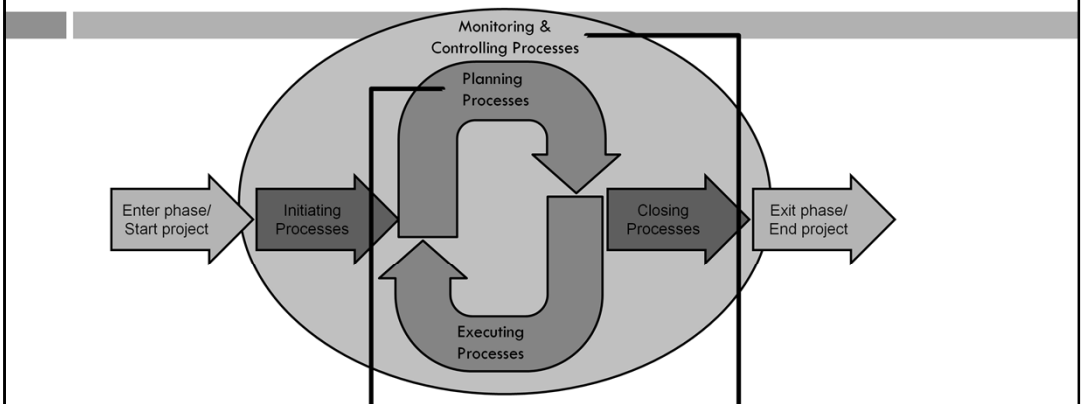
U08784
SOFTWARE PROJECT MANAGEMENT

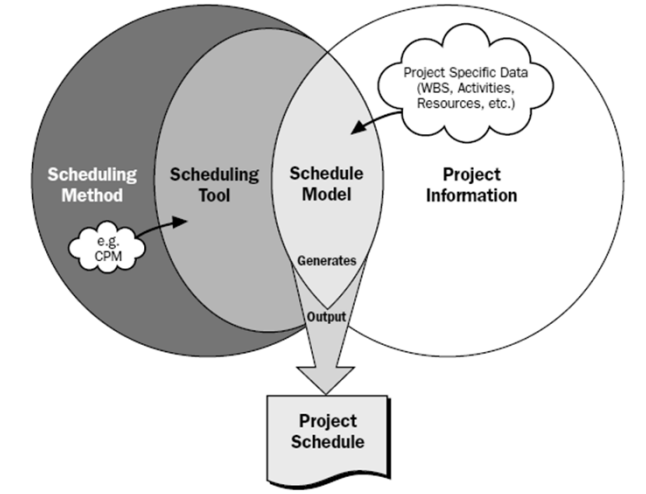
Lecture 4: Project Time Management

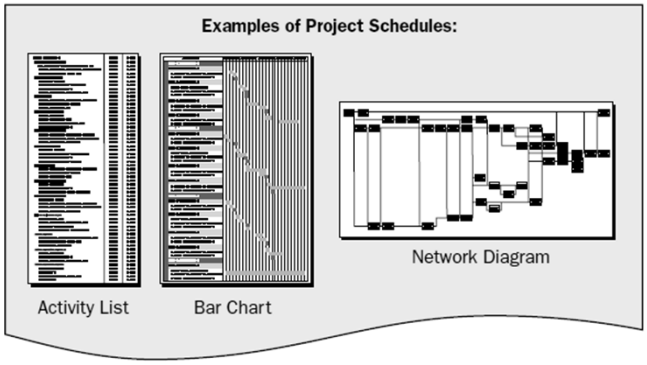
Peter Lo

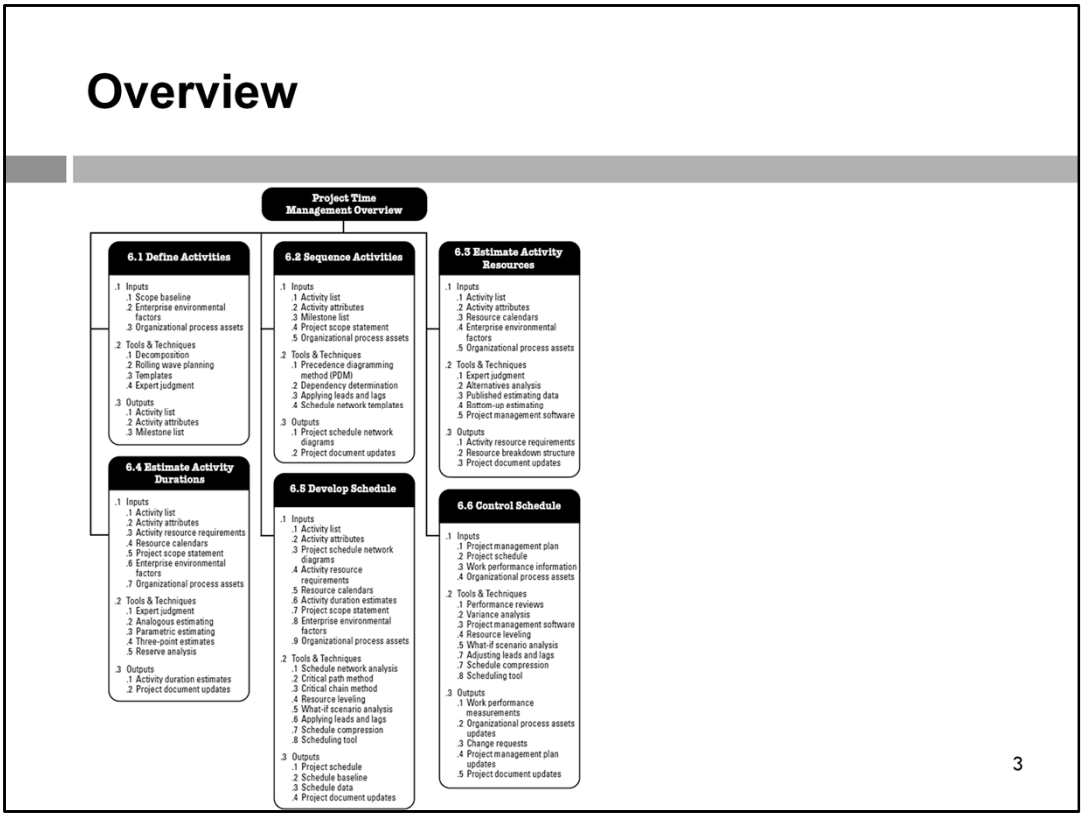
Overview



Knowledge Area	Process				
	Initiating	Planning	Executing	Monitoring & Control	Closing
Time		Activity Definition Activity Sequencing Activity Resource Estimating Activity Duration Estimating Schedule Development		Schedule Control	



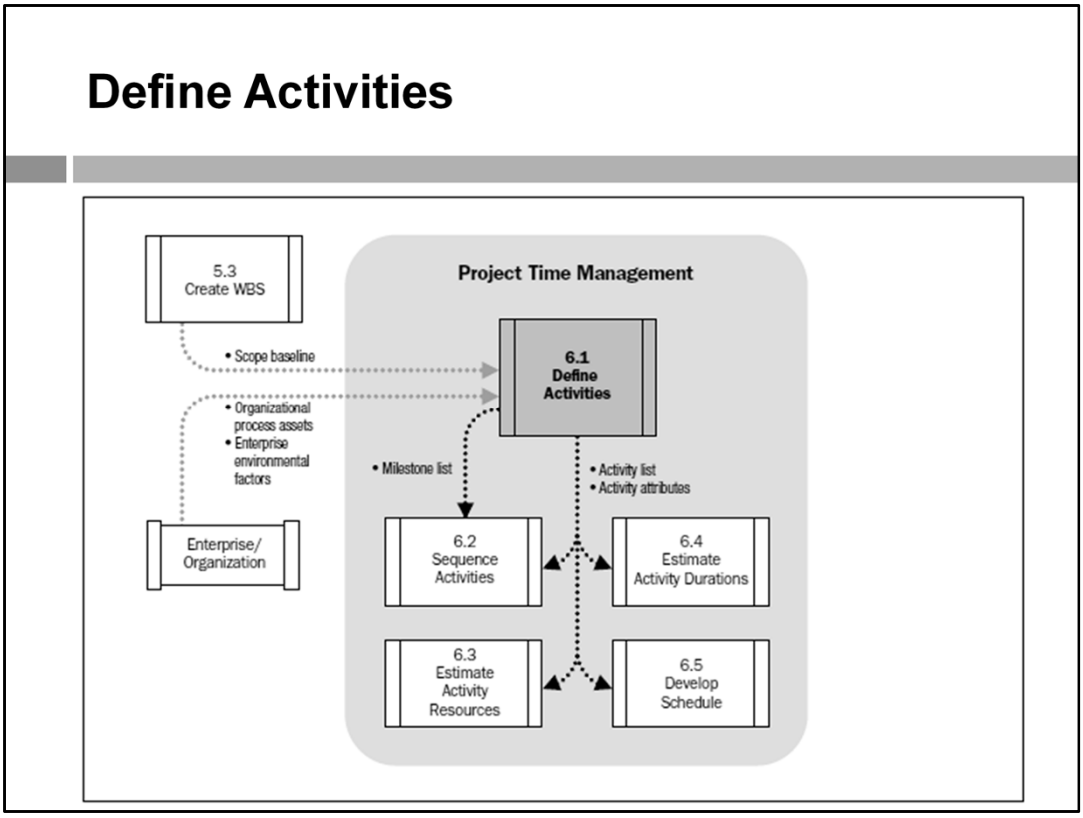




The process required to manage timely completion of the project. Project time management start with planning by the project management team (not shown as a discrete process). In small project, defining & sequencing activities, estimating activity resource & duration, developing schedule are viewed as a single process.

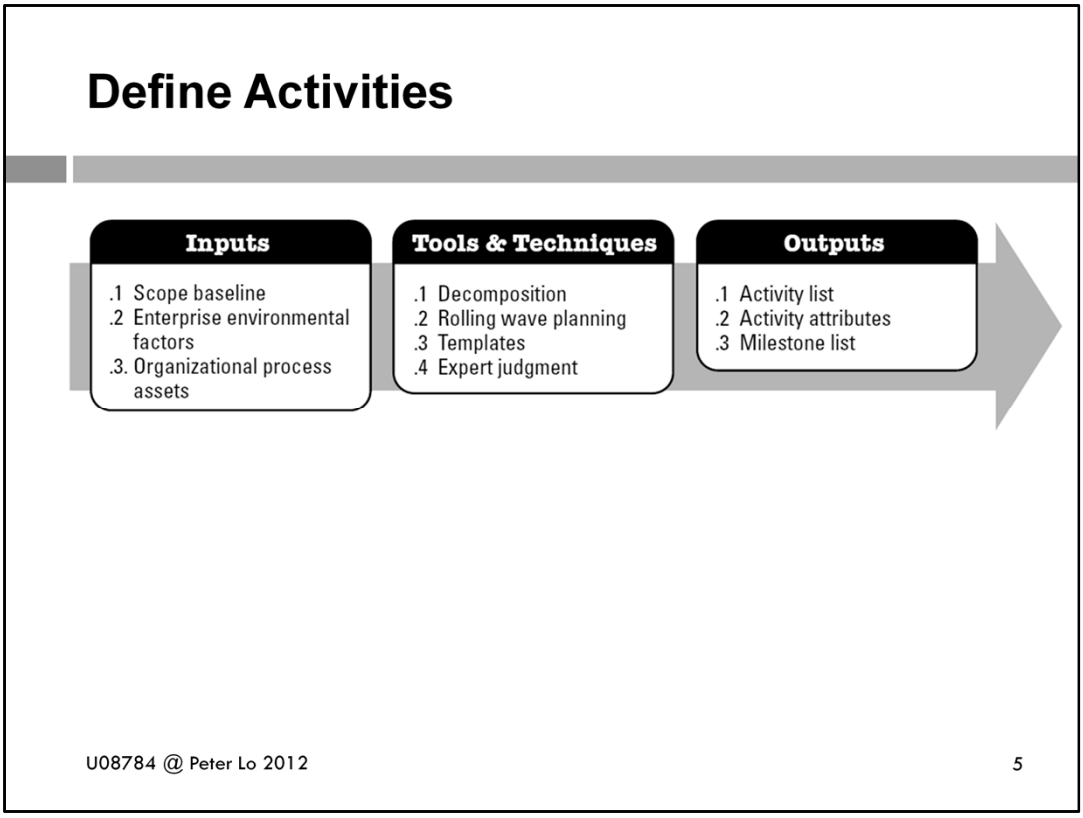
Project Time Management includes the processes required to manage timely completion of the project.

- **Define Activities** – The process of identifying the specific actions to be performed to produce the project deliverables.
- **Sequence Activities** – The process of identifying and documenting relationships among the project activities.
- **Estimate Activity Resources** – The process of estimating the type and quantities of material, people, equipment, or supplies required to perform each activity.
- **Estimate Activity Durations** – The process of approximating the number of work periods needed to complete individual activities with estimated resources.
- **Develop Schedule** – The process of analyzing activity sequences, durations, resource requirements, and schedule constraints to create the project schedule.
- **Control Schedule** - The process of monitoring the status of the project to update project progress and managing changes to the schedule baseline.



The define activities process is a further breakdown of the work packages elements of the WBS. It documents the specific activities needed to fulfill the deliverables detailed in the WBS.

- Identify the deliverables at the lowest level in the WBS, which is called the Work Package
- Identifying and documenting the work that is planned to be performed
- Identifying the specific activities to produce the project deliverables
- Defining and planning the schedule activities such that the project objectives will be met



Inputs

- Scope Baseline
- Enterprise Environmental Factors
- Organizational Process Assets

Tools & Techniques

- Decomposition (Breaking the work packages into smaller, more manageable units of work called activities - not deliverables but individual units of work).
- Rolling wave planning (Planning near term work in more detail than future term work. It's a form of progressive elaboration)
- Template
- Expert Judgment (Project team members with prior experience developing project scope statements and WBS, can help you here)

Outputs

- Activity List (should contain all the schedule activities with a scope of work description of each activity and an identifier)
- Activity attributes (Describe the characteristics of the activities and are an extension of the activity list)
- Milestone Lists (Major accomplishments of the project and they mark the completion of major deliverables or some other key event in the project. They record the accomplishments and document whether it is mandatory or optional)

Define Activities (Input)

- Scope baseline
  - Deliverables, constraints, assumptions
- Enterprise Environmental Factors
  - PMIS (Program Management Information Systems)
- Organizational Project Assets
  - Existing guidelines, Internal policies, procedure, lessons learned, knowledge database

U08784 @ Peter Lo 2012 6

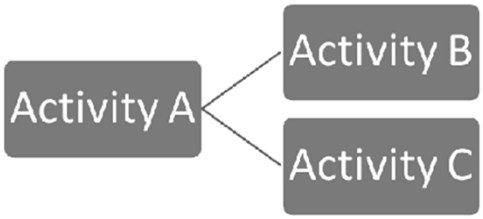
Scope Baseline
- Project deliverables, constraints, assumptions documented

Enterprise environmental factors
- PMIS (Program Management Information Systems)
- Organization culture, infrastructure, tools, HR, market conditions

Organizational process assets
- Existing formal and informal activity planning related policies, procedures, and guidelines
- Existing guidelines and policies
- Lessen learned

Define Activities (Tools and Techniques)

- Decomposition
  - Subdividing project work into smaller, more manageable components



Decomposition
- Subdividing the project work package
- Developing WBS, WBS dictionary and decomposing it for development of the final activity list

Define Activities (Tools and Techniques)

- Templates
  - A standard activity lists from previous projects
  - Used to identify typical schedule milestones

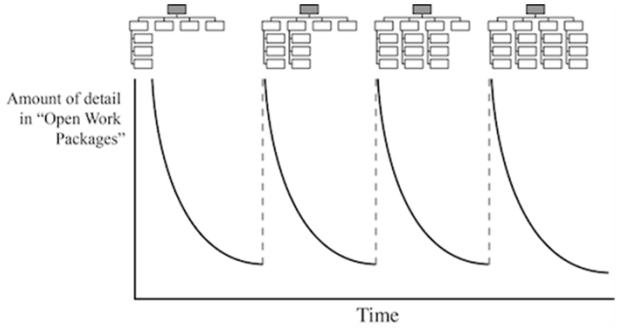
Project Acronym	Activity Attribute Worksheet	Your Logo
Customer Name	Project # No.:	

ID: *From activity list*	Activity: *From activity list*				
Description of Work: *A description of the activity in enough detail so that the person(s) performing the work understands what is required to complete it.*					
Predecessors	**Relationship**	**Lead or Lag**	**Successor**	**Relationship**	**Lead or Lag**
Any activities that must occur before the activity.	*The nature of the relationship, such as start-to-start, finish-to-start, or finish-to-finish.*	*Any required delays between activities (lag) or accelerations (lead).*	*Any activities that must occur after the activity.*	*The nature of the relationship, such as start-to-start, finish-t-start, or finish-to-finish.*	*Any required delays between activities (lag) or accelerations (lead).*
Number and Type of Resources Required:	**Skill Requirements:**	**Other Required Resources:** *Any equipment, supplies, or other types of resources needed to complete the work.*			
Type of Effort: *Indicate if the work is a fixed duration, fixed amount of effort, level of effort, apportioned effort or other type of work.*					

Templates
- A standard activity lists from previous projects
- Used to identify typical schedule milestones

Define Activities (Tools and Techniques)

- Rolling Wave Planning
  - Work to be accomplished in the near future is planned in detail, future work is planned at a higher level
  - Progressive elaboration planning where you do not to plan activities until you start the project management process for that phase is in the project life cycle



U08784 @ Peter Lo 2012 9

Rolling Wave Planning
- Progressive elaboration planning where the work to be accomplished in the near term is planned in detail at a low level of the WBS
- Schedule activities can exist at various levels of detail in the project life cycle
- When insufficient definition of the project scope is available
  - Control Account - documented in a control account plan
  - Planning Package

Rolling Wave Planning is just what it sounds like. In a company you may do five year plans every year. Each year you plan for the next five and update the plan from last year. In a project you may plan near term work in greater detail and at a low level in WBS, but plan work farther in the future at a high level. The key is you keep rolling the wave of planning. Every month you may plan ahead for five months. Consider Rolling-Wave technique for large, complicated, and unfamiliar projects/sub-projects.

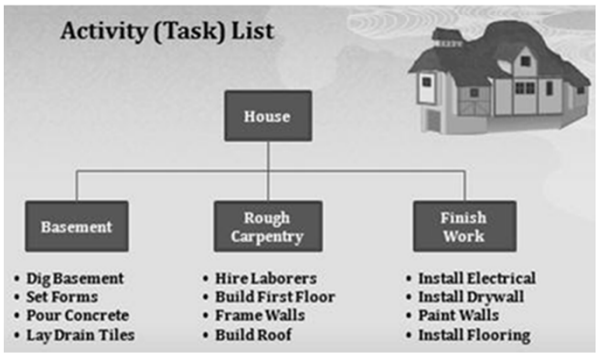
Define Activities (Tools and Techniques)

- Expert Judgment
  - Project team members or other experts, who are experienced and skilled in developing detailed project scope statements, the WBS, and project schedules, can provide expertise in defining activities.

