | 0.9991 | 0.9991 | 0.9991 | 0.9992 | 0.9992 | 0.9992 | 0.9992 | 0.9993 | 0.9993 |
| **3.2** | 0.9993 | 0.9993 | 0.9994 | 0.9994 | 0.9994 | 0.9994 | 0.9994 | 0.9995 | 0.9995 | 0.9995 |
| **3.3** | 0.9995 | 0.9995 | 0.9996 | 0.9996 | 0.9996 | 0.9996 | 0.9996 | 0.9996 | 0.9996 | 0.9997 |
| **3.4** | 0.9997 | 0.9997 | 0.9997 | 0.9997 | 0.9997 | 0.9997 | 0.9997 | 0.9997 | 0.9998 | 0.9998 |
| **3.5** | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.9998 |

# THE PLANNING STAGE
# - NETWORK DIAGRAMS -

- ✓ With the Z-value, find the probability of completing the project by a certain time using a cumulative normal distribution table
  - In this table we can find the probability associated with a Z-value.

- ✓ For example, assume that a design project has an expected critical path duration of 75 days, with a critical path variance of 36 days. What is the probability of completing the project in 85 days.
  - Step #1: Calculate a Z-value
    - Z-value = 1.67 using the equation $Z = \frac{D - EPT}{\sqrt{\sigma^2_{EPT}}}$
  - Step #2: Find the probability using the cumulative normal distribution table
    - Probability = 0.9525 (go down the left column until 1.6, then across to the right until column 0.07).
    - This means that there is a 95.25 percent probability of finishing the design project in 85 days, given the variance in the expected project duration time.

# THE PLANNING STAGE
# -ESTABLISH CHANGE CONTROL POLICY-

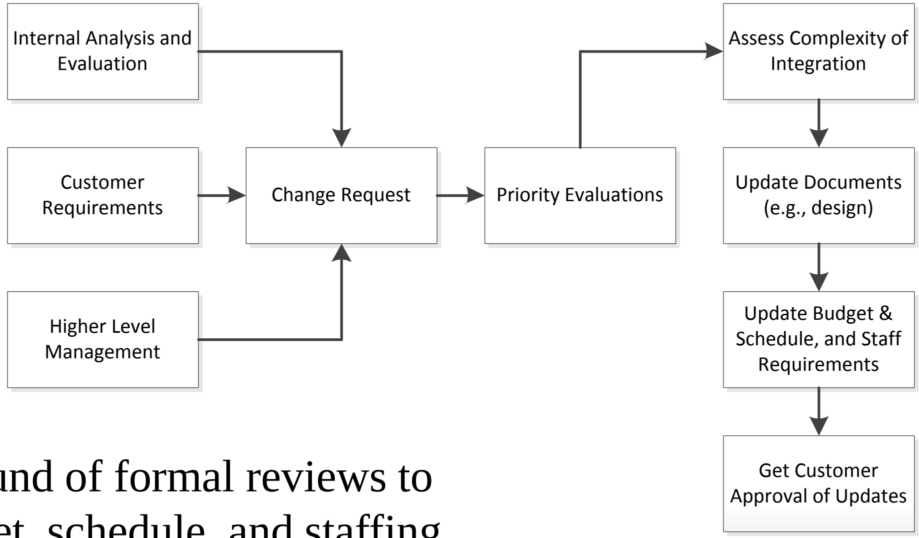
- Managing the design of software systems is often a very complex endeavor, especially when changes to the design occur after the design has successfully gone through peer reviews.
  - ✓ Even a small design change can result in system failures if not managed appropriately.
  - ✓ To avoid major problems later in the project, it is critical that a formal change control process gets adopted early in the planning phase.

- A change control process helps to ensure the technical integrity of the design in the presence of changes.

- Let’s see an example of a change control process…..

# ESTABLISH CHANGE CONTROL POLICY (AN EXAMPLE)

- Steps:
  - ✓ A change request is submitted to the Change Control Authority (i.e., the project manager, the design lead, or the group that makes the final decision on the status and priority of a change.)
  - ✓ Request is evaluated based on technical merit, complexity of integration, & potential side effects.
  - ✓ From the assessments, the request is assigned a change priority.




  - ✓ Developers then take their initial design documents through another round of formal reviews to obtain updated estimates for the budget, schedule, and staffing.
  - ✓ Finally, updated design documents and schedule estimates must then be approved by the customer since they represent changes to the original estimates.

## THE IMPLEMENTATION STAGE - EARNED VALUE MANAGEMENT (EVM)-

- During the implementation stage, main managerial activities deal with *monitoring and controlling* that the design project activities are being accomplished according to the plan (i.e., schedule and budget).
  - ✓ Monitoring and controlling mechanisms exist to:
    - Identify any deviations from the plan, and
    - Ensure that these deviations get corrected in a timely manner in order to avoid major future problems.

- Appropriate project management tools for this phase are:
  - ✓ *Gantt charts* and
  - ✓ *Earned value management*.

- Since we have already covered Gantt charts in the previous section, let's focus now on *earned value management* (EVM).

# THE IMPLEMENTATION STAGE
# - EARNED VALUE MANAGEMENT (EVM)-

- EVM is a project management technique to determine the progress of tasks based on the value of the work currently completed versus the work that was expected to be completed at that particular time.

