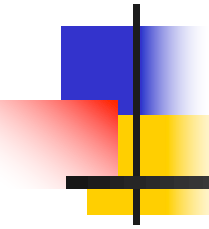
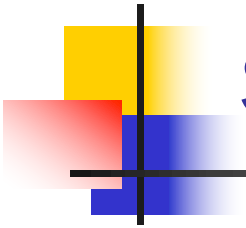


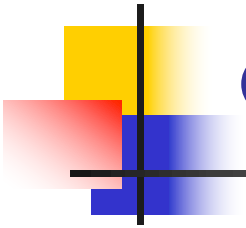
Software Project Management

Planning



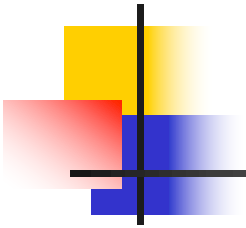
Software Project

- Goal: Build a software system to meet commitments on
  - Cost
  - Schedule
  - Quality
- Worldwide – many projects fail
  - One-third are runaways with cost
  - Schedule overrun of more than 125%



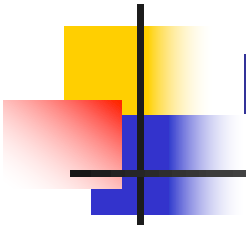
Cont…

- We conduct planned and controlled software projects for one primary reason. It is the only way to manage complexity.
- Report in 2009
  - only 32 percent of projects succeeding; that is, they are delivered on time, on budget, with required features and functionality.
  - Forty four percent were “challenged,” meaning they were late, over budget, and/or had less than the required features and functionality.
  - Twenty four percent failed; that is, they were cancelled prior to completion, or delivered and never used.



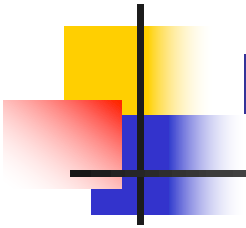
Project Failures

- Major reasons for project runaways
  - Unclear objectives
  - Bad Planning
  - No project management methodology
  - New Technology
  - Insufficient staff
- All of these relate to project management
- Effective project management is key to successfully executing a project



Improving Project Management

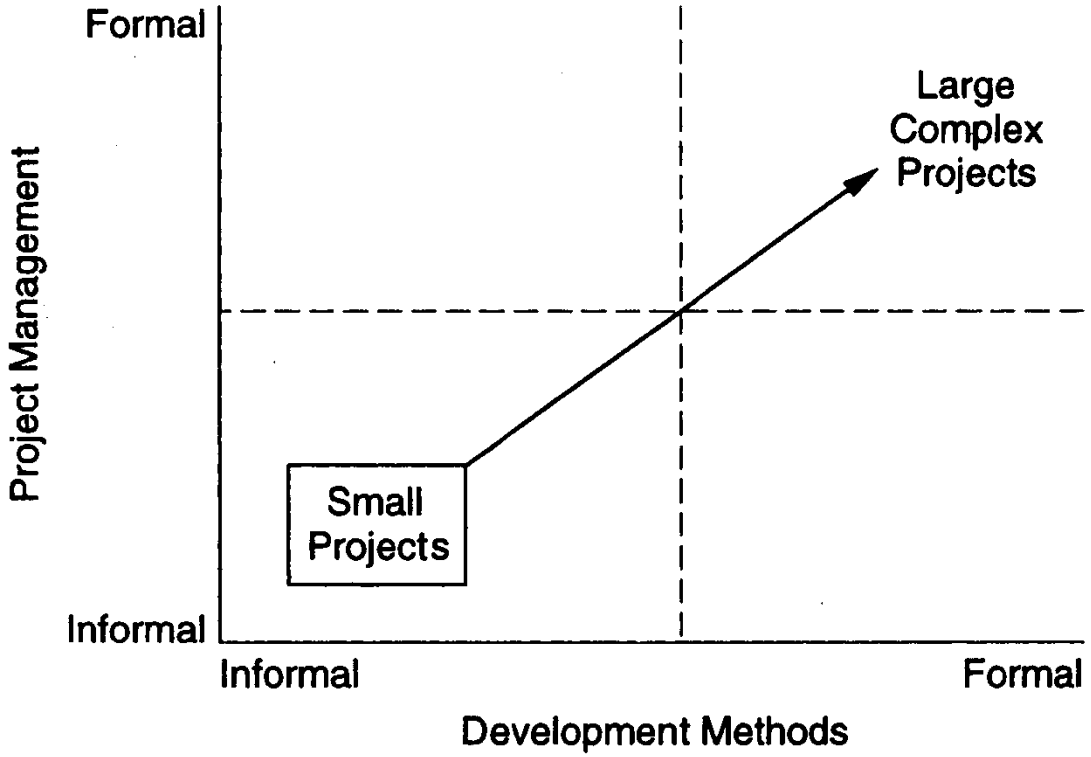
- Better predictability leading to commitments that can be met
- Lower cost through reduced rework, better resource management, better planning,…
- Improved quality through proper quality planning and control
- Better control through change control , CM, monitoring, etc…