Expert judgment
- Other units in the organization
- Consultants
- Stakeholders
- Professional/technical associations
- Industry groups
- Subject matter experts

Define Activities (Output)

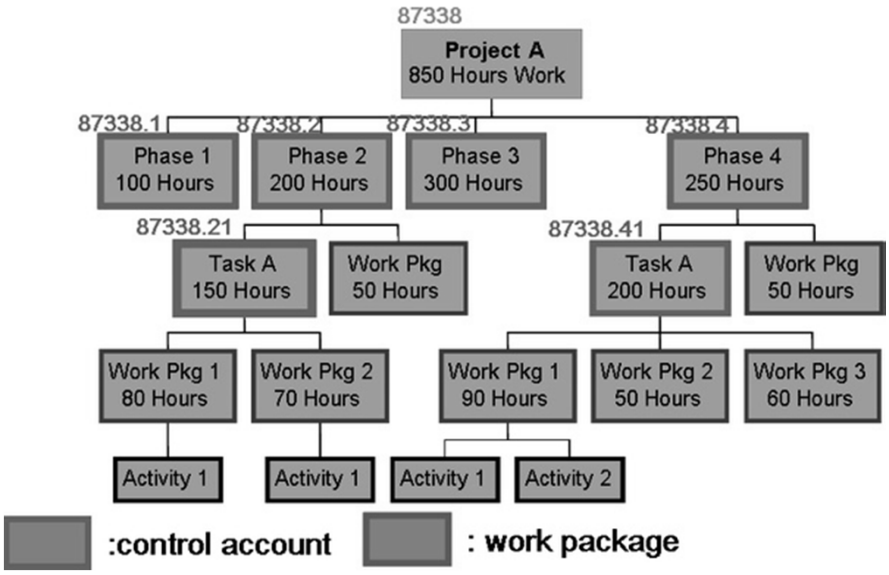
- Activity List
  - A comprehensive list including all schedule activities
  - Used in the schedule model and is a component of the project management plan



U08784 @ Peter Lo 2012 11

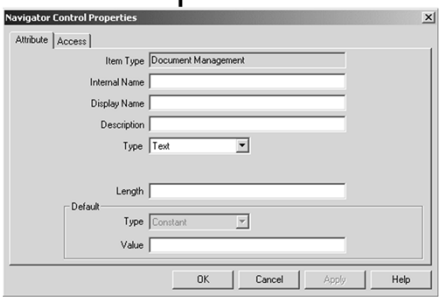
Activity list

- A comprehensive list including all schedule activities required for the project
- Sufficient detail to understand what is required to be completed
- Scope of work description for each activity
- Use for schedule development, selecting, ordering, sorting the planned schedule activities
- Activity identifier, used to identify e.g. responsible person, place, level of effort (LOE), apportioned effort (AE)



Define Activities (Output)

- Activity Attributes
  - Extension of the activity attributes
  - E.g. the activity identifier, activity codes, active description, predecessor activities, logical relationships



U08784 @ Peter Lo 2012 12

Activity attributes

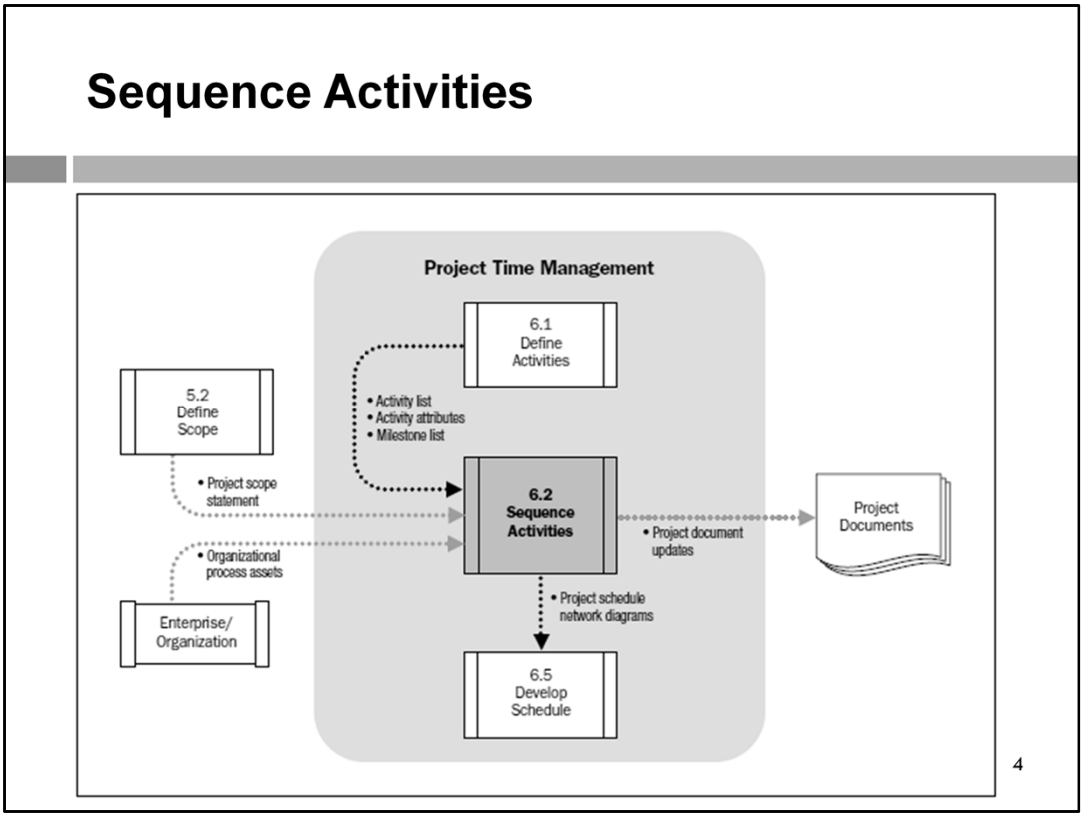
- Activity ID, WBS ID, activity name, when completed activity codes, activity description, predecessors, successors, logical relationships, leads, lags, resource requirement, imposed dates, constraints, and assumptions, person responsible, geographic area, Level of Effort (LOF)

Define Activities (Output)

- Milestone List
  - Indicates whether the milestone is mandatory (required by the contract) or optional

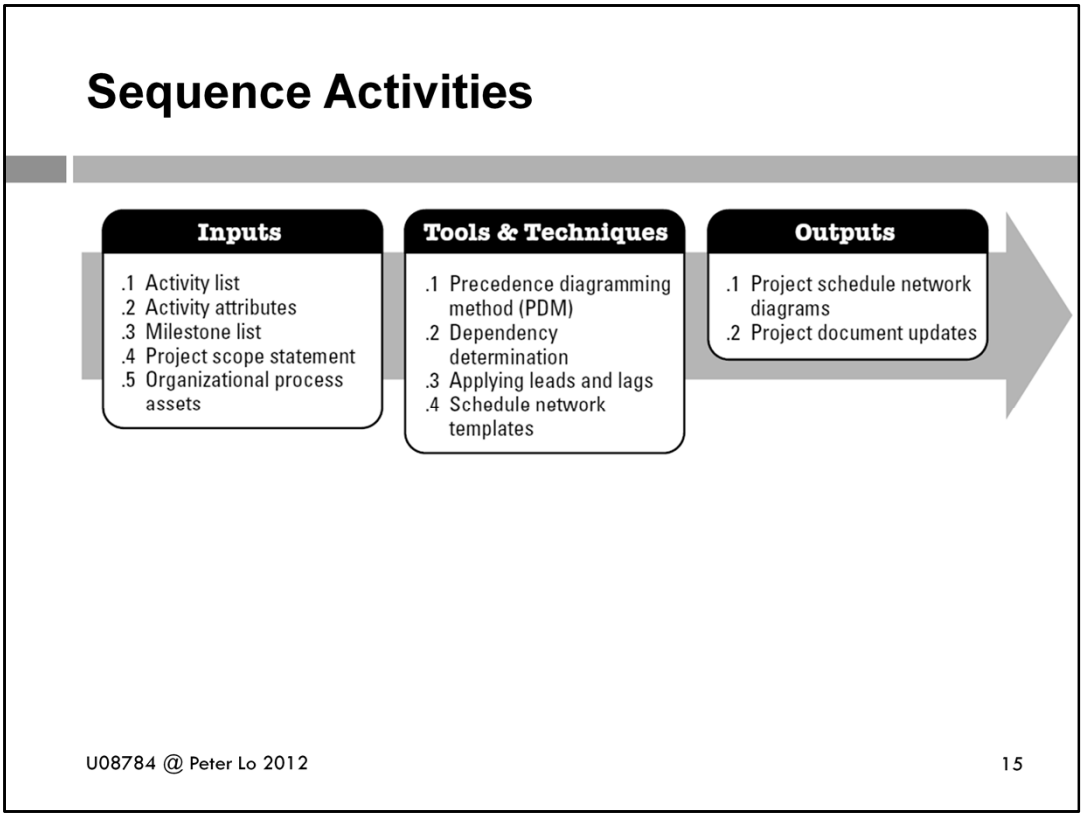
Title	Subtitl	Type	Project	Flag	Notice
Acknowledgements document	Milestone	Activity	Ph.D.		
Analysis summary document fc	Milestone	Activity	Ph.D.		
Analysis summary of Exp 1 anc	Milestone	Activity	Ph.D.		
Appendix 1 document	Milestone	Activity	Ph.D.		
Appendix 2 document	Milestone	Activity	Ph.D.		
Appendix 3 document	Milestone	Activity	Ph.D.		
Appendix 4 document	Milestone	Activity	Ph.D.		
Appendix 5 document	Milestone	Activity	Ph.D.		
Appendix 6 document	Milestone	Activity	Ph.D.		
Appendix 7 document	Milestone	Activity	Ph.D.		
Appendix 8 document	Milestone	Activity	Ph.D.		
Bibliography document	Milestone	Activity	Ph.D.		
CVS/Subversion everything	Milestone	Activity	Ph.D.		
Chapter 1 document	Milestone	Activity	Ph.D.		
Chapter 10 document	Milestone	Activity	Ph.D.		
Chapter 11 document	Milestone	Activity	Ph.D.		
Chapter 2 document	Milestone	Activity	Ph.D.		
Chapter 3 document	Milestone	Activity	Ph.D.		
Chapter 4 document	Milestone	Activity	Ph.D.		

Milestone List
- A significant point or event in the project.
- Not a work activity
- Checkpoint to help control the project
- Additional milestone can be add in Sequence Activities & Develop Schedule process
- The list can indicates the level of milestone (mandatory, optional, etc)



Sequence Activities is the process of identifying and documenting relationships among the project activities.

- The process of identifying and documenting relationships among the project activities
- Every activity (except the first and the last ones) is connected to one predecessor and one successor
- Provides leads and lags to support later development of a realistic and achievable project schedule
- PM software, manual, and/or automated



Input
- Activity List
- Activity Attributes
- Milestone List
- Project Scope Statement
- Organizational Process Asset

Tools and Techniques
- Precedence Diagramming Method (PDM)
- Dependency Determination
- Apply Leads and Lags
- Schedule Network Template

Outputs
- Project schedule network diagrams
- Project document updates.

Sequence Activities (Input)

- Activity List
- Activity Attributes
- Milestone List
- Project Scope Statement
- Organizational Process Asset

U08784 @ Peter Lo 2012 16

Activity List
- The activity list is a comprehensive list including all schedule activities required on the project.
- The activity list includes the activity identifier and a scope of work description for each activity in sufficient detail to ensure that project team members understand what work is required to be completed

Activity Attributes
- Activity attributes may describe a necessary sequence of events or defined predecessor or successor relationship.

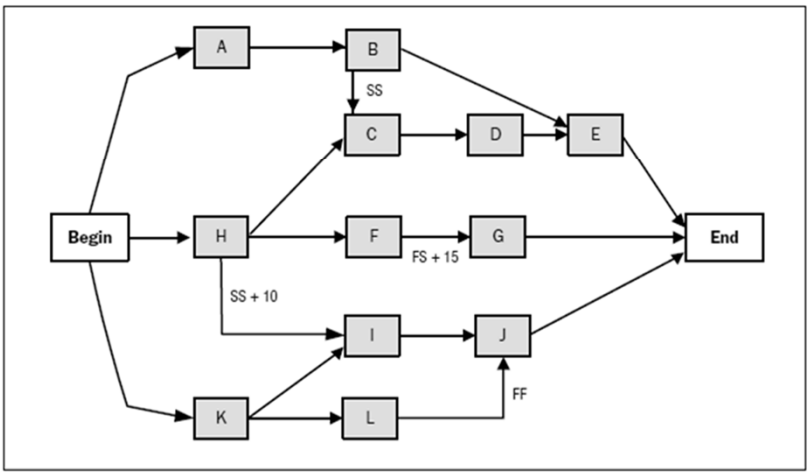
Milestone List
- The milestone list may have scheduled dates for specific milestones.

Project Scope Statement
- The project scope statement contains the product scope description, which includes product characteristics that may affect activity sequencing, such as the physical layout of a plant to be constructed or subsystem interfaces on a software project.
- While these effects are often apparent in the activity list, the product scope description is generally reviewed to ensure accuracy.

Organizational Process Assets
- The organizational process assets that can influence the Sequence Activities process include, but are not limited to, project files from the corporate knowledge base used for scheduling methodology.

Sequence Activities (Tools and Techniques)

- Precedence Diagramming Method (PDM)



12 Activities (not include Begin and End)
19 Logic Dependencies (No. of Line)

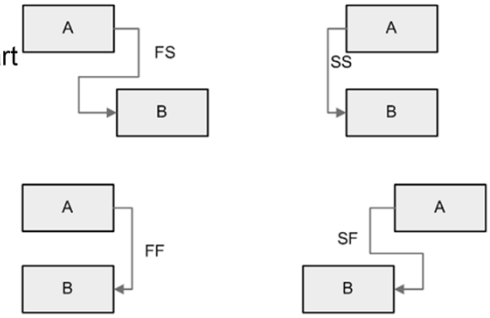
U08784 @ Peter Lo 2012 17

Precedence Diagramming Method (PDM)

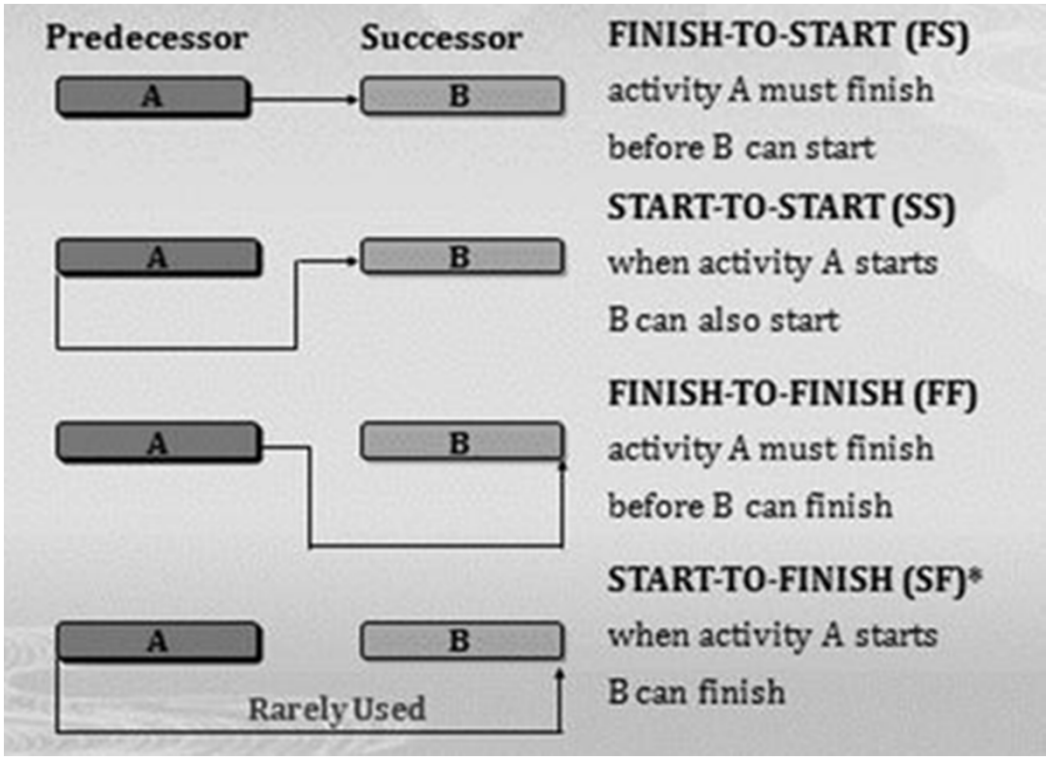
- It uses boxes or rectangle with activity name and arrows showing dependency between activities.
- Connects with arrows that show the dependencies
- Method used in Critical Path Methodology(CPM)
- No dummy activities
- Also called activity-on-node (AON)
- Graphical evaluation and review technique (GERT). It allows conditions, branches and loops.