- The term used to define the dollar amount of the work currently completed is called *earned value (EV)*.
  - ✓ EV is calculated by multiplying the percentage work completed times the planned total cost of the work when completed.

$$EV = (\%\ \text{work completed}) * \text{planned cost for the work}$$

  - ✓ A key factor is to accurately estimate the percentage of work completed,
    - Challenges to estimate % work completed:
      - Complexity of tasks and high degree of subjectivity that is typically involved.
      - Workers tend to inflate the % of work completed to give the perception that work is progressing smoothly for the completion of the task.
    - There are various ways to estimate percentage completion.
      - Let's see some guidelines to estimate % completion of a task…..

# THE IMPLEMENTATION STAGE
# - EARNED VALUE MANAGEMENT (EVM)-

- Guidelines for estimating % completion for a task [1]

| Approach Name | Brief Description |
|---|---|
| 50-50 | Assumes 50% task completion as soon as the task begins, and 100% completion when the task is completed |
| 0-100 percent | Assumes 0% until the task gets fully completed |
| Critical input use | Percent completion is a function of how much of a critical input is used versus its overall expected total to be used |
| Proportionality | Assumes work completion as the proportion of actual time (or cost) spent over the total time (or cost) planned for the task to be completed |

- With EV calculations for each task (i.e., each element in the WBS), we can:
  - ✓ Examine deviations to cost and schedule, and then
  - ✓ Establish a dollar amount to these deviations.

# The Implementation Stage
# - Earned Value Management (EVM)-

- EVM Definition of Parameters

| Parameter | Description |
|---|---|
| EV | Also called *budgeted cost of work performed (BCWP),* it is the dollar amount of the work currently completed |
| AC | Also called *actual cost of work performed (ACWP)*, it represents the amount of money that was spent in the work currently completed |
| PV | Also called *budgeted cost of work scheduled (BCWS)*, it represents the value of the work that was expected to be completed at that point in time of the schedule |
| CV | Describes the difference between the value of work completed and the actual cost spent to complete it |
| SV | Describes the difference between the value of work currently completed and the cost of the work that was expected to be completed |
| CPI | Describes the current cost efficiency of the work ( CPI < 1 means over budget; CPI > 1 means under budget; CPI = 1 means right on budget) |
| SPI | Describes the current schedule efficiency of the work ( SPI < 1 means behind schedule; SPI > 1 means ahead of schedule; SPI = 1 means right on schedule) |
| BAC | Total budget for the project |
| ETC | Represents the estimated cost remaining to complete the project |
| EAC | Represents is the total cost of the entire project |

- Equations for EVM parameters

| Parameter | Equation |
|---|---|
| EV | EV = (% work completed) * planned cost for the work |
| CV | CV = EV - AC |
| SV | SV = EV - PV |
| CPI | CPI = EV / AC |
| SPI | SPI = EV / PV |
| ETC | ETC = (BAC – EV) / CPI |
| EAC | EAC = ETC + AC |

# The Implementation Stage
## - Earned Value Management (EVM)-

- Let's show the implementation of the EVM equations with an example:

*Sara is managing a software design project composed of 10 tasks. Each task was estimated to cost $1,000; therefore, BAC = $10,000. Each task was estimated to be completed in a month; therefore, the total planned duration for the project was estimated to be 10 months. Since each task is expected to have the same cost and duration, the completion of an individual task represents 10% completion of the design project. In the current fifth month, only three tasks have been fully completed at a cost of $4,000. Sara needs to provide upper management with current progress status.*

- We can now use the EVM equations

- Let's examine what this means…

| Parameter | Calculation |
|---|---|
| EV | 0.3 * $10,000 = $3,000. |
| CV | $3,000 - $4,000 = -$1,000 |
| SV | $3,000 - $5,000 = -$2,000 |
| CPI | $3,000 / $4,000 = 0.75 |
| SPI | $3,000 / $5,000 = 0.6000 |
| ETC | ($10,000 - $3,000) / 0.75 = $9,333 |
| EAC | $9,333 + $7,000 = $16,333 |

## THE IMPLEMENTATION STAGE
## - EARNED VALUE MANAGEMENT (EVM)-

- EV is calculated to be $3,000 given that the three tasks completed represent 30% of the project, and the budget for the entire project is $10,000.

- To calculate SV, we use a PV of $5,000 because each task was budgeted at $1,000 and five tasks should have been completed in the fifth month.

- The Table in the previous slide shows the calculations for the rest of the EVM parameters.

- These results clearly show that the project is over both budget and schedule estimates.
  - ✓ The CPI index, for example, shows that only $0.75 of earned value was received for every dollar actually spent.
  - ✓ The EAC shows an estimated $6,333 over budget amount.
  - ✓ These alarming values give Sara and upper management the opportunity to implement risk mitigation policies to either get the project to its budgeted path, or at least attempt to minimize losses.

# THE TERMINATION STAGE

- During the termination stage, key activities involve verification that everything is in place for a smooth transition into the code construction phase.

- Some of the activities during the termination stage include:
  - ✓ Making sure that the latest versions of the design documents are securely stored based on configuration management procedures,
  - ✓ Updating schedule and cost current values,
  - ✓ Re-evaluating schedule and budget plans based on the resulting performance measures of the design phase, and
  - ✓ Communicating results to upper management.

# WHAT'S NEXT…

- This session accomplished the following:
  - ✓ Defined the *Planning* component of the software design management framework
  - ✓ Defined the *Implementation* component of the software design management framework
  - ✓ Defined the *Termination* component of the software design management framework

- Next session will focus on the *Leadership & Ethics* component of the design management framework,

# QUESTIONS?

# REFERENCES

[1] Meredith, Jack, and Samuel Mantel. *Project Management: A Managerial Approach*. 7th. John Wiley & Sons, 2009.