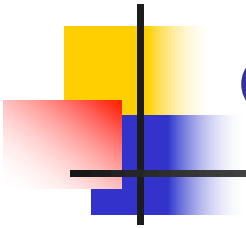


Improving Project Management

- Better visibility into project health and state leading to timely intervention
- Better handling of risks reducing the chances of failure
- All this leads to higher customer satisfaction
- And self and organization improvement

Two Dimensions in Project Execution





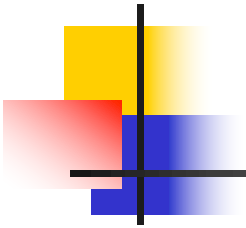
Cont…

- Small project both the Project Management and Engineering can be done informally
- Large projects require formality
- Formality: well defined processes used for each task; measurements used to control

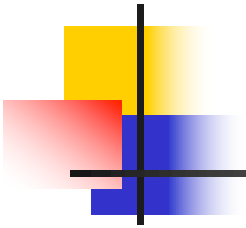
The Project Management Process

- Has three phases - planning, monitoring and control, and closure
- Planning is done before the main engineering life cycle and closure after the life cycle
- Monitoring and controlling phase is in parallel with life cycle
- So, we can broadly classified project manager's activity into:
  - Project planning,
  - Project monitoring and controlling activity

Project Planning Activities

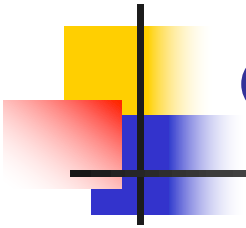


- Estimation
  - Effort, cost, resource, and project duration
- Project Scheduling
- Suitable process for the execution
- Staff organization
  - Staffing plans
- Risk Handling
  - Identification, analysis and mitigation
- Miscellaneous plans
  - Quality assurance plan, configuration management plan, etc…
- Main output - a project management plan and the project schedule



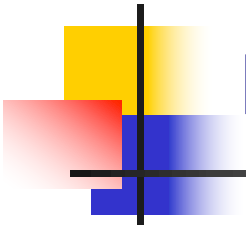
Process Planning

- Plan how the project will be executed, i.e. the process to be followed
- Process will decide the tasks, their ordering, milestones
- Hence process planning is an important project planning task
- Various life cycle models - waterfall, iterative, prototyping; diff models suit different projects



Cont …

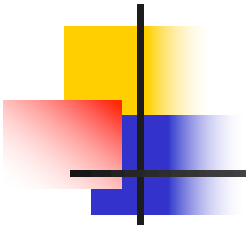
- During planning can select the model that is best for the project
- This gives the overall process which has to be fine-tuned to suit the project needs
- Usually done by process tailoring - changing the process to suit the project
- Tailoring finally results in adding, deleting, modifying some process steps



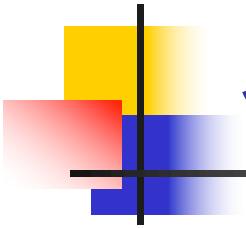
Reluctance to Planning

- Takes too much time and cost
  - Preventive action
  - Long-term payoff is greater than short-term cost
- Too tedious (mental activity)
  - "Thinker" and "doer"
- Ego (shoot from the hip)
  - Not realistic

Project Planning Consequence



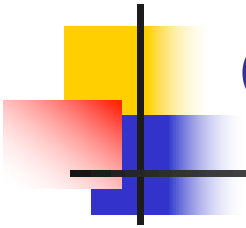
- Project planning requires proper care and attention
- Unrealistic time and resource estimates
  - Annoying delays
  - Customer dissatisfaction
  - Affect team morale
    - Poor quality work
  - **Project Failure**