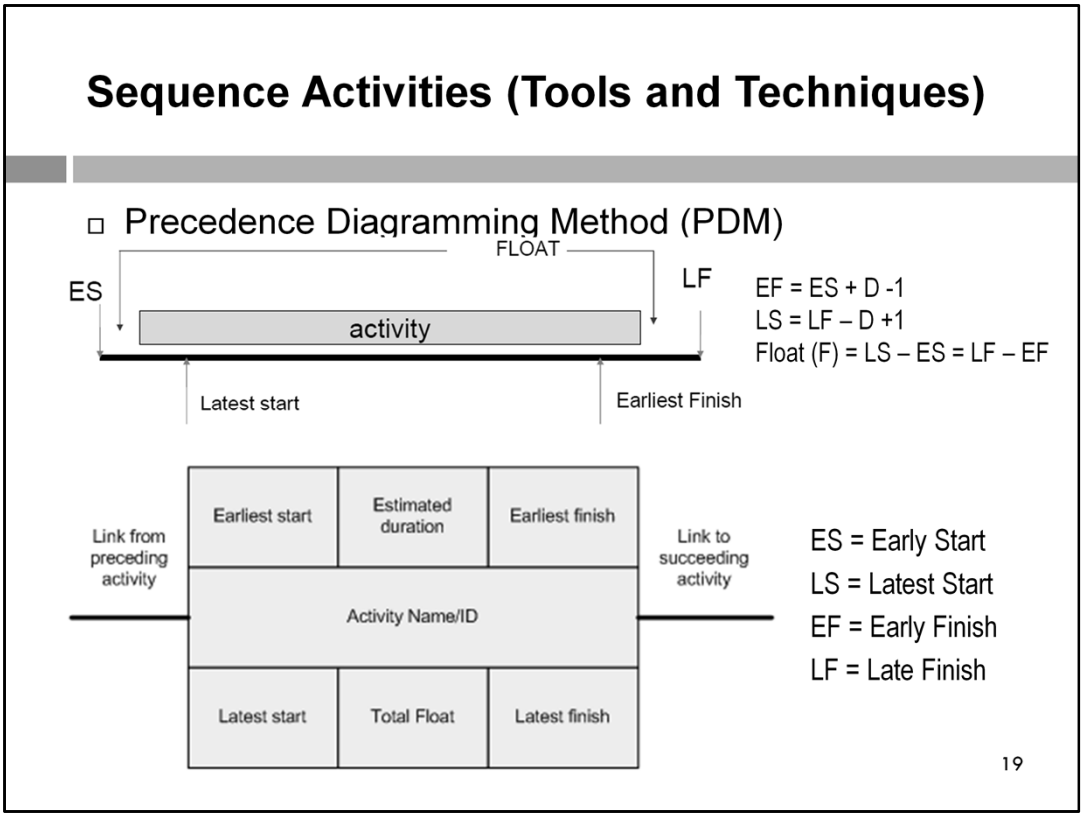
Sequence Activities (Tools and Techniques)

- Precedence Diagramming Method (PDM)
  - Logical Relationship
    - Finish-to-Start (FS)
      - A must finish before B can start
    - Finish-to-Finish (FF)
      - A must finish before B can finish
    - Start-to-Start (SS)
      - When A start, B can also start
    - Start-to-Finish (SF)
      - When A starts, B can finish



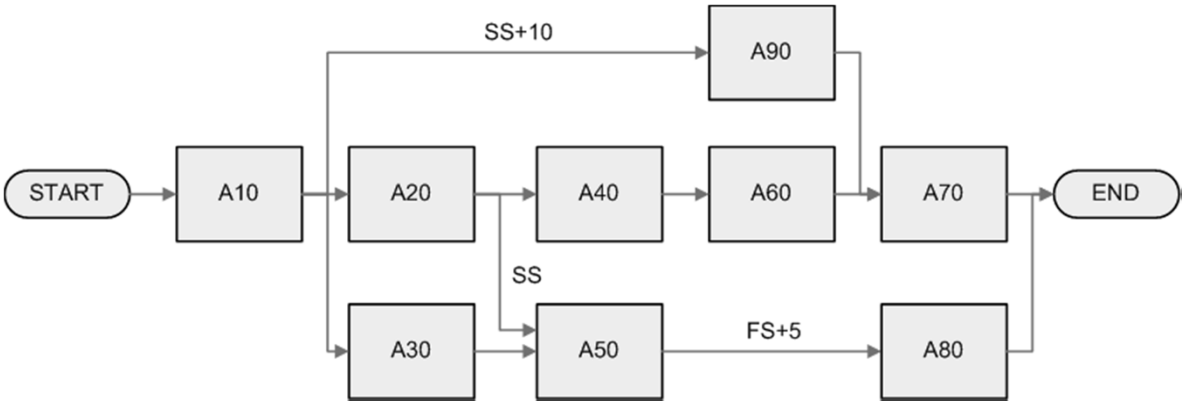
U08784 @ Peter Lo 2012





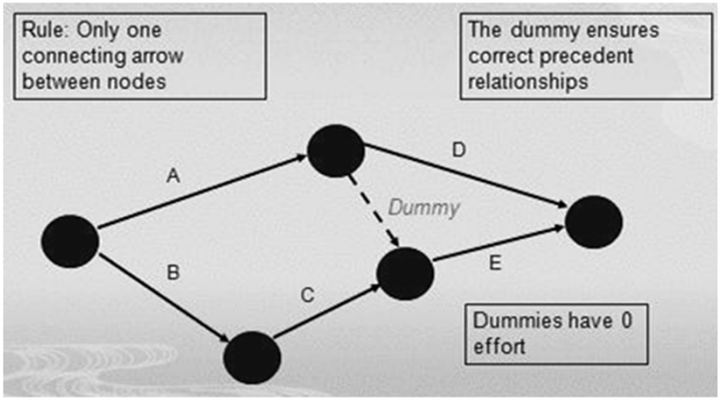
Activity 'write report software'
- Earliest start (ES)
- Earliest finish (EF) = ES + duration
- Latest finish (LF) = latest task can be completed without affecting project end
- Latest start = LF - duration

Example of PDM which showing logical relationship and leads or lags



Sequence Activities (Tools and Techniques)

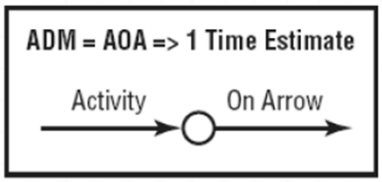
- Arrow Diagramming Method (ADM)
  - Also called (AOA)
  - Uses only Finish to Start (FS), sometimes dummy activities must be plugged to display dependencies.

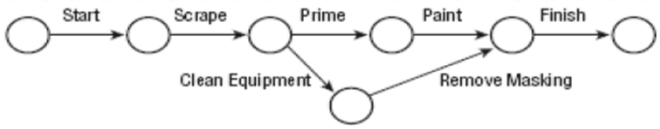


U08784 @ Peter Lo 2012

The Arrow Diagramming Method (ADM) is a method for drawing CPM and PERT network diagrams using arrows to represent activities and nodes show relationships (dependencies).

ADM =Arrow diagramming Method = Activity on Arrow=AOA





Sequence Activities (Tools and Techniques)

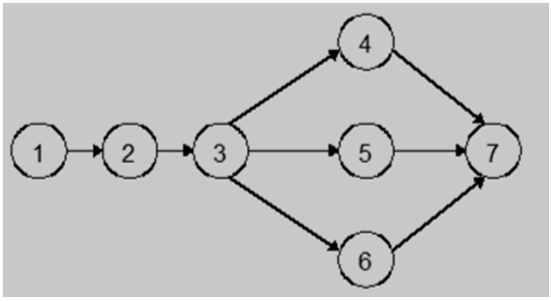
- Comparing PDM and ADM

Tools	Output
Precedence Diagramming Method (PDM)	Activity Node Diagram (AON)
Arrow Diagramming Method (ADM)	Activity on Arrow Diagram (AOA)

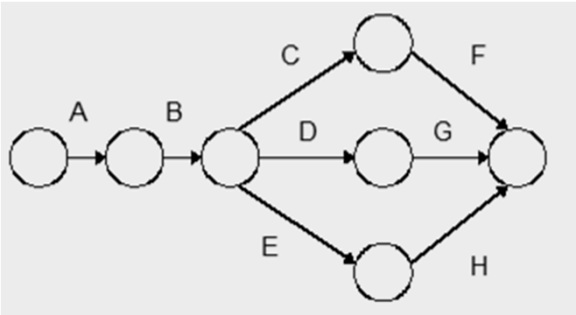
Network diagram
- ≠ PERT chart
- Shows just dependencies (logical relationship)
- Could show the critical path if activity duration estimates added

U08784 @ Peter Lo 2012 21

Precedence Diagramming Method (PDM) is also sometimes called Activity-on-Node (AOD) and sometime called Task on Node.



Arrow Diagramming Method (ADM) using Activity-on-Arrow. Sometimes called Task on Arrow.



Although these to figures look the same they are very different projects with a different number of tasks. According to PMI, which one you use depends on your preference.

Sequence Activities (Tools and Techniques)

- Dependency Determination
  - Mandatory Dependencies (Hard Logic)
    - By contract or inherent
    - Inherent in the nature of work being done or required by the contract
  - Discretionary Dependencies (Preferred Logic, Preferential Logic, or Soft Logic based on Best Practices)
    - Define base on knowledge
    - Can be changed if needed
    - Important when how to shorten or re-sequence the project
  - External dependencies
    - Relationship between project and non project activities (out of project team's control)

U08784 @ Peter Lo 2012 22

To define sequence among activity, these type of dependency are used:

- Mandatory (Hard Logic)
  - Mandatory Dependencies are specific to the nature of the project work. Absolutely must happen in the described manner. Referred to as 'hard logic'
  - Example: You must file your tax return before you can receive your tax refund check. The IRS is not sending you a check without reviewing that return; You must design before you can develop
- Discretionary (preferred, preferential, or soft logic)
  - Discretionary dependencies are set by the project team based on their experience. Also referred to as soft logic, preferred logic, or preferential logic. Here there may be a preferred order of events that reduces cost, risk or time, but other orders can be acceptable to the quality of the product
  - Example: Refilling your gas tank when it hits the 1/4 full mark on the gas gage. If you do it at that point you will never run out of gas, but you could wait until later. You may run the risk of not being close to a gas station, or having to accept whatever price is at the closest station
- External
  - Based on the need of the party outside the project
  - External Dependencies link between the project activities and activities external to the project.
  - Example: The release date of a new government safety regulation related to your product might precede your definition of safety requirements. Before the OSHA Right to Know Laws came out, companies were righting Material Data Sheets into projects so as to be prepared for the time when they would be required.

Sequence Activities (Tools and Techniques)

- Applying Leads and Lags
  - Lead
    - Speedup Successor
    - Allow an acceleration of successor activity
  - Lag
    - Delays successor
    - Directs a delay in the successor activity

U08784 @ Peter Lo 2012 23

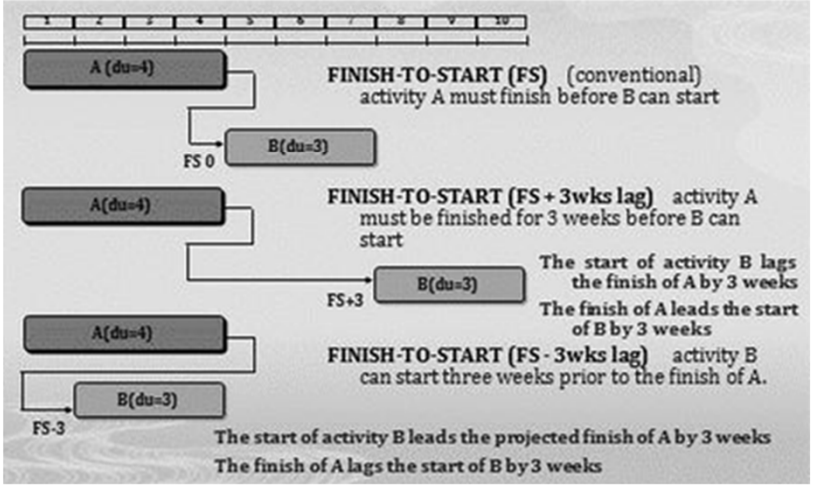
Leads and Lags
- Use leads and lags to support realistic and achievable project schedule.
- Each activity is connected at least to one predecessor and one successor except the start and the end.

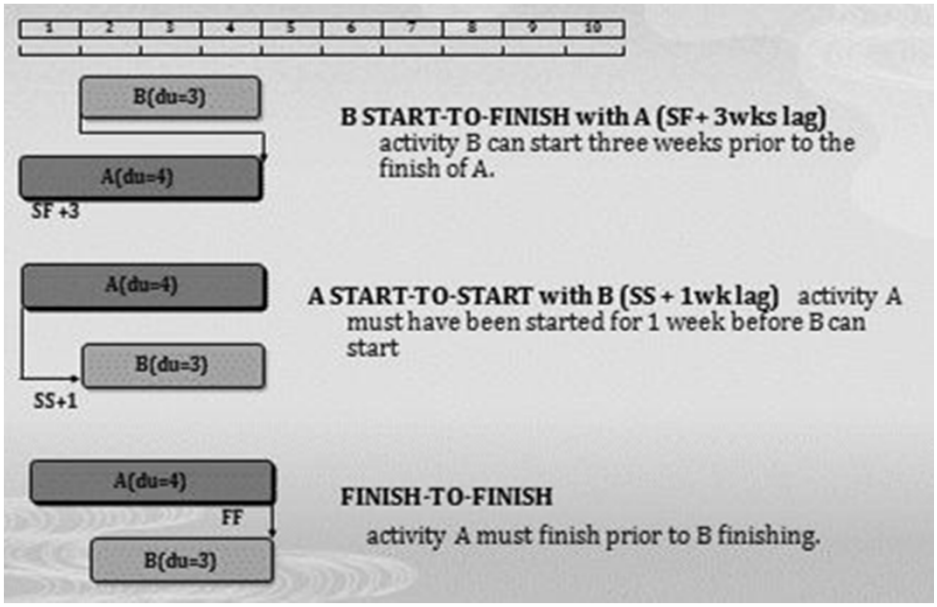
Lead
- Speedup Successor
- Allow an acceleration of successor activity

Lag
- Delays successor
- Directs a delay in the successor activity

Sequence Activities (Tools and Techniques)

- Applying Leads and Lags





Activity Sequencing (Tools and Techniques)

- Schedule Network Templates
  - Especially useful when a project includes several identical or nearly identical deliverables
  - Specially good for repeated deliverables
  - You can use a previous project schedule network diagram as a template for the current project.

U08784 @ Peter Lo 2012 25

Network Diagrams:

- Project Network Diagramming starts as a tool and output of Activity Sequencing. It is used to display and/or determine:
  - Activity dependencies (3.8.0.P)
  - Critical sequence of activities (3.8.0.P)
  - Length of schedule (3.10.0.P)
  - Float of activities (regular, free & negative) (3.11.0.P)
  - Early and late start and finish of activities (3.11.0.P)
- It is rarely done manually, but always used by scheduling software. The Network Diagrams are used to show activity dependencies and determine critical path. Three that are common, Activity-on-node, Activity-on-arrow, and logic bar charts (a Gantt chart that connects the tasks).

Bar Charts

- Bar Charts, also called Gantt charts, show start and end dates, expected duration, and progress-to-date.

Milestone Charts

- Milestone Charts identify scheduled start and end dates of major deliverables and key project interfaces. It is best used for presenting the project at a high-level.

Sequence Activities (Output)

- Project Schedule Network Diagrams
- Hammocks
- Project Document Updates

U08784 @ Peter Lo 2012 26

Project schedule network diagrams.

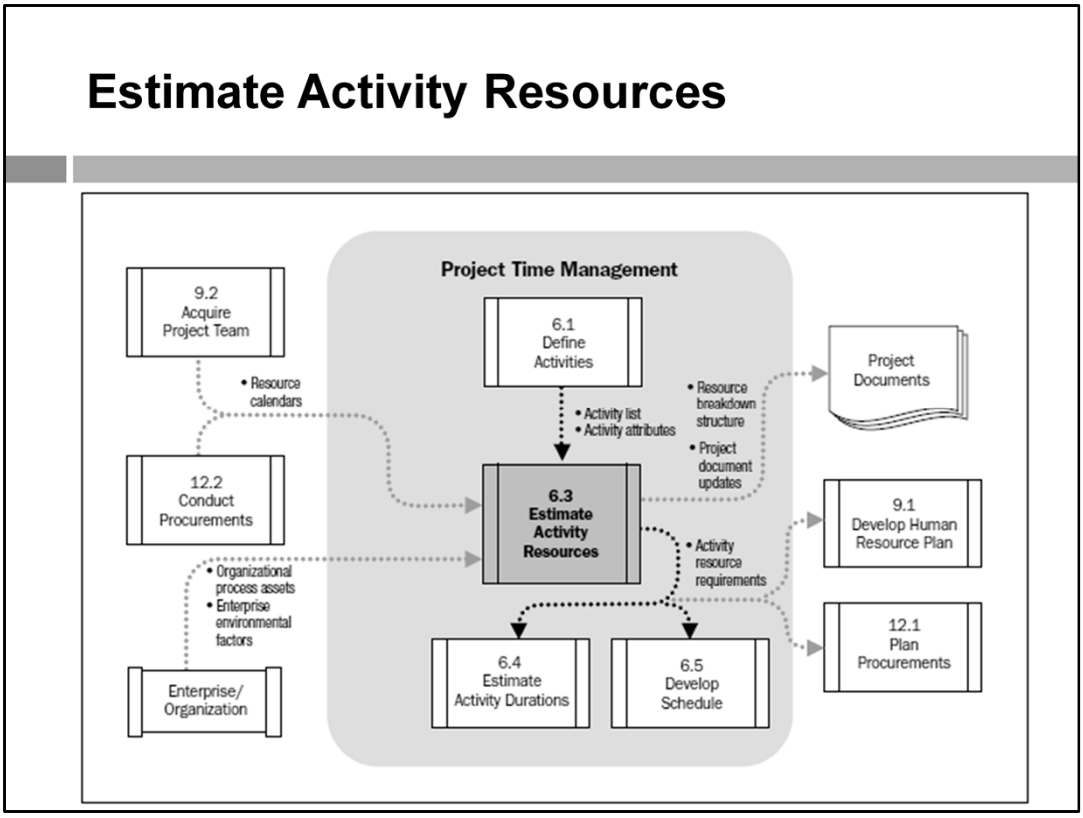
- Might contain project details or only summary level details, depending on the complexity of the project.

Hammocks.

- A group of related activities rolled up into a summary heading.

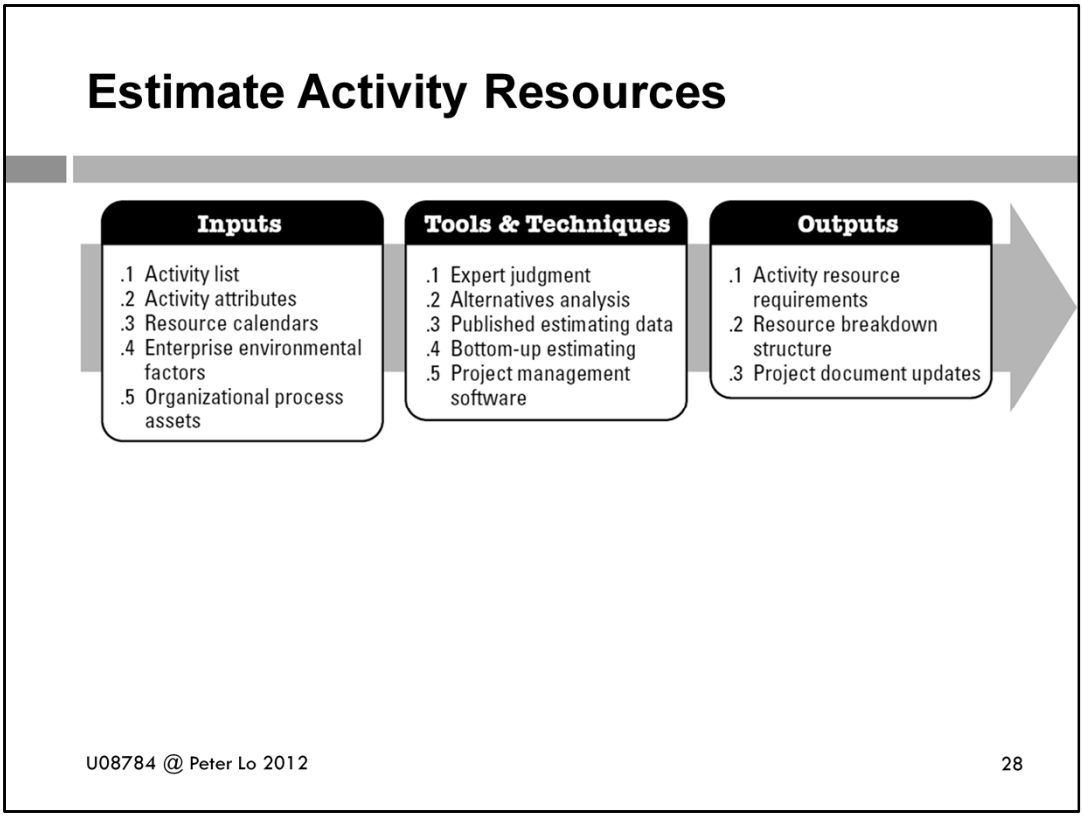
Project document updates.

- The construction of the network diagrams might bring activities to light that you missed when defining your activity list, or break 1 activity in 2, then you need to update your
  - Activity list
  - Activity attributes
  - Risk register



It's concerned with determining the types of resources needed (both human and material) and in what quantities for each schedule activity within a work package.

- The process of estimating the type and quantities of material, people, equipment or supplies
- Determine what resources, what quantities of each resource will be used
- Determine when each resource will be available
- Process of estimating the type and quantities of material, people, equipment or supplies required to perform each activity.



Inputs
- Activity List
- Activity Attributes
- Resource Calendars
- Enterprise Environmental Factors
- Organizational Process Assets

Tools & Techniques
- Expert Judgment
- Alternative Analysis
- Published Estimating Data
- Bottom-up Estimating
- Project Management Software

Outputs
- Activity Resource Requirements
- Resource Breakdown Structure (RBS)
- Project Document Updates

Estimate Activity Resource (Input)

- Activity List
- Activity Attributes
- Resource Calendars
- Enterprise Environmental Factors
- Organizational Process Assets

U08784 @ Peter Lo 2012 29

Activity List
- The activity list is a comprehensive list including all schedule activities required on the project.
- The activity list includes the activity identifier and a scope of work description for each activity in sufficient detail to ensure that project team members understand what work is required to be completed

Activity Attributes
- Activity attributes may describe a necessary sequence of events or defined predecessor or successor relationship.

Resource calendars
- When and how long a resources is available + its attributes (experience/skill, geographic location, i.e., availability, capability)
- Information (skill, location, etc.) in which resource (people, equipment, material, etc) are potentially available.
- It describes the time frames in which resources are available and their skills, abilities, quantity and availability.

Enterprise environmental factors
- resource availability and skills

Organizational process assets
- policies, procedures related to staffing, rental or purchase of equipment, historical information regarding types of resources

Estimate Activity Resource (Tools and Techniques)

- Expert Judgment
- Alternatives Analysis
  - Using various levels of resource capability or skills, type of machines, and make-or-buy decisions
- Published Estimating Data
- Bottom-up estimating
  - Work is decomposed into more detail when an activity cannot be estimated with a reasonable degree of confidence
  - Detailed piece of work are estimated, and aggregated
- Project Management Software

U08784 @ Peter Lo 2012 30

Expert judgment

Alternative analysis
- various level of resource capability, skills, size or type of machines, different tools, make or buy decisions
- It's used when thinking about the methods you might use to accomplish the activities your resources have been assigned.

Published estimating data:
- Use company's rates
- Organizational guidelines, industry rates or estimates, production rates and so on.

Bottom up estimating:
- Activity is decomposed to be more confidence in estimating
- Estimating individual activities or costs and adding them

Project management software
- Help to plan, organize and estimate resources needs and document their availability.

Estimate Activity Resource (Output)

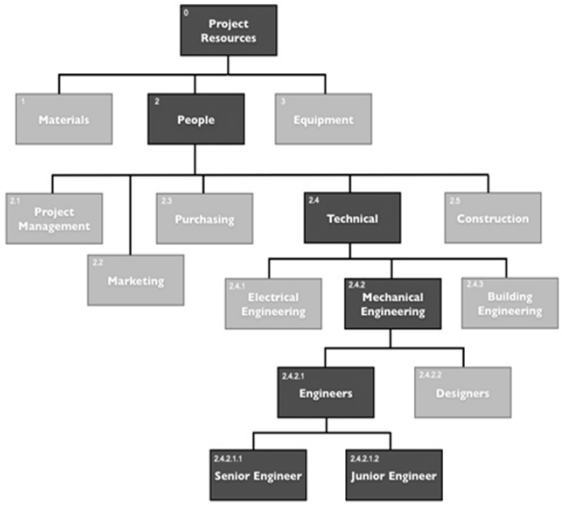
- Activities Resource Requirements
  - Document for each schedule activity can include the basis of estimate for each resource, and assumptions
- Resource Breakdown Structure (RBS)
  - Lists the resources by category and type.
  - Hierarchy of identified resources by resource category, and type (labor, material, equipment, and supplies)
- Project Document update

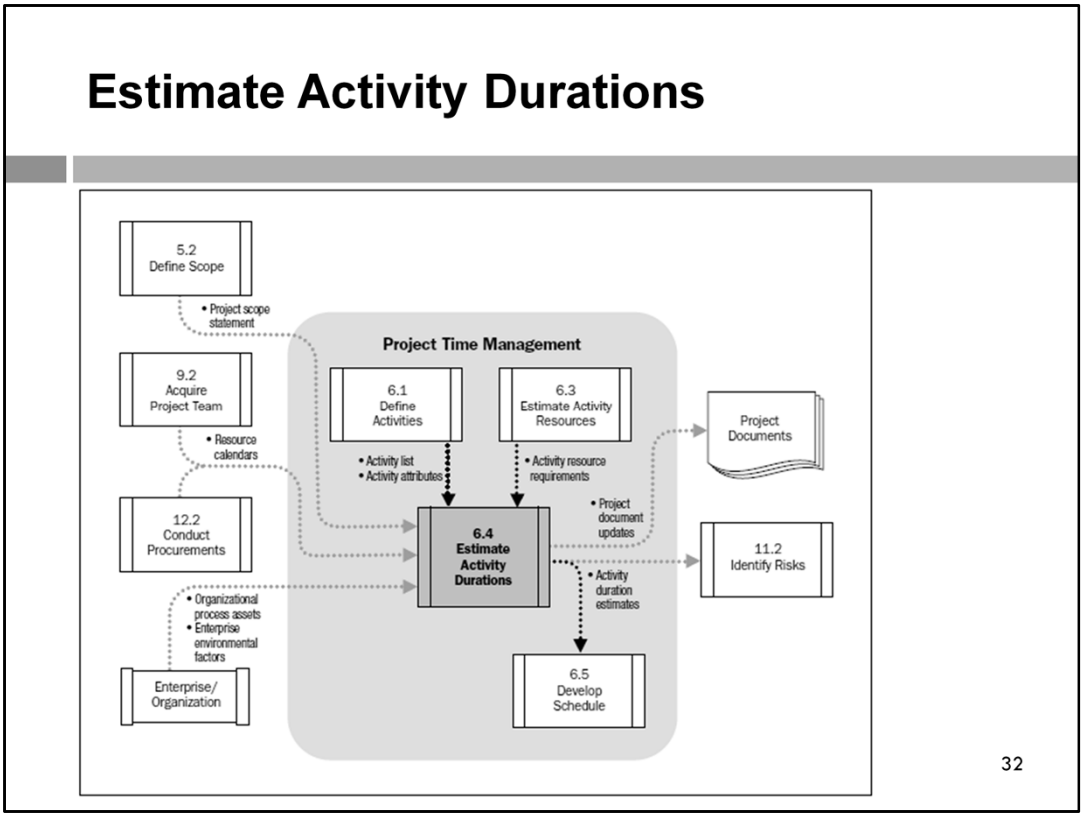
U08784 @ Peter Lo 2012 31

Activity resource requirements
- Types and quantities of resources
- Activity resource requirements.
- It's the purpose of this process, it describes the type of resources and the quantity needed for each activity.

Project documents updates
- Refers to updating the activity list, activity attributes and the resource calendars with changes to any of the elements recorded here.
  - Activity list
  - Activity attribute
  - Resource calendar (Identifies resource-specific holidays and resource availability periods)

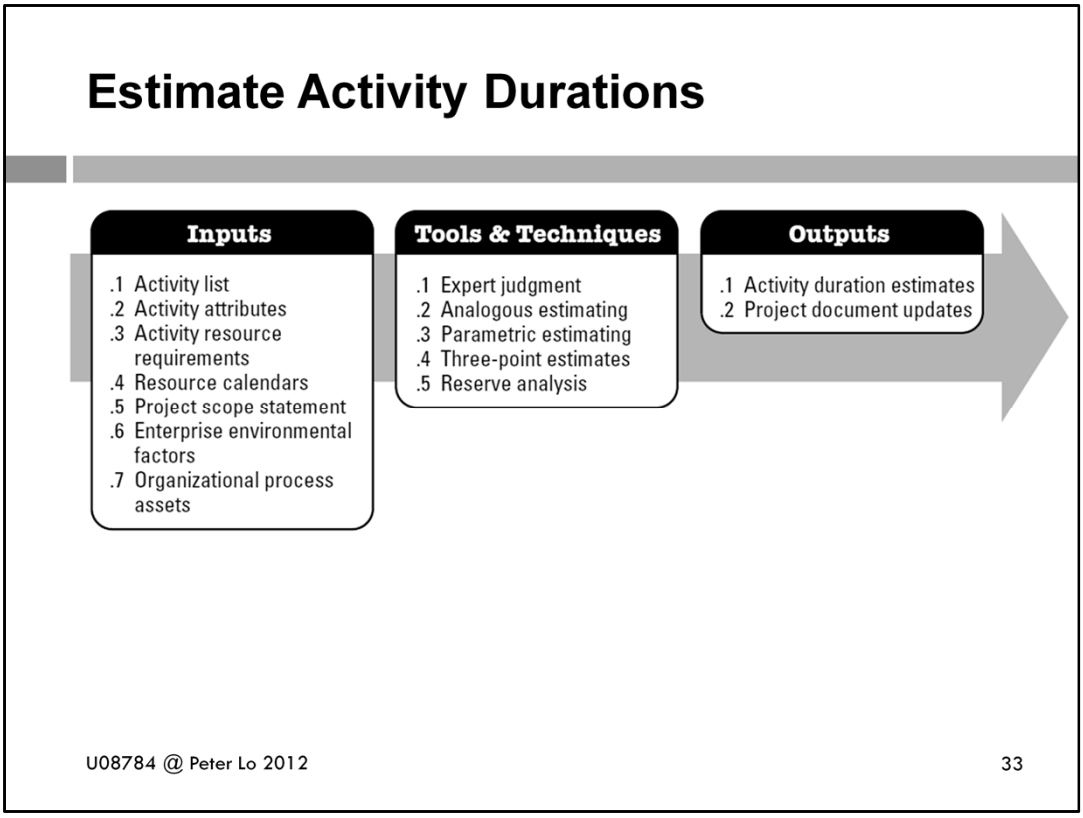
Resource breakdown structure





Attempts to estimate the work effort, resources, and number of work periods needed to complete each activity.

- Estimating the number of work periods needed to complete activities with estimated resources
- Progressively elaborated considering quality and availability of data
- Duration estimate is progressively elaborated
- Process of approximating the number of work periods to complete individual activities with estimated resources.
- Schedule shall be as believable and realistic as possible (do not allow padding)



Inputs

- Activities List
- Activities Attributes
- Activities Resource Requirements
- Resource Calendar
- Project Scope Statement
- Enterprise Environment Factors
- Organizational Process Assets

Tools and Techniques

- Expert Judgment
- Analogous estimating
- Parametric estimating
- Three point estimating
- Reserve analysis.

Outputs

- Activity duration estimates
- Project document update

Estimate Activity Durations (Input)

- Activity list
- Activity attributes
- Activity resource requirements
- Resources calendar
  - E.g. a senior staff member vs. junior staff member
- Project Scope Statement
  - Non-deliverable schedule activities that often have frequency and durations
- Enterprise environmental factor
  - Especially useful when activity durations are not driven by the actual work content (e.g. how long Government response a certain request)
- Organizational process assets
  - Records of previous project
  - Project calendar

U08784 @ Peter Lo 2012 34

- Activity list
- Activity attributes
- Activity resource requirements
- Resource calendars
  - Type, quantity, availability, capability of human resources, equipment, material
  - E.g. a senior staff member vs. junior staff member
- Project scope management
  - Non-deliverable schedule activities that often have frequency and durations
  - constraints and assumptions such as existing conditions, availability of information , and length of reporting periods, for example: available skilled resources, contract terms and requirements
- Enterprise environmental factors
  - Especially useful when activity durations are not driven by the actual work content (e.g. how long Government response a certain request)
  - Duration estimating DB, productivity metrics, published commercial information)
- Organizational process assets
  - Historical duration information, project calendars, scheduling methodology, and lesson learned
  - Project calendars and activity resource requirements are especially useful during this process.
  - Records of previous project

Estimate Activity Durations (Tools and Techniques)

- Expert judgment
  - Staff members who will perform activities.
- Analogous Estimating
  - Using actual cost of **previous similar projects** as basis for estimating the cost of current one
- Parametric Estimating
  - Technique using **statistical relationship** between historical data and other variables
  - Basis for activity durations can be determined by multiplying the quantity of work by the productivity rate
- Reserve Analysis
  - Incorporate additional time referred to as contingency reserves

U08784 @ Peter Lo 2012 35

Expert Judgment.
- Staff members who will perform activities.

Analogous Estimating (Top down):
- use actual duration of previous activity (historical) that has similarity
- Duration, budget, size, weight and complexity from a previous similar project
- Less costly and time consuming but less accurate
- Top down estimating, it's a form of Expert Judgment. Typically less time consuming and less costly than other estimating technique , but also less accurate

Parametric estimating (using statistical relationship between historical data and variables (e.g.: square footage)
- Multiplies the quantity of work by the rate. Highly accurate if the data is reliable.
- use statistical relationship between historical data and other variables (e.g. learning curve)
- The result can become heuristics (experience based technique/rule of thumb)

Reserve Analysis:
- Contingency reserves or time reserves or buffers) to account for schedule uncertainty shown by % of duration, fixed number of work periods. It should be clearly documented
- Reserve time, also called, buffers, time reserves, or contingency reserves.

Estimate Activity Durations (Tools and Techniques)

- Three-point Estimates
  - Use for time and cost estimation
  - Expected calculated from Most-likely, Optimistic, Pessimistic
  - Analytical technique average out three cost or duration estimates to represent **Optimistic**, **Most Likely** and **Pessimistic** scenarios
    - 3-Point Estimate = (O + P + 4 x ML) / 6

U08784 @ Peter Lo 2012 36

Three-point estimation (originated from Program Evaluation and Review Technique (PERT)

- Most likely, optimistic, pessimistic . This was originated with the PERT technique which uses the three estimates to define an approximate range for an activity's duration.
- Most likely (Tm), optimistic (To), pessimistic (Tp), expected (Te): Te=(To+4Tm+Tp)/6

Example

- A design company receives a work order to produce 500 clips arts. Provide an estimation by using parametric estimating and three-point estimate technique. Here are the fact:
  - The company has 4 junior designers and 4 senior designers
  - The production rate for senior designers is doubling the junior designer
  - According to the past project records, it took 2 senior designer and 1 junior designer one day to complete a similar project with 100 clip arts.
  - Staffs are also engaged into another project, so they may not be available.
  - Either 1,2 or 3 staffs can be allocated to the project daily, with average of 2 staffs.
  - The maximum project duration will be 10 days
- Calculate the project duration with:
  - Parametric estimation
  - Three-point estimation

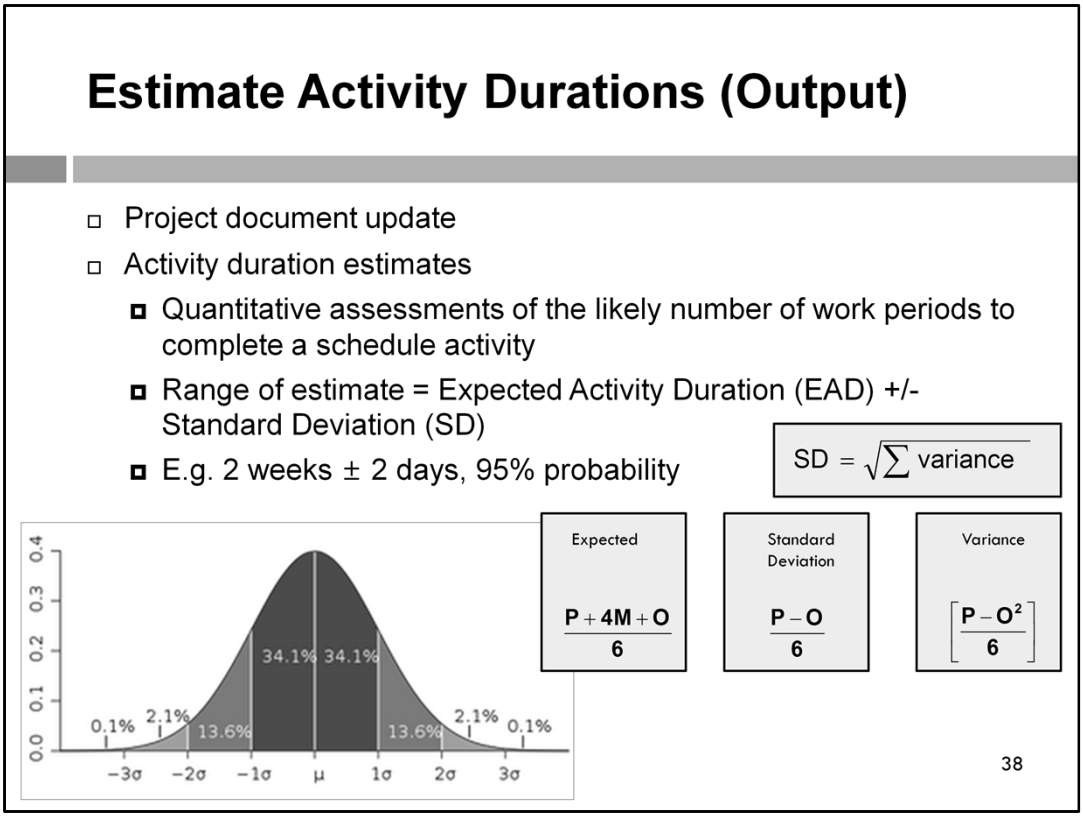
Let S = Productivity of Senior Designers, and j = Productivity of Junior Designer

Step 1: Calculate the productivity of the
S = 2j, S + 2j = 100 ➔ S = 40, j = 20

Step 2: Calculate the Optimistic, Most Likely and Pessimistic scenarios
P = 500 / (j x 1) = 500 / 20 = 25
O = 500 / (S x 3) = 500 / (40 x 3) = 4.17
ML = 500 / (j + s) = 500 / (20 + 40) = 8.33

1) Parametric Estimation
2s + j can complete 100 clip arts for 1 days.
So the project duration is 5 days

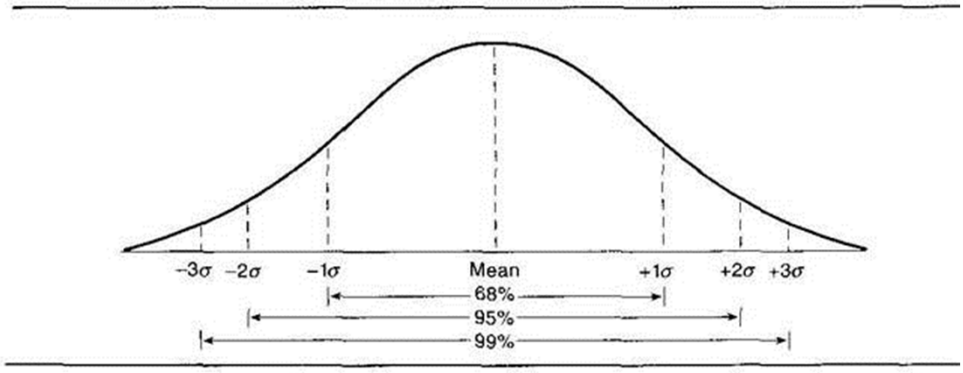
2) Three-Point Estimation
3-Point Estimation = (25 + 4.17 + 4 x 8.33) / 6 = 10.42.
Since the maximum project duration is 10, so the project duration is 10.