SPMP Document

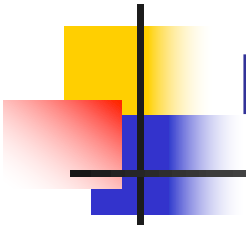
- After Planning is complete
  - Document the plans
  - In a Software Project Management Plan (SPMP) document

Organization of SPMP Document



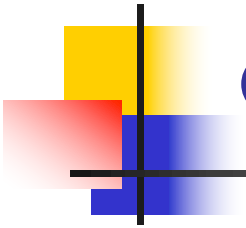
- Introduction (Objectives, Major Functions, Performance Issues, Management and Technical Constraints)
- Project Estimates (Historical Data, Estimation Techniques, Effort, Cost, and Project Duration Estimates)
- Project Resources Plan (People, Hardware and Software, Special Resources)
- Schedules (Work Breakdown Structure, Task Network, Gantt Chart Representation, PERT Chart Representation)
- Risk Management Plan (Risk Analysis, Risk Identification, Risk Estimation, Abatement Procedures)
- Project Tracking and Control Plan
- Miscellaneous Plans (Process Tailoring, Quality Assurance)

Software Cost Estimation



Effort Estimation

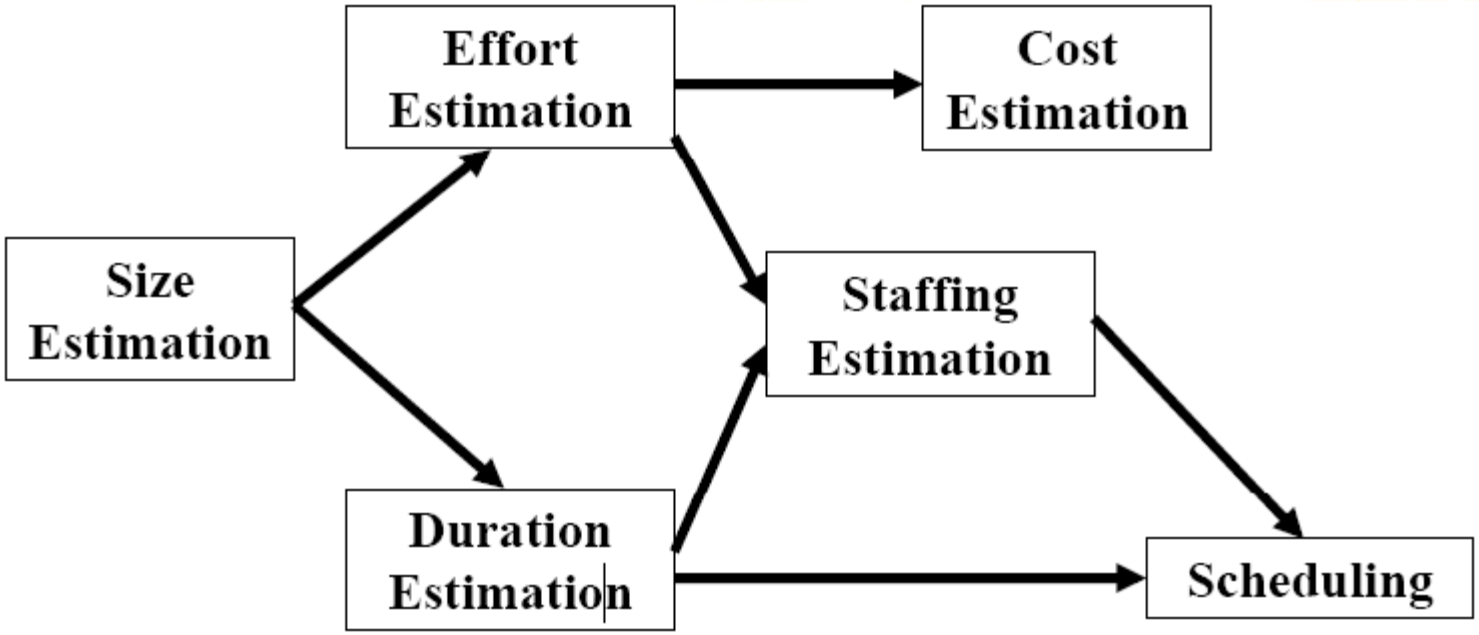
- For a project total cost and duration has to be committed in start
- Requires effort estimation, often in terms of person-months
- Effort estimate is key to planning – schedule, cost, resources depend on it
- Many problems in project execution stem from improper estimation