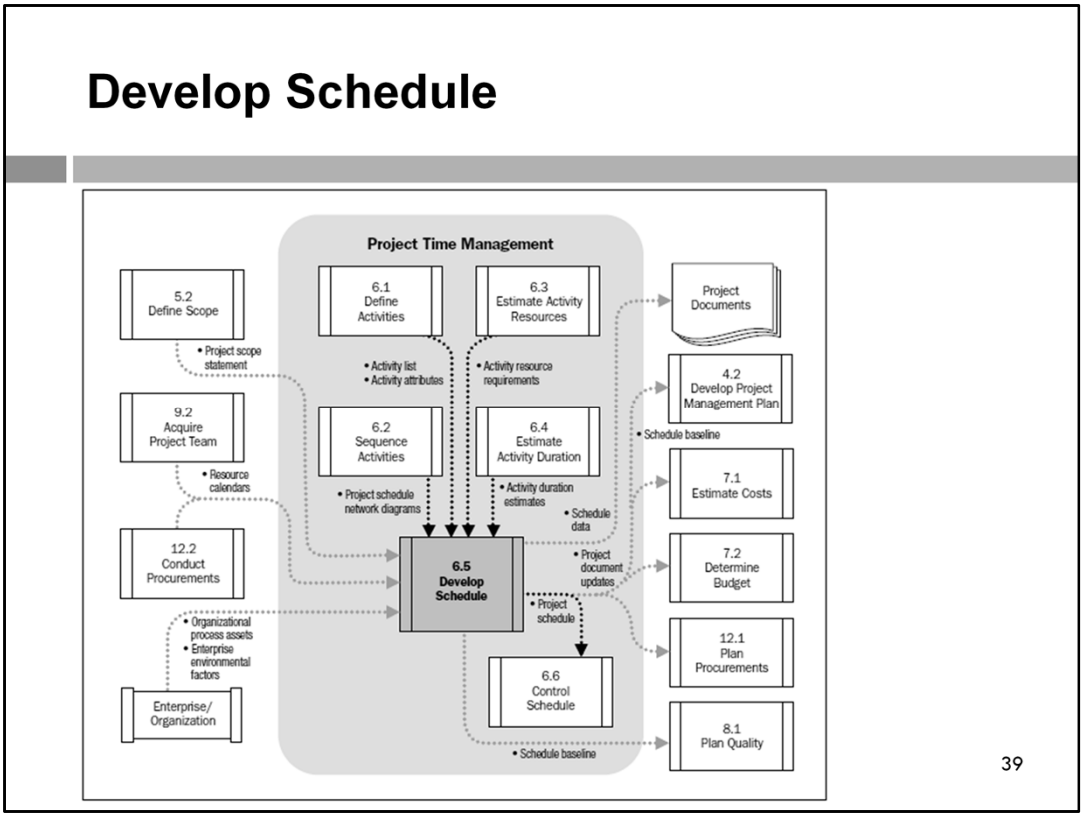


Project document updates
- Activity attributes
- Assumption made in developing the activity duration estimate such as skill levels and availability

Activity duration estimates
- Do not include any lag
- Are an estimate of the required work periods need to complete the activity usually expresses in hours, weeks, days or months. Final estimates should contain a range of
- For examples:
  - 2 weeks ± 2 days, 15% probability of exceeding 3 weeks
  - 100 hours+-10, or percentages.

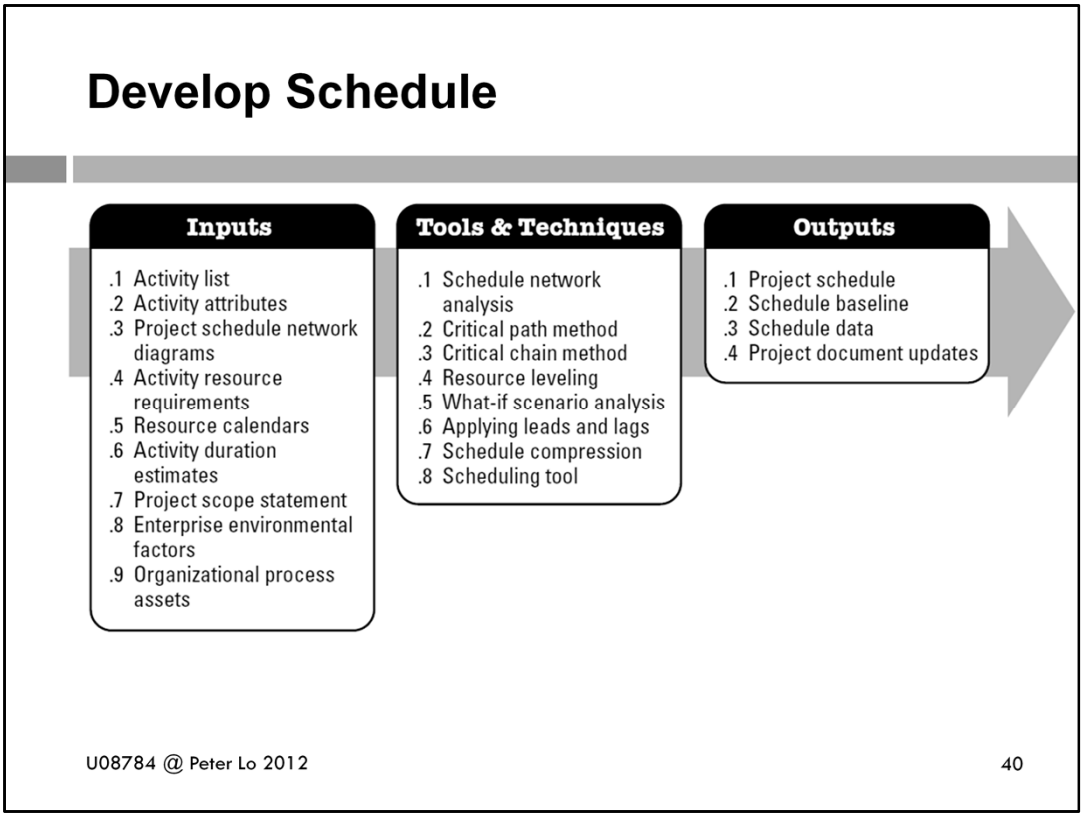
FIGURE 10.24 **Normal Probability Distribution**





The hearth of the planning process group, once it's approved, it serves as the schedule baseline for the project that you can track against in later processes.

- The process of analyzing activity sequence, durations, resource requirements and schedule constraints to create the project schedule
- Determine the start and finish dates for project activities
- Continues throughout the project as work progresses, the project management plan changes, and risk events
- Often an iterative process



Input
- Activity List
- Activity Attributes
- Project Schedule Network Diagram
- Activity Resource Requirements
- Resource Calendars
- Activity Duration Estimates
- Project Scope Statement
- Enterprise Environmental Factors
- Organizational Process Assets

Tools and Techniques
- Schedule Network Analysis
- Critical Path Method
- Critical Chain Method
- Resource Leveling
- What-if Scenario Analysis
- Applying Leads and Lags
- Schedule Compression
- Schedule Tool

Outputs
- Project Schedule
- Schedule Baseline
- Schedule Data
- Project Document Updates

Develop Schedule (Input)

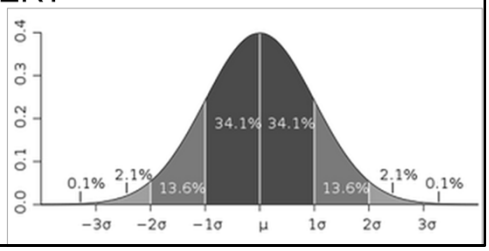
- Activity List
- Activity Attributes
- Project Schedule Network Diagram
- Activity Resource Requirements
- Resource Calendars
- Activity Duration Estimates
- Project Scope Statement
- Enterprise Environmental Factors
- Organizational Process Assets

U08784 @ Peter Lo 2012 41

Inputs
- Activity list
- Activity attributes
- Project schedule network diagrams
- Activity resource requirements
- Resource calendars
- Activity duration estimates
- Project scope statement
- Enterprise environmental factors (scheduling tool)
- Organizational process asset (scheduling methodology and project calendar)

Develop Schedule (Tools and Techniques)

- Schedule Network Analysis
  - It is a set of data that is researched that we seek to understand.
  - It is a technique that generates the project management schedule.
  - It is a detailed report of how and when you execute the next step of your project.
  - The first technique is called PERT



U08784 @ Peter Lo 2012

Schedule network analysis diagrams provide a graphical view of activities and how they relate to one another within a project management plan. Sometimes no matter how well you develop a project management schedule, something will happen which may dramatically change it. Identification and managing such events or event chains is why you would create network diagrams. You want to be able to see the problem and fix it as soon as it starts.

The schedule network analysis typically will include all elements from the pre-planning stages through all ongoing processes that may take place during the active period, and to any and all portions of the project from start to finish.

In essence, this method is used to help manage your money and time efficiently. A schedule network of analysis will always keep yon on your target goal.

There are many Schedule Network Analysis techniques. The first technique is called PERT (Program Evaluation and Review Technique). This is based on three time estimate for an activity. These three estimates are Optimistic estimate, Most Likely estimate and Pessimistic Estimate

Expected

$$\frac{P + 4M + O}{6}$$

Standard Deviation

$$\frac{P - O}{6}$$

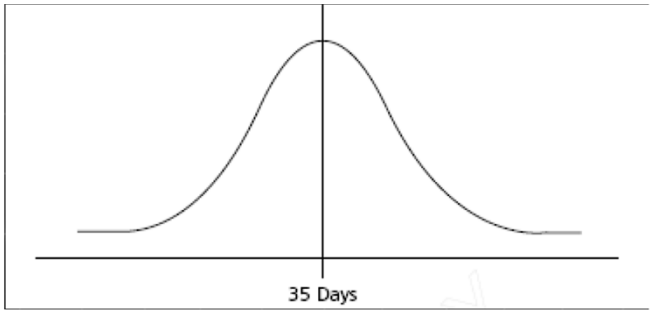
Variance

$$\left[\frac{P - O}{6}\right]^2$$

$$SD = \sqrt{\sum \text{variance}}$$

Schedule Probability

- In a normal distribution, about 68% of the scores are within one standard deviation of the mean and about 95% of the scores are within two standard deviations of the mean.



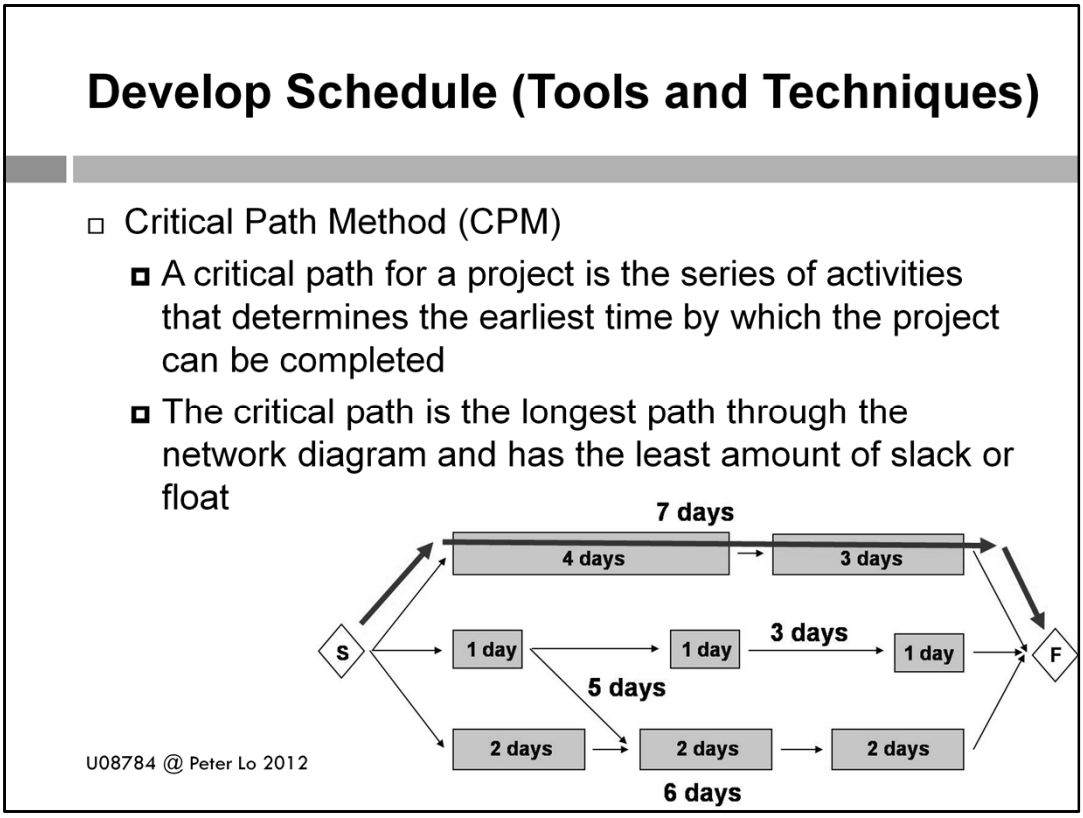
U08784 @ Peter Lo 2012 43

Optimistic time (O): the minimum possible time required to accomplish a task, assuming everything proceeds better than is normally expected
Pessimistic time (P): the maximum possible time required to accomplish a task, assuming everything goes wrong (but excluding major catastrophes).
Most likely time (M): the best estimate of the time required to accomplish a task, assuming everything proceeds as normal.
Expected time (TE): the best estimate of the time required to accomplish a task, assuming everything proceeds as normal (the implication being that the expected time is the average time the task would require if the task were repeated on a number of occasions over an extended period of time).

- TE = (O + 4M + P) ÷ 6
- Standard deviations = (P-0) ÷ 6

There are 95% chance that the project will be finish by : 94 ± (1.76 x 2) days

Act	Description	Optimistic	Pessimistic	Most Likely	EV	SD	Variance	CP EV	CP Variance
1	Develop project deliverables	13	16	15	14.83	0.50	0.2500	14.83	0.2500
2	Approval from stakeholders	4	6	5	5.00	0.33	0.1111	5.00	0.1111
3	Site selection	4	4	4	4.00	0.00	0.0000		
4	Evaluate and select vendor	4	5	4	4.17	0.17	0.0278		
5	Purchase hardware	3	3	3	3.00	0.00	0.0000		
6	Design software	14	17	15	15.17	0.50	0.2500	15.17	0.2500
7	Write code	24	33	30	29.50	1.50	2.2500	29.50	2.2500
8	Test software	4	4	4	4.00	0.00	0.0000	4.00	0.0000
9	Test hardware	9	11	10	10.00	0.33	0.1111		
10	Integrate hardware and software	20	23	20	20.50	0.50	0.2500	20.50	0.2500
11	Install and final acceptance	5	5	5	5.00	0.00	0.0000	5.00	0.0000
					Sum =			94.00	3.1111
					sq. rt. var. = SD				1.763834



Critical Path Method (CPM), is a procedure for using network analysis to identify those tasks which are on the critical path: ie where any delay in the completion of these tasks will lengthen the project timescale, unless action is taken.
For all tasks off the critical path, a degree of tolerance is possible (E.g. late start, late completion, early start, etc.).
Network charts and CPM analysis used to be carried out by hand.
Software is now available which requires the user only to enter the tasks, duration of each task and dependencies upon other tasks; a network chart and CPM is then automatically created.

Used at project level
Requires
- Decomposition of project scope into activities
- Definition of logical sequence and relationships between activities
- Estimation of activities time

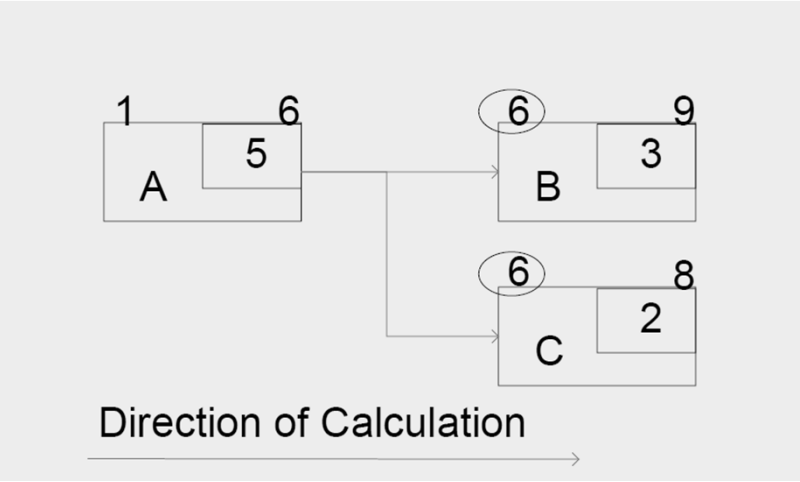
Determine series defining
- Minimum project duration (early finish)
- Longest path (late finish)

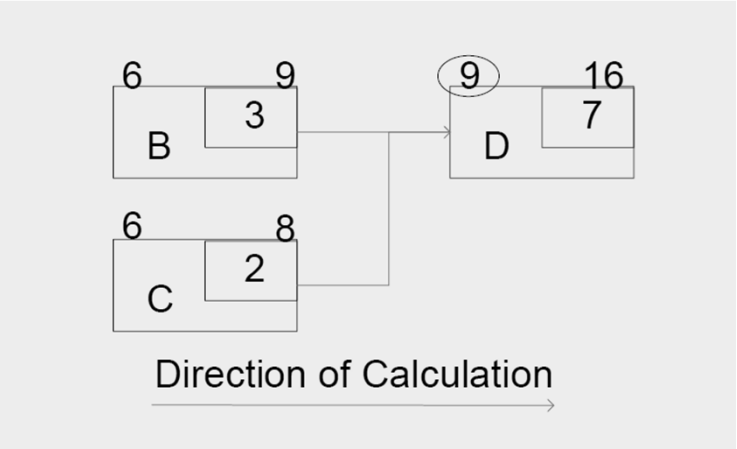
Calculate Float
- Determines least schedule flexibility
- Zero float mean activity is on critical

CPM – Network Calculation

- **Forward Pass** through network determines **early start** for each activity
  - Early start defines the earliest an activity can start based on the defined schedule logic
    - (ES + Duration) $_{\text{1st activity}}$ = (EF) $_{\text{1st activity}}$
    - (ES + Duration) $_{\text{1st activity}}$ = (ES) $_{\text{Subsequent activity}}$

ES = Earliest Start
EF = Earliest Finish

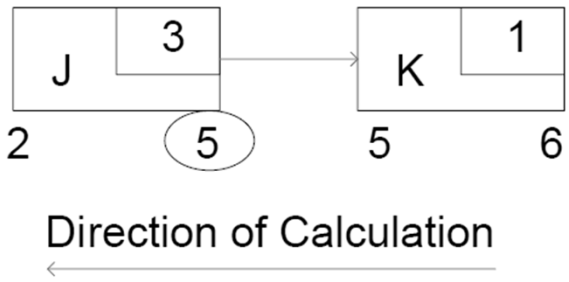


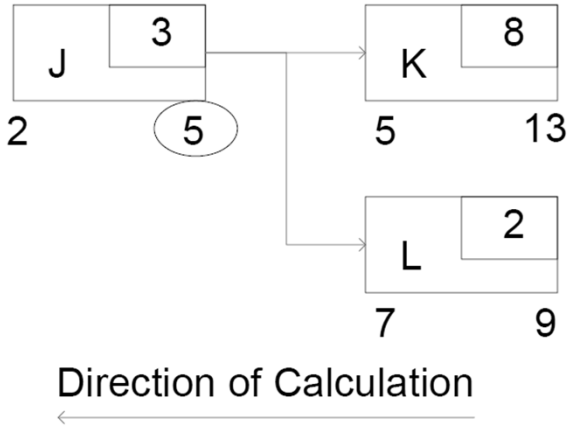


CPM – Network Calculation

- **Backward Pass** through network determines the **last start** for each activity
  - Last start defines the latest an activity can start without delay the project completion date, based on the defined schedule logic
    - (LF - Duration) $_{\text{Last activity}}$ = (LS) $_{\text{Last activity}}$
    - (LF - Duration) $_{\text{Last activity}}$ = (LF) $_{\text{Prior activity}}$

      LF = Late Finish
      LS = Late Start

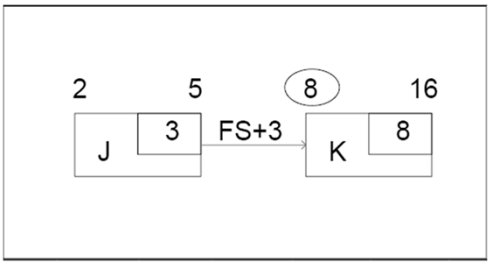


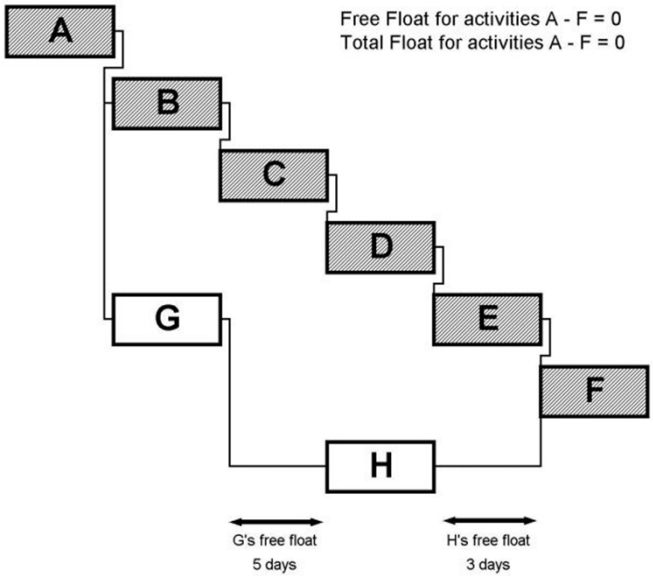


CPM – Network Calculation

- Float or Slack
  - Float = Late Finish – Early Finish
  - Float = Late Start – Early Start

  When
    - Float is > 0, time is available
    - Float = 0, on critical path
    - Float < 0, schedule can not be achievable

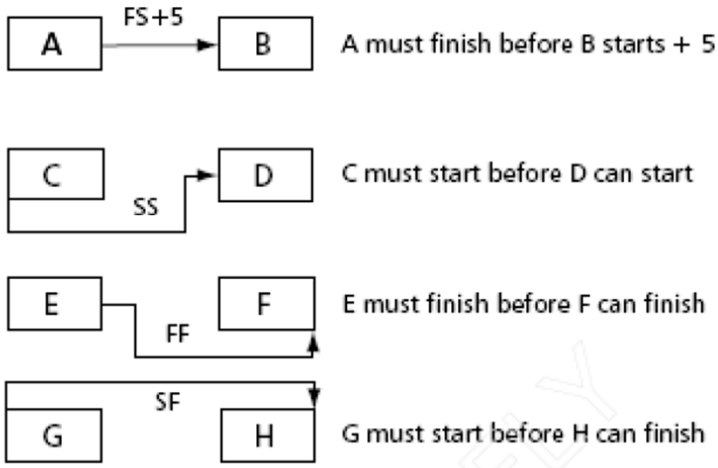




G's Free Float = 5 days
G's Total Float = 8 days

H's Free Float = 3 days
H's Total Float = 3 days

CPM - Precedence Relationships



A must finish before B starts + 5 days

C must start before D can start

E must finish before F can finish

G must start before H can finish

Leads or lags add or subtract time to the arrowhead event.

CPM vs. PERT

- Critical Path Method (CPM)
  - Assumes that both the activity estimates and the precedence logic are fixed. Concentrates on Time/Cost trade-off .
- Project Evaluation & Review Technique (PERT)
  - Allows for probabilistic treatment of the activity duration estimates by using a three-point estimate (pessimist, most likely, optimist), but assumes network logic is fixed.

Develop Schedule (Tools and Techniques)

- Critical Chain Method
  - Shifts focus from individual task completion to the all-important final project completion date
  - The buffer schedule activities are determined, the planned activities are scheduled to their latest possible planned start and finish dates

U08784 @ Peter Lo 2012 50

Critical Chain Method is a schedule network analysis technique that will modify the project schedule by accounting for limited or restricted resources (resource-constrained CPM + buffers at the beginning and end of scheduled activities)

The longest duration path through the project considering both activity dependencies and resource constraints.
Network diagram and critical path are identified first
Type of buffers
- Project buffer
- Feeding buffer
- Resource buffer

Critical Chain Scheduling
- Technique that addresses the challenge of meeting or beating project finish dates and an application of the Theory of Constraints (TOC)
- Developed by Eliyahu Goldratt in his books The Goal and Critical Chain
- A method of scheduling that takes limited resources into account when creating a project schedule and includes buffers to protect the project completion date
- Critical chain scheduling assumes resources do not multitask because it often delays task completions and increases total durations

Develop Schedule (Tools and Techniques)

- Resource Leveling
  - Trade-off between cost / schedule constraints and stable resources supply for the project
  - Allocate resource to critical path activities first

U08784 @ Peter Lo 2012 51

Resource Leveling also called the resource based method, is used when resources are over allocated, when they are time constrained or when you need to meet specific schedule dates and are concerned about resource availability.
Reverse Resource Allocation Scheduling is used when key resources are required at a specific point in the project and they're the only ones available.

Develop Schedule (Tools and Techniques)

- What-if Scenario Analysis
  - Uses different sets of activity assumptions to produce multiple project durations.
  - Analysis on effect of changes on a particular thing (assumption) on the project which make activity duration change.

U08784 @ Peter Lo 2012 52

Monte Carlo Simulation

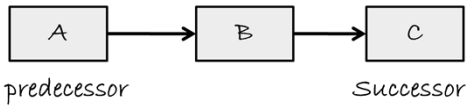
Used when there is **possibility that the critical path will be different** for a given set of project conditions.

Using probability distribution for each activity or group of activities

- Using computer software
- Using three-point estimates and network diagram
- Help deal with "**Path Convergence**"
  - Multiple paths converge into one or more activities (but adding risk)

Sequence Activities (Tools and Techniques)

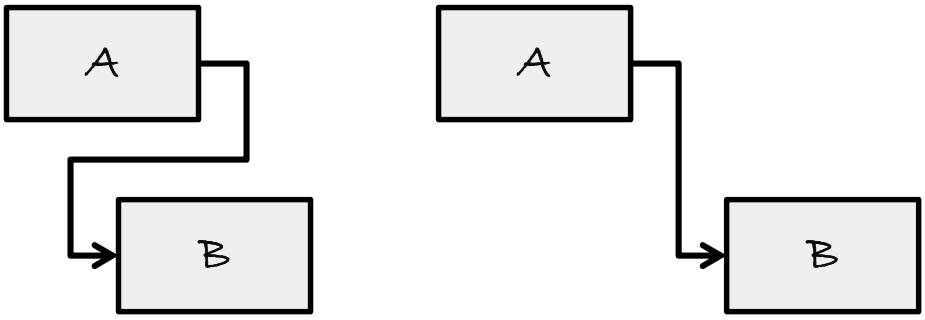
- Applying Leads and Lags
  - Use leads and lags to support realistic and achievable project schedule.
  - Each activity is connected at least to one predecessor and one successor except the start and the end.



  - Leads (Start before number of days) – May be added to start an activity before the predecessor activity is complete.
  - Lags (delay number of days) – Inserted waiting time between activities

Lead
- Speedup Successor
- Allow an acceleration of successor activity

Lag
- Delays successor
- Directs a delay in the successor activity



Develop Schedule (Tools and Techniques)

- Schedule Compression - Analysis used to shorten the project schedule without changing the project scope.
  - Crashing (approving overtime, adding resources for effort driven activities)
    - Cost /schedule trade-off analysis
    - Increase cost
  - Fast tracking (overlapping the activities)
    - Performing two tasks in parallel that were previously scheduled to start sequentially.
    - Increased risk

U08784 @ Peter Lo 2012 54

Fast Tracking

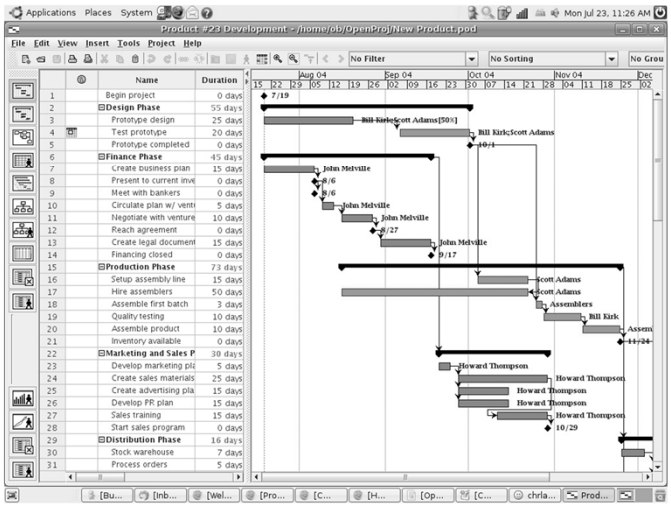
- Performing two tasks in parallel that were previously scheduled to start sequentially.
- Performing critical path activities in **parallel.**
- Usually **increase risk** and requires **more attention to communication.**
- May need a **rework**.
- E.g. Design is half finished and start coding.

Crashing

- A technique in which cost and schedule tradeoffs are analyzed to determine how to obtain the greatest amount of compression for the least incremental cost.
- Analyze cost and schedule trade-offs.
- Determine most compression for least cost.
- Crash the tasks that cost the least first, focusing on minimizing project cost.
- Always results in **increased cost**.

Develop Schedule (Tools and Techniques)

- Schedule Tool



Develop Schedule (Output)

- Project Schedule
  - Details the start and finish dates and resource assignments.

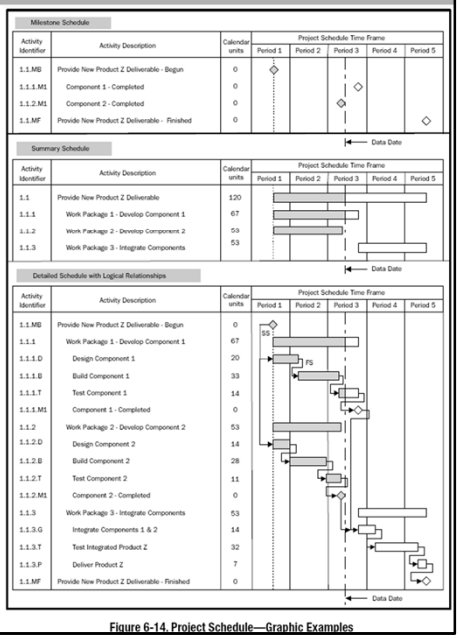


Figure 6-14. Project Schedule—Graphic Examples

U08784 @ Peter Lo 2012

Project schedule - Schedule can be shown with or without dependencies (logical relationship).

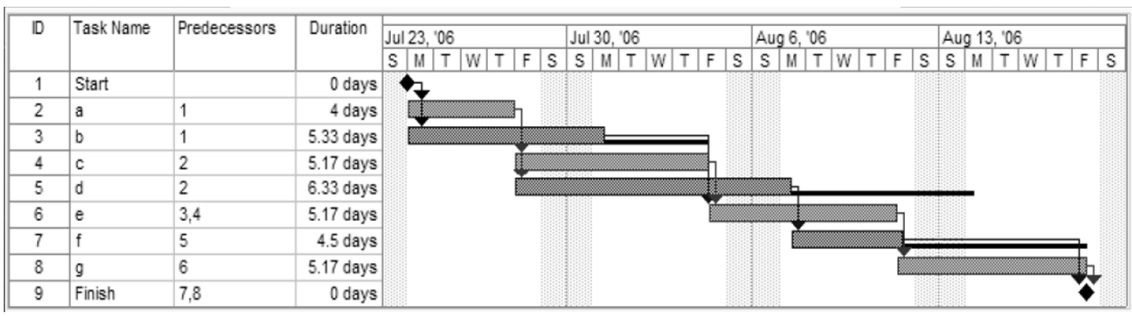
- Milestone charts
- Bar charts
- Project schedule network diagrams

Presented as

- Summary form e.g. Master Schedule, Milestone schedule
- Detailed form

Format:

- Network diagram
- Milestone chart
- Bar chart (Gantt chart)



ID	Task Name	Predecessors	Duration
1	Start		0 days
2	a	1	4 days
3	b	1	5.33 days
4	c	2	5.17 days
5	d	2	6.33 days
6	e	3,4	5.17 days
7	f	5	4.5 days
8	g	6	5.17 days
9	Finish	7,8	0 days

Develop Schedule (Output)

- Schedule Baseline
  - Final approved version of the project schedule.

Project Status Report

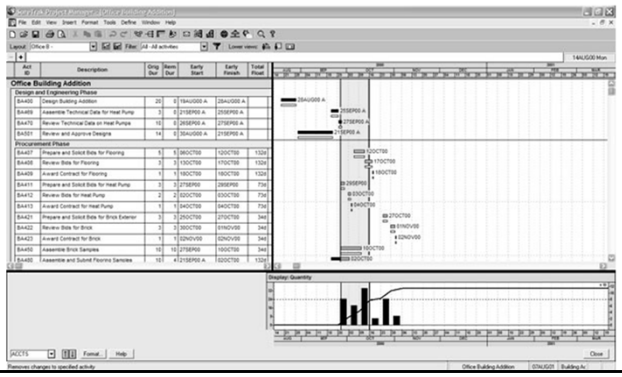
Page 1 of 2 9/29/07

WBS	Task Name	'98 '99 '00	Duration	Remaining Duration	Remaining Duration %	Cost	Remaining Cost	Remaining Cost %	Resource Names
1	PROJECT MANAGEMENT	0%	135 dis/s	135 dis/s	0	$72,360.00	$72,360.00	0	
1.1	PLANNING	0%	135 dis/s	135 dis/s	0	$72,360.00	$72,360.00	0	John,Bill,Hanne
2	SYSTEM ENGINEERING	0%	305 dis/s	305 dis/s	0	$163,480.00	$163,480.00	0	
2.1	REQUIREMENTS	0%	130 dis/s	130 dis/s	0	$69,680.00	$69,680.00	0	John,Bill,Hanne
2.2	ANALYSIS	0%	40 dis/s	40 dis/s	0	$21,440.00	$21,440.00	0	John,Bill,Hanne
2.3	INTEGRATION	0%	45 dis/s	45 dis/s	0	$24,120.00	$24,120.00	0	John,Bill,Hanne
2.4	MISSION OPERATIONS	0%	30 dis/s	30 dis/s	0	$16,080.00	$16,080.00	0	John,Bill,Hanne
2.5	SYSTEM TEST PLANNING	0%	60 dis/s	60 dis/s	0	$32,160.00	$32,160.00	0	John,Bill,Hanne
2.6	System Engineering Complete		0 dis/s	0 dis/s	0	$0.00	$0.00	0	John,Bill,Hanne
3	SPACECRAFT ENGINEERING	0%	732 dis/s	732 dis/s	0	$386,992.00	$386,992.00	0	
3.1	COMMAND AND CONTROL	0%	500 dis/s	500 dis/s	0	$268,000.00	$268,000.00	0	John,Bill,Hanne
3.2	ATTITUDE CONTROL	0%	132 dis/s	132 dis/s	0	$70,752.00	$70,752.00	0	John,Bill,Hanne

Summary Baseline — Summary — Current Plan — Baseline — Status

Develop Schedule (Output)

- Schedule Data
  - Documenting the supporting data for the schedule. Milestones, schedule activities and activity attributes, and the assumptions and constrains regarding the schedule.