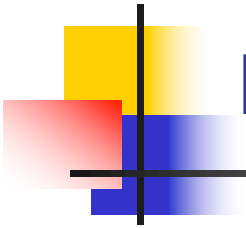


Cont…

- No easy way, no silver bullet
- Estimation accuracy can improve with more information about the project
- Early estimates are more likely to be inaccurate than later
  - More uncertainties in the start
  - With more info, estimation becomes easier

Cont…

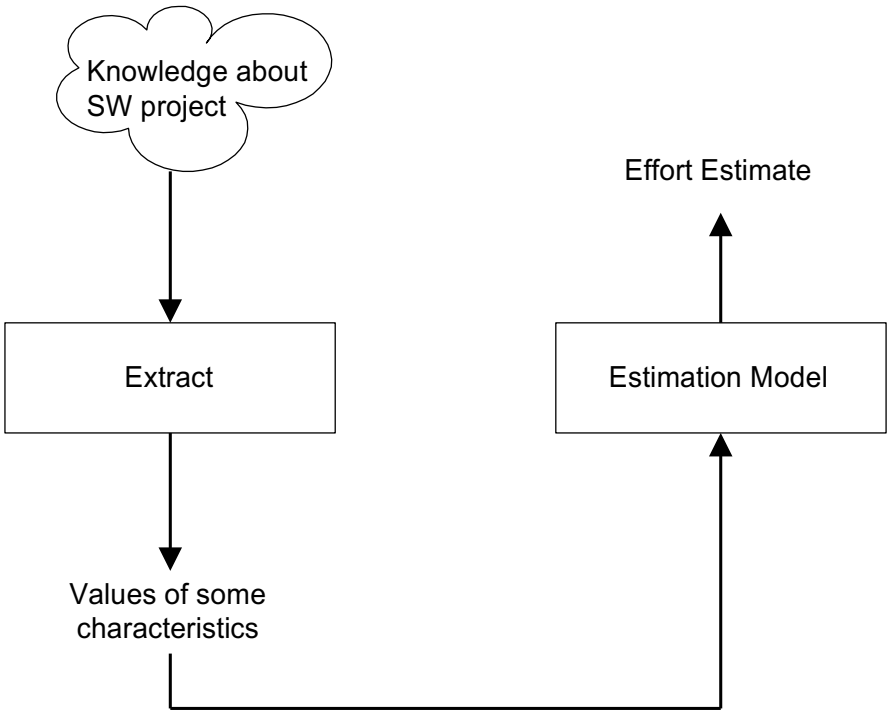

Effort Estimation
Cost Estimation
Size Estimation
Staffing Estimation
Duration Estimation
Scheduling

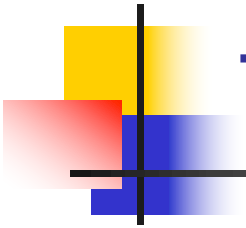


Effort Estimation Models

- A model tries to determine the effort estimate from some parameter values
- A model also requires input about the project, and cannot work in vacuum
- So, to apply a model, we should be able to extract properties about the system
- Two types of models
  - Top-down and
  - Bottom-up

Cont…


Knowledge about SW project
Extract
Values of some characteristics
Effort Estimate
Estimation Model

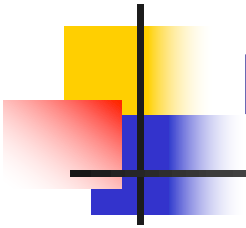


Top-down Estimation

- First determines the total effort, then effort for components
- Usually works with overall size
- One method is to see estimate as a function of effort; the common function used is

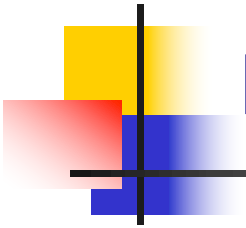
$$\text{Effort} = a * \text{size}^{b}$$

- E is in person-months, size in KLOC
- Constants a and b determined through regression analysis of past project data



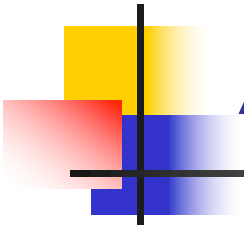
LOC

- LOC (Lines of Code)
  - Simplest and most widely used metric.
  - Comments and blank lines should not be counted.
  - Disadvantage
    - Size can vary with coding style
    - Focuses on coding activity alone
    - Difficult to estimate LOC from problem description