U08784 @ Peter Lo 2012 58

Schedule data (milestones, activities, activity attributes, documented assumptions and constraints)

- Resource histogram
- Alternative schedules (best, worst case, resource-leveled, not resource-leveled, with or without imposed dates
- Scheduling of contingency reserves

Includes at least:

- Schedule milestone
- Schedule activities
- Activity attributes
- Assumptions & Constraints

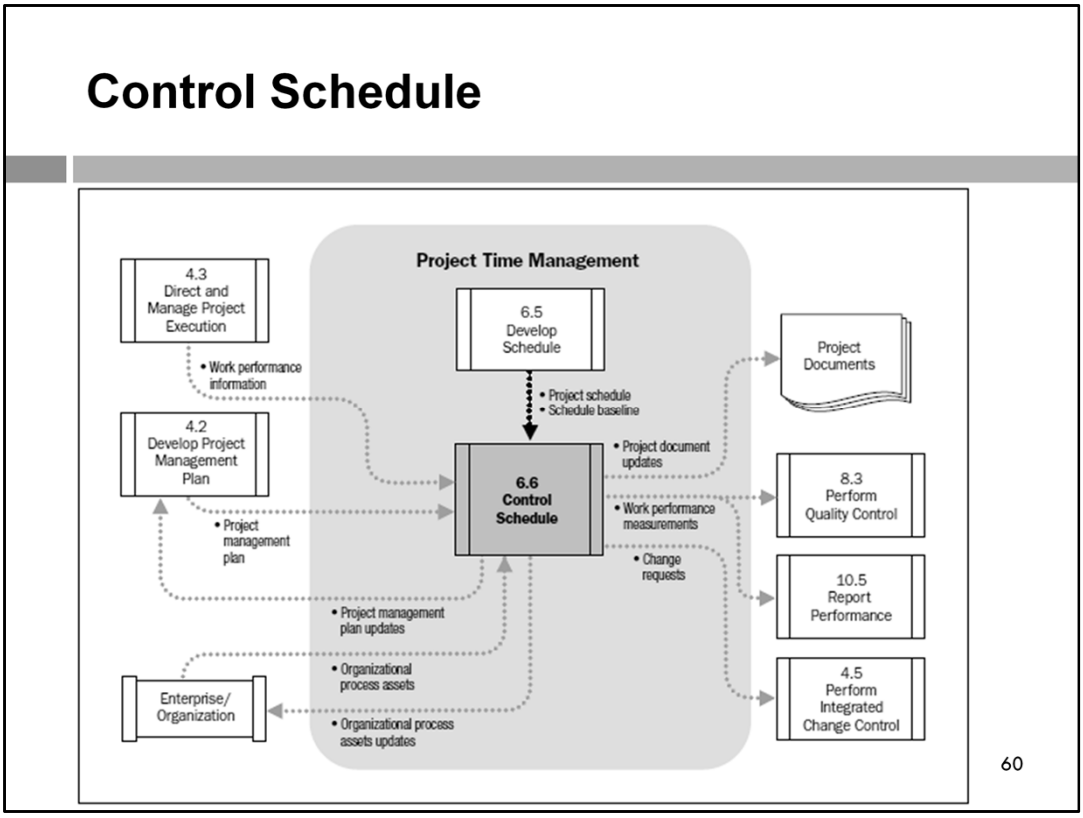
Additional information can be added, such as

- Resource histograms
- Cash-flow projections
- Order & delivery schedules
- Alternative schedules

Develop Schedule (Output)

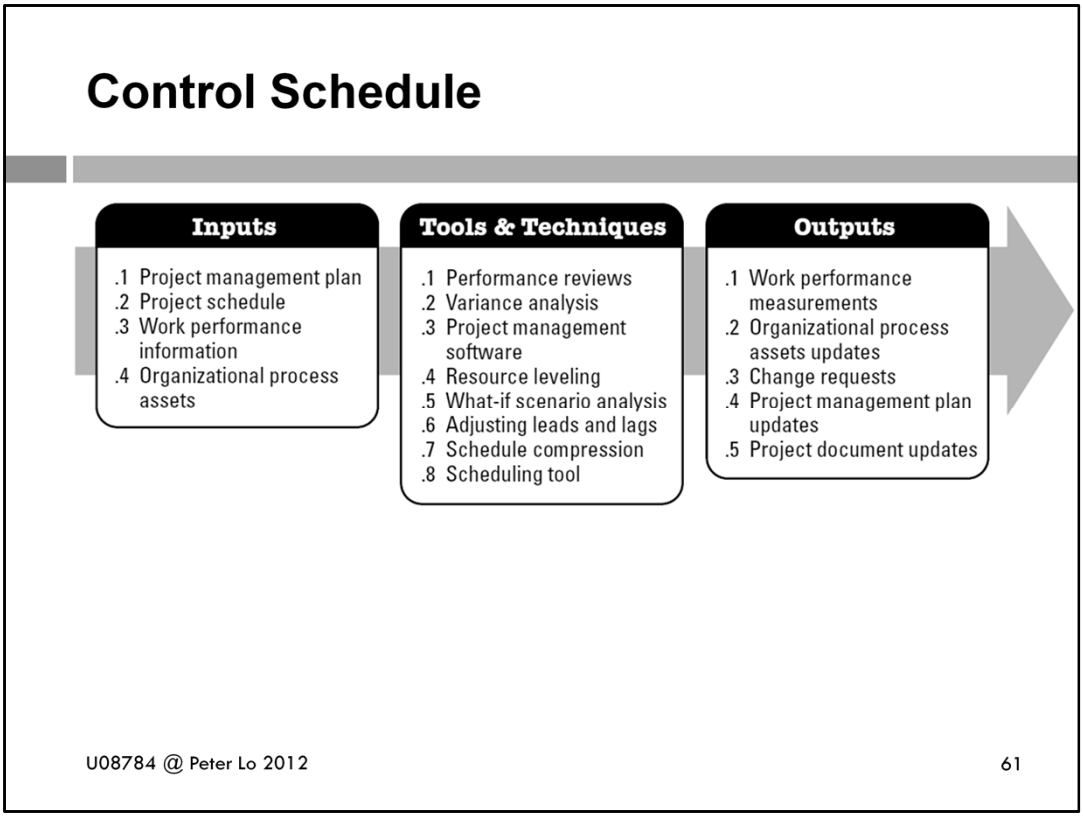
- Project Document Updates
  - Updates to the activity resource requirements documents, activity attributes, calendars and the risk register.

Project document updates
- Activity resource requirements
- Activity attributes
- Calendar
- Risk register



Process of analyzing activity sequences, durations, resource requirements, and schedule constraints to create the project schedule.

- The process of monitoring the status of the product to update project progress and manage changes to the schedule baseline
- Influencing the factors that create changes
- Determining the current status of the project schedule influencing the factors that create schedule changes
- Determining that the project schedule has changed
- Managing the actual changes as they occur
- Part of the Integrated Change Control process



Input
- Project Management Plan
- Project Schedule
- Work Performance Information
- Organizational Process Assets

Tools & Techniques
- Performance Reviews
- Variance Analysis
- Project Management Software
- Resource Leveling
- What-If Scenario Analysis
- Adjusting Leads and Lags
- Schedule Compression
- Scheduling Tool

Outputs
- Work Performance Measurements
- Organizational Process Assets Updates
- Change Requests
- Project Management Plan Updates
- Project Document Updates

Control Schedule (Input)

- Project Management Plan
- Project Schedule
- Work Performance Information
- Organizational Process Assets

U08784 @ Peter Lo 2012 62

Inputs

- Project management plan
- Project schedule
  - The most recent version with notations to indicate updates, completed activities and started activities as of the indicated data date
- Work performance information
  - which activities have started/finished, their progress
- Organizational process asset
  - Existing formal and informal schedule control-related policies, procedure and guidelines
  - Schedule control tools
  - Monitoring and reporting methods to be used

Control Schedule (Tools and Techniques)

- Performance Reviews
- Variance Analysis
- Project Management Software
- Resource Leveling
- What-If Scenario Analysis
- Adjusting Leads and Lags
- Schedule Compression
- Scheduling Tool

U08784 @ Peter Lo 2012 63

Tools and Techniques

- Performance reviews (measure, compare and analysis of performance (start, finish, percent complete and remained)). In Earned)) Value Management (EVM), Schedule Variance (SV), and Schedule Performance Index (SPI) are used to assess the schedule variance
- Variance analysis (determining the cause and degree of variance WRT schedule baseline)
- Project management software (tracking planned dates versus actual dates, forecast the change effects)
- Resource leveling
- What-if scenario analysis (to bring the schedule into alignment with the plan)
- Adjusting leads and lags
- Schedule compression
- Scheduling tool (to update schedule data, and project schedule

Control Schedule (Output)

- Work Performance Measurements
- Organizational Process Assets Updates
- Change Requests
- Project Management Plan Updates
- Project Document Updates

U08784 @ Peter Lo 2012 64

Work performance measurements
- SV and SPI for WBS components, work packages are documented and communicated to stakeholders

Organizational process assets updates
- Causes of variances
- Corrective action chosen and the reasons
- Other types of lessons learned

Change request

Project management plan updates
- Schedule baseline
- Schedule management plan
- Cost baseline

Project document updates
- Schedule data
- Project schedule

Lines of Code

Model	Formula
Walston-Felix Model	$E = 5.2 * (KLOC)^{0.91}$
Bailey-Basili Model	$E = 5.5 + 0.73 * (KLOC)^{1.16}$
Boehm Simple Model	$E = 3.2 * (KLOC)^{1.05}$
Doty Model for KLOC > 9	$E = 5.288 * (KLOC)^{1.047}$
Albrecht & Gaffney Model	$E = -13.39 + 0.0545 * FP$
Kemerer Model	$E = 60.62 * 7.728 * 10^{-8} * FP^{3}$
Matson, Barnett & Mellichamp Model	$E = 585.7 + 15.12 * FP$

65

Before an estimate for software is made, it is important and necessary to understand software scope and estimate its size.
LOC is a direct approach method and requires a higher level of detail by means of decomposition and partitioning.

Lines of Code **(LOC)** method measures software and the process by which it is being developed.

- Productivity = KLOC / Person-month
- Quality = Defects / KLOC
- Cost = $ / LOC
- Documentation = pages of documentation / KLOC

Where,

- ***KLOC*** stand for no. of lines of code (in thousands).
- ***Person-month*** stand for is the time(in months) taken by developers to finish the product.
- ***Defects*** stand for Total Number of errors discovered

Function Point Analysis

- **Function Point Analysis (FPA)** is designed to estimate and measure the time, and thereby the cost, of developing new software applications and maintaining existing software applications.
- It is also useful in comparing and highlighting opportunities for productivity improvements in software development.
- It was developed by A.J. Albrecht of the IBM Corporation in the early 1980s.
- The main other approach used for measuring the size, and therefore the time required, of software project is Lines of Code (LOC) – which has a number of inherent problems.

U08784 @ Peter Lo 2012 66

Function Point Analysis has been proven as a reliable method for measuring the size of computer software. In addition to measuring output, Function Point Analysis is extremely useful in estimating projects, managing change of scope, measuring productivity, and communicating functional requirements.

There have been many misconceptions regarding the appropriateness of Function Point Analysis in evaluating emerging environments such as real time embedded code and Object Oriented programming. Since function points express the resulting work-product in terms of functionality as seen from the user's perspective, the tools and technologies used to deliver it are independent.

One of the initial design criteria for function points was to provide a mechanism that both software developers and users could utilize to define functional requirements. It was determined that the best way to gain an understanding of the users' needs was to approach their problem from the perspective of how they view the results an automated system produces. Therefore, one of the primary goals of Function Point Analysis is to evaluate a system's capabilities from a user's point of view. To achieve this goal, the analysis is based upon the various ways users interact with computerized systems.

Objectives of Function Point Analysis

- Function Point Analysis measures software by quantifying the functionality the software provides to the user based primarily on logical design.
- The objectives of function point analysis are:
  - Measure functionality that the user requests and receives
  - Measure software development and maintenance independently of technology used for implementation
- The process of counting function points should be:
  - Simple enough to minimize the overhead of the measurement process
  - A consistent measure among various projects and organizations

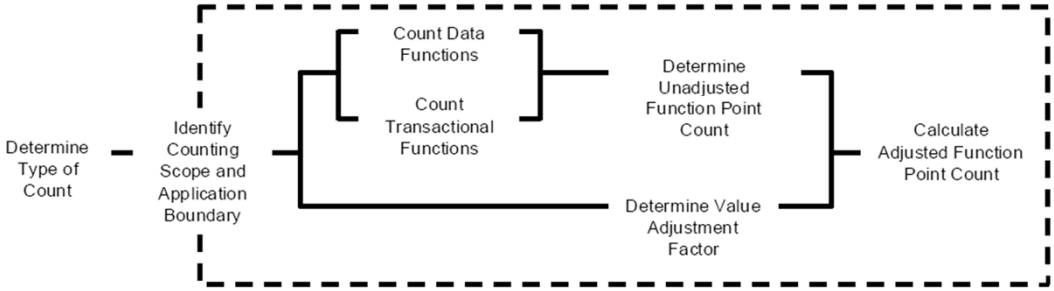
U08784 @ Peter Lo 2012 67

Benefits of Function Point Analysis

- Organizations can apply function point analysis as:
  - A tool to determine the size of a purchased application package by counting all the functions included in the package
  - A tool to help users determine the benefit of an application package to their organization by counting functions that specifically match their requirements
  - A tool to measure the units of a software product to support quality and productivity analysis
  - A vehicle to estimate cost and resources required for software development and maintenance
  - A normalization factor for software comparison

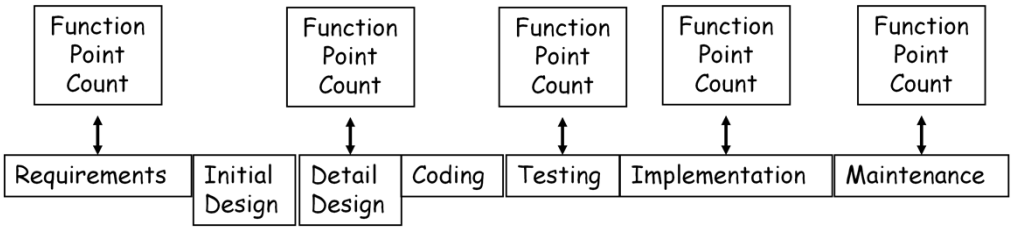
Function Point Counting Procedure

- This diagram presents the high-level procedure for function point counting.



Step 1: Determine the Type of Function Point Count

- Function point counts can be associated with either projects or applications.
- Three types of function point counts:
  - Development project function point count
  - Enhancement project function point count
  - Application function point count



Step 2: Identify the Counting Scope and Application Boundary

- The function point counting boundary indicates the border between the project or application being measured and the external applications or user domain.
- Rules:
  - The boundary is determined based on the user's point of view. The focus is on what the user can understand and describe.
  - The boundary between related applications is based on separate business functions as seen by the user, not on technological concerns.
  - For enhancement projects, the initial boundary must conform to the boundaries already established for the application or applications being modified.

U08784 @ Peter Lo 2012 70

The counting scope defines the functionality which will be included in a particular function point count.
The application boundary indicates the border between the software being measured and the user.
Purpose:

The scope of the product being measured
Data ownership required for function point counting (owned by one application and/or another)
Processing relationships required for function point counting (where processing occurs and purpose)..

Rules:

- The boundary is determined based on the user's point of view. The focus is on what the user can understand and describe.
- The boundary between related applications is based on separate business functions as seen by the user, not on technological concerns.
- For enhancement projects, the initial boundary must conform to the boundaries already established for the application or applications being modified.

Procedures:

- Identify the application or project boundary using the boundary rules..
- Document the following items:
  - The boundary identified
  - The purpose of the count
  - Any assumptions that the count is based on

Step 3: Define All Data Functions and Their Degree of Complexity

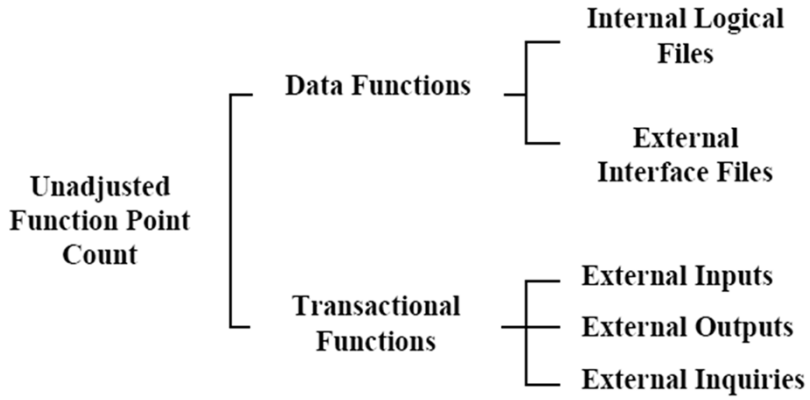
- Data functions represent the functionality provided to the user to meet internal and external data requirements.
- Data functions are either
  - Internal Logical Files (ILF)
  - External Interface Files (EIF)

Step 4: Define all Transactional Functions and Their Complexity

- Transactional functions represent the functionality provided to the user to process data.
- Transactional functions are either
  - External Inputs
  - External Outputs
  - External Inquiries

Step 5: Calculate the Unadjusted Function Point Count

- Unadjusted Function Point Count (UFPC) reflects the specific countable functionality provided to the user by the project or application.



U08784 @ Peter Lo 2012 73

The application's specific user functionality is evaluated in terms of what is delivered by the application, not how it is delivered. Only user-requested and defined components are counted.

The unadjusted function point count has two function types—data and transactional. These function types are further defined as shown in the following diagram

Step 6: Calculate the VAF Based on a Set of General System Characteristics

- The Value Adjustment Factor (VAF) is multiplied by the Unadjusted Function Point (UAF) to come up with a Final Adjusted Function Point total.

$$VAF = 0.65 + [(\sum_{i=1}^{14} Ci) / 100]$$

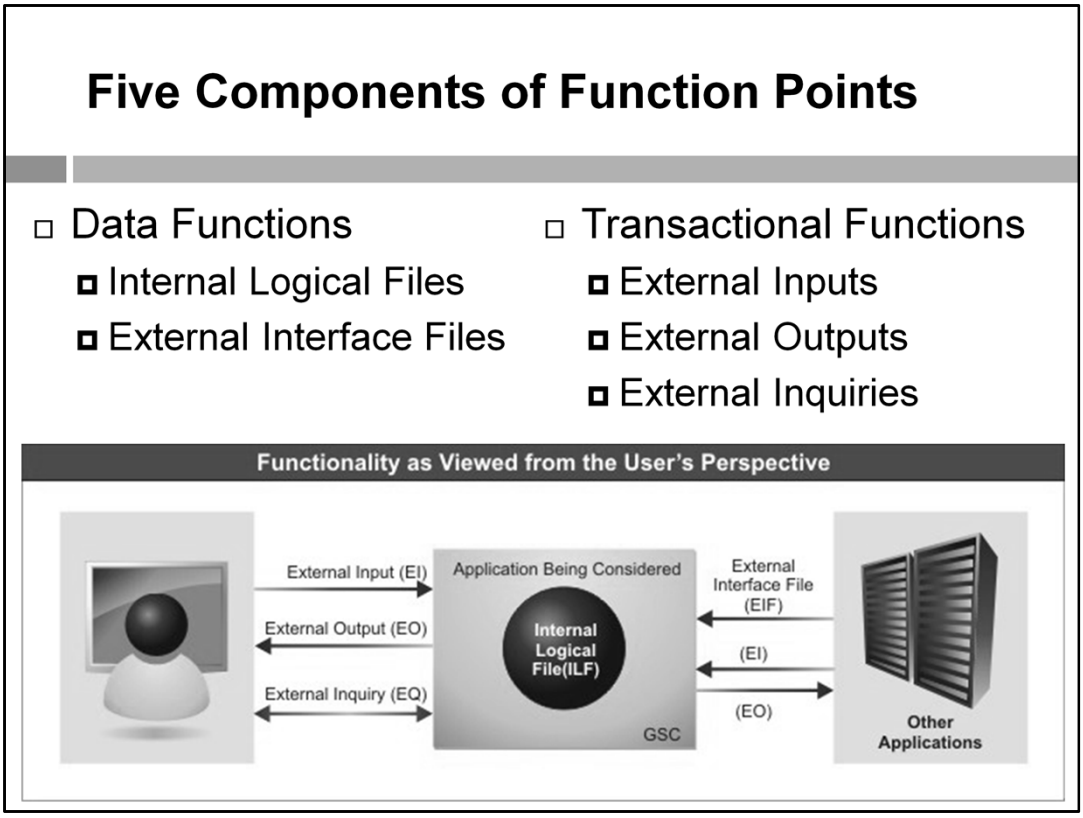
where: Ci = degree of influence for each General System Characteristic
.i = is from 1 to 14 representing each GSC.
$\sum$ = is summation of all 14 GSC's.

In identifying each ILF, EIF, EO, El, and EQ, a complexity matrix was used to determine the complexity for each data and transactional function type in terms of low, average, or high complexity.

Step 7: Calculate the Final Adjusted Function Point Count

- Adjusted FP Count = Unadjusted Function Point (UAF)
  x Value Adjustment Factor (VAF)

Type of Component	Complexity of Components			
	Low	Average	High	Total
External Inputs	___ x 3 = ___	___ x 4 = ___	___ x 6 = ___	
External Outputs	___ x 4 = ___	___ x 5 = ___	___ x 7 = ___	
External Inquiries	___ x 3 = ___	___ x 4 = ___	___ x 6 = ___	
Internal Logical Files	___ x 7 = ___	___ x 10 = ___	___ x 15 = ___	
External Interface Files	___ x 5 = ___	___ x 7 = ___	___ x 10 = ___	
		Total Number of Unadjusted Function Points		_____
		Multiplied Value Adjustment Factor		_____
		Total Adjusted Function Points		_____



Internal Logical Files - The first data function allows users to utilize data they are responsible for maintaining. For example, a pilot may enter navigational data through a display in the cockpit prior to departure. The data is stored in a file for use and can be modified during the mission. Therefore the pilot is responsible for maintaining the file that contains the navigational information. Logical groupings of data in a system, maintained by an end user, are referred to as Internal Logical Files (ILF).

External Interface Files - The second Data Function a system provides an end user is also related to logical groupings of data. In this case the user is not responsible for maintaining the data. The data resides in another system and is maintained by another user or system. The user of the system being counted requires this data for reference purposes only. For example, it may be necessary for a pilot to reference position data from a satellite or ground-based facility during flight. The pilot does not have the responsibility for updating data at these sites but must reference it during the flight. Groupings of data from another system that are used only for reference purposes are defined as External Interface Files (EIF).

The remaining functions address the user's capability to access the data contained in ILFs and EIFs. This capability includes maintaining, inquiring and outputting of data. These are referred to as Transactional Functions.

External Input - The first Transactional Function allows a user to maintain Internal Logical Files (ILFs) through the ability to add, change and delete the data. For example, a pilot can add, change and delete navigational information prior to and during the mission. In this case the pilot is utilizing a transaction referred to as an External Input (EI). An External Input gives the user the capability to maintain the data in ILF's through adding, changing and deleting its contents.

External Output - The next Transactional Function gives the user the ability to produce outputs. For example a pilot has the ability to separately display ground speed, true air speed and calibrated air speed. The results displayed are derived using data that is maintained and data that is referenced. In function point terminology the resulting display is called an External Output (EO).

External Inquiries - The final capability provided to users through a computerized system addresses the requirement to select and display specific data from files. To accomplish this a user inputs selection information that is used to retrieve data that meets the specific criteria. In this situation there is no manipulation of the data. It is a direct retrieval of information contained on the files. For example if a pilot displays terrain clearance data that was previously set, the resulting output is the direct retrieval of stored information. These transactions are referred to as External Inquiries (EQ).

Internal Logical File (ILF)

- Internal Logical File (ILF) is a user identifiable group of logically related data or control information maintained within the boundary of the application.
- The primary intent of an ILF is to hold data maintained through one or more elementary processes of the application being counted.

External Interface File (EIF)

- External Interface File (EIF) is a user identifiable group of logically related data or control information referenced by the application, but maintained within the boundary of another application.
- The primary intent of an EIF is to hold data referenced through one or more elementary processes within the boundary of the application counted.
- This means an EIF counted for an application must be in an ILF in another application.

External Input (EI)

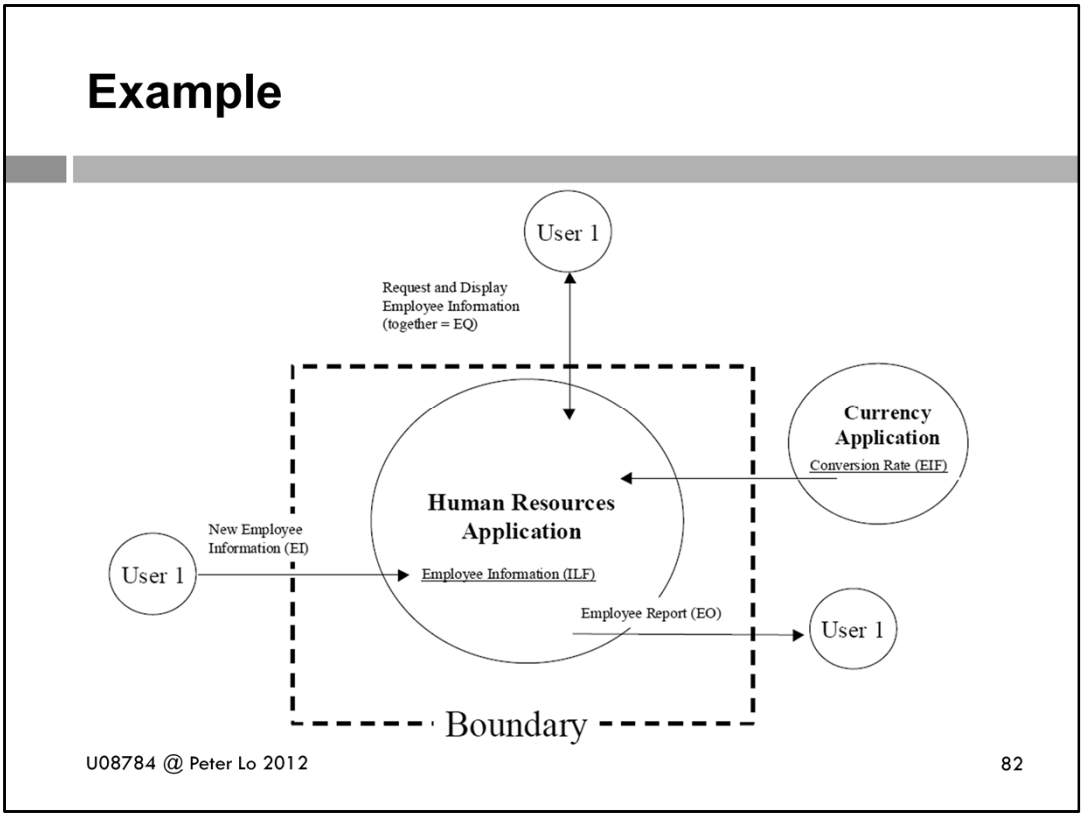
- An external input (EI) is an elementary process that processes data or control information that comes from outside the application's boundary.
- The primary intent of an EI is to maintain one or more ILFs and/or to alter the behavior of the system.

External Output (EO)

- An elementary process that sends data or control information outside the application's boundary.
- The primary intent of an external output is to present information to a user through processing logic other than or in addition to the retrieval of data or control information.
- The processing logic must contain at least one mathematical formula or calculation, or create derived data.
- An external output may also maintain one or more ILFs and/or alter the behavior of the system.

External Inquiry (EQ)

- An elementary process that sends data or control information outside the application boundary.
- The primary intent of an external inquiry is to present information to a user through the retrieval of data or control information.
- The processing logic contains no mathematical formula or calculation, and creates no derived data.
- No ILF is maintained during the processing, nor is the behavior of the system altered.



The counting scope defines the functionality which will be included in a particular function point count.
The application boundary indicates the border between the software being measured and the user.
The example shows the application boundary between the Human Resources Application being measured and the external Currency Application. It also shows the application boundary between the Human Resources Application and the user

Internal Logical File (ILF) - A group of related employee data maintained within the Human Resources Application.
External Interface File (EIF) - Conversion rate information maintained by the Currency Application and referenced by the Human Resources Application
External Input (EI) - The process of entering employee information into the Human Resources Application.
External Output (EO) - The process of producing a report that lists all employees stored in the Human Resources Application.
External Inquiry (EQ) - The process of inquiring on employee information (input request) and viewing an employee's information when it appears on a screen (output retrieval).

Record Element Type (RET)

- A Record Element Type (RET) is a user recognizable subgroup of data elements within an ILF or EIF.
- There are two types of subgroups:
  - **Optional Subgroups** are those that the user has the option of using one or none of the subgroups during an elementary process that adds or creates an instance of the data.
  - **Mandatory Subgroups** are subgroups where the user must use at least one.

U08784 @ Peter Lo 2012 83

For example, in a Human Resources Application, information for an employee is added by entering some general information. In addition to the general information, the employee is a salaried or hourly employee.
The user has determined that an employee must be either salaried or hourly. Each type of employee has unique attributes. Either type can have information about dependents. For this example, there are three subgroups or RETs as shown below:
• Salaried employee (mandatory); includes general information
• Hourly employee (mandatory); includes general information
• Dependent (optional)

One of the following rules applies when counting RETs:
Count a RET for each optional or mandatory subgroup of the ILF or EIF.
Or
If there are no subgroups, count the ILF or EIF as one RET.

Data Element Types (DET)

- A data element type is a unique user recognizable, non-repeated field.
- DET Rules
  - Count a DET for each unique user recognizable, non-recursive field on the ILF or EIF.
  - Count a DET for each piece of data in an ILF or EIF that exists because the user requires a relationship with another ILF to be maintained.
- For example, an account number that is stored in multiple fields is counted as one DET.

U08784 @ Peter Lo 2012 84

Count a DET for each unique user recognizable, non-repeated field maintained in or retrieved from the ILF or EIF through the execution of an elementary process.
For example, an account number that is stored in multiple fields is counted as one DET.
For example, a before or after image for a group of 10 fields maintained for audit purposes would count as one DET for the before image (all 10 fields) and as one DET for the after image (all 10 fields) for a total of 2 DETs.

Count Data Function

- Rate the functional complexity using the following complexity matrix.

	1 to 19 DET	20 to 50 DET	51 or more DET
1 RET	Low	Low	Average
2 to 5 RET	Low	Average	High
6 or more RET	Average	High	High

Count Transaction Function

- External Input

	1 to 4 DET	5 to 15 DET	16 or more DET
0 to 1 FTR	Low	Low	Average
2 FTRs	Low	Average	High
3 or more FTRs	Average	High	High

- External Outputs and External Inquiries

	1 to 5 DET	6 to 19 DET	20 or more DET
0 to 1 FTR	Low	Low	Average
2 to 3 FTRs	Low	Average	High
4 or more FTRs	Average	High	High

Calculate the Unadjusted Function Point

- Once we have the score of all the functionalities, we can get the UFP as
  - UFP = Sum of all the Complexities of all the EI's, EO's EQ's, ILF's and EIF's

Complexity				
Description	Low	Medium	High	Total
Inputs EIs	___ x 3	___ x 4	___ x 6	___
Outputs EOs	___ x 4	___ x 5	___ x 7	___
Queries EQs	___ x 3	___ x 4	___ x 6	___
Files ILFs	___ x 7	___ x 10	___ x 15	___
Program Interfaces ELFs	___ x 5	___ x 7	___ x 10	___

Contribution Procedure Translation Table

	Low	Average	High
Internal Logical File	7	10	15
External Interface File	5	7	10
External Input	3	4	6
External Inquiry	3	4	6
External Output	4	5	7

General System Characteristics

- The value adjustment factor (VAF) is based on 14 general system characteristics (GSC's) that rate the general functionality of the application being counted.
- Each characteristic has associated descriptions that help determine the degrees of influence of the characteristics.
- The degrees of influence range on a scale of zero to five.

Scale	Description
0	Not present, or no influence
1	Incidental influence
2	Moderate influence
3	Average influence
4	Significant influence
5	Strong influence throughout

General System Characteristics List

- Data communications
- Distributed data processing
- Performance
- Heavily used configuration
- Transaction rate
- Online data entry
- End-user efficiency
- Online update
- Complex processing
- Reusability
- Installation ease
- Operational ease
- Multiple sites
- Facilitate change

Calculate the Adjusted Function Point Count

- The adjusted function point count is calculated using a specific formula for a development project, enhancement project, or application (system baseline) function point count.

$$VAF = 0.65 + [(\sum_{i=1}^{14} Ci) / 100]$$

where: Ci = degree of influence for each General System Characteristic
.i = is from 1 to 14 representing each GSC.
$\sum$ = is summation of all 14 GSC's.

Advantages

- The project is viewed from the perspective of the user rather than the developer, that is, in terms of user functions rather than programs, files, or objects.
- The estimates can be developed from knowledge of the requirements without a detailed design solution being known. This provides for a level of independence from the specific hardware platform, languages, developer's skill level, and the organization's line of business.
- The use of FPA is accepted internationally. There is also a users group, International Function Point Users Group (IFPUG), which has established standards to help encourage consistency in counting function points.

Disadvantages

- Function point counts are affected by project size
- Difficult to apply to massively distributed systems or to systems with very complex internal processing
- Difficult to define logical files from physical files
- The validity of the weights that Albrecht established – and the consistency of their application – has been challenged
- Different companies will calculate function points slightly different, making intercompany comparisons questionable

U08784 @ Peter Lo 2012 91

This approach does not accurately estimate systems that are largely algorithmic such as military systems, space systems, robotics, process control, and middleware.
Function Points can be complicated to administer. Formal training is needed before you can consistently count, and therefore track, function points.
The use of function points is not widely accepted within IT Services. As a result, we have not gathered any estimating guidelines or metrics for function point estimating.
Since the concept of FPA was developed with older technologies and development approaches, it is not certain how well this concept applies to newer technologies and development approaches such as object-oriented development. However, variations of Function Point Analysis are being developed to address the newer technologies and development approaches.

Constructive Cost Model

- **Co**nstructive **Co**st **Mo**del (COCOMO) predicts the effort and schedule for a software product development based on inputs relating to the size of the software and a number of cost drivers that affect productivity
- It classifies software as
  - **Organic Software** is more-or-less freestanding, e.g. applications or system software.
  - **Embedded Software** is intimately bound up with a larger hardware / software system, e.g. missile control systems software.
  - **Semi-detached** is between the two.

COCOMO is one of the most widely used software estimation models in the world
It was developed by Barry Boehm in 1981

COCOMO Model

- The model has three levels,
  - Basic – quick, easy, first approximation
  - Intermediate – 'effort multipliers', which are the product of 15 cost drivers
  - Advanced – includes still more cost drivers

Model Level	Parameters Used
Basic	Program Size
Intermediate	Program Size + Cost Driver (Subjective assessment of product, Hardware, Personnel, Project's attribute. There are totally 4 cost drivers with 15 attributes.)
Advanced	Program Size + Cost Driver + Cost Driver Impact on analysis, design etc.

Assumptions for COCOMO

- Primary cost driver is the number of Delivered Source Instructions (DSI) / Delivered Line Of Code developed by the project
- COCOMO estimates assume that the project will enjoy good management by both the developer and the customer
- Assumes the requirements specification is not substantially changed after the plans and requirements phase

Basic COCOMO

- The basic COCOMO equations take the form
  - Effort Applied in man-months (E) = a x KLOC x b
  - Development Time in months (D) = c x E x d
  - People required (P) = E / D
- where, KLOC is the estimated number of delivered lines of code for project, The coefficients a, b, c and d are given in the following table.

Software Project	a	b	c	d
Organic	2.40	1.05	2.50	0.38
Semi-detached	3.00	1.12	2.50	0.35
Embedded	3.60	1.20	2.50	0.32

Intermediate COCOMO

- The intermediate COCOMO equations take the form
  - Effort Applied in man-months (E) = a x $KLOC^b$ x EAF
  - Development Time in months (D) = c x E x d
  - People required (P) = E / D
- where KLOC is the estimated number of thousands of delivered lines of code for the project, and EAF is the factor. The coefficient a and the exponent b are given in the next table.

Software Project	a	b	c	d
Organic	3.20	1.05	2.50	0.38
Semi-detached	3.00	1.12	2.50	0.35
Embedded	2.80	1.20	2.50	0.32

Intermediate COCOMO computes software development effort as function of program size and a set of "cost drivers" that include subjective assessment of product, hardware, personnel and project attributes.

Typical values for EAF

Cost Drivers	Ratings					
	Very Low	Low	Nominal	High	Very High	Extra High
Product attributes						
Required software reliability	0.75	0.88	1.00	1.15	1.40	
Size of application database		0.94	1.00	1.08	1.16	
Complexity of the product	0.70	0.85	1.00	1.15	1.30	1.65
Hardware attributes						
Run-time performance constraints			1.00	1.11	1.3	1.66
Memory constraints			1.00	1.06	1.21	1.56
Volatility of the virtual machine environment		0.87	1.00	1.15	1.30	
Required turnabout time		0.87	1.00	1.07	1.15	
Personnel attributes						
Analyst capability	1.46	1.19	1.00	0.86	0.71	
Applications experience	1.29	1.13	1.00	0.91	0.82	
Software engineer capability	1.42	1.17	1.00	0.86	0.7	
Virtual machine experience	1.21	1.10	1.00	0.90		
Programming language experience	1.14	1.07	1.00	0.95		
Project attributes						
Application of software engineering methods	1.24	1.10	1.00	0.91	0.82	
Use of software tools	1.24	1.10	1.00	0.91	0.83	
Required development schedule	1.23	1.08	1.00	1.04	1.10	

Reference

- Ch. 6, PMBOK Guide, 4th Edition
- Ch. 3, 5-6, Software Project Management, 4th Edition
- Ch. 2-3, 9-10, Project Management for Information Systems, 5th Edition
- Function Point Counting Practices Manual Release 4.2.1
  - http://www.ifpug.org