Bottom-Up Estimation

- Effort for components and phases first estimated, then the total
- use activity based costing - all activities enumerated and then each activity estimated separately
- group activities into classes - their effort estimate from past data

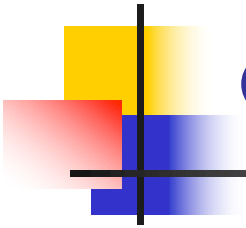


An Estimation Procedure

- Identify programs in the system and classify them as simple, medium, or complex (S/M/C)
- Define the average coding effort for S/M/C
- Get the total coding effort.
- Use the effort distribution in similar projects to estimate effort for other tasks and total
- Refine the estimates based on project specific factors

COCOMO Model for Estimation

- COCOMO (COnstructive COst MOdel) proposed by Boehm
- a top-down approach
- Uses size, but adjusts using some factors
- Basic procedure
  - Obtain initial estimate using size
  - Determine a set of 15 multiplying factors from different project attributes
  - Adjust the effort estimate by scaling it with the final multiplying factor

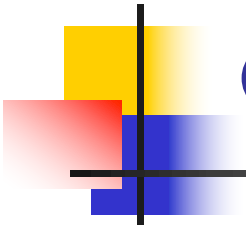


Cont…

- Initial estimate: a * size ^{b}; some standard values for a, b given for different project types
- There are 15 cost driver attributes like reliability, complexity, application experience, capability, …
- Each factor is rated, and for the rating a multiplication factor is given
- Final effort adjustment factor is the product of the factors for all 15 attributes

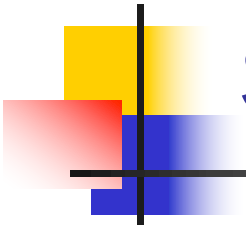
COCOMO-Cost Driver

Cost Driver	Very low	Low	Nominal	High	Very High
Required reliability	.75	.88	1.0	1.15	1.4
Database size		.94	1.0	1.08	1.16
Product complexity	.7	.85	1.0	1.15	1.3
Execution time constraint			1.0	1.11	1.3
Memory constraint			1.0	1.06	1.21
Analyst capability	1.46	1.19	1.0	.86	.71
Application experience	1.29	1.13	1.0	.91	.82
Programmer capability	1.42	1.17	1.0	.86	.70
Use of software tools	1.24	1.10	1.0	.91	.83
Development schedule	1.23	1.08	1.0	1.04	1.1



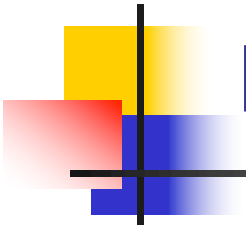
COCOMO-Effort Distribution

- Effort distribution among different phases is given as a percent of effort
- Eg. For medium size product it is
  - Product design – 16%
  - Detailed design – 24%
  - Coding and UT – 38%
  - Integration and test – 22%



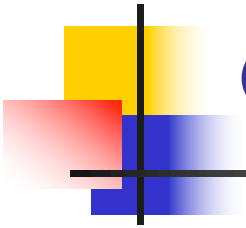
Size

- Effort for project depends on many factors
- Size is the main factor – many experiments and data analysis have validated this
- Size in the start is only an estimate; getting size estimates from requirement is hard
- Need a size unit that can be "computed" from requirements
- Function points attempt to do this



Function Points

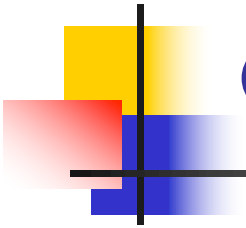
- a size measure like LOC
- Determined from SRS
- Defines size in terms of " functionality "
- Why "measure" size early ?
  - Needed for estimation and planning
- Five different parameters
  - external input type
  - external output type



Cont…

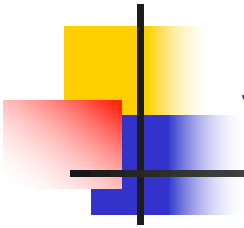
- logical internal file type
- external interface file type
- external inquiry type

- These five parameters capture the functionality of a system
- within a type , an element may be simple , average or complex
- A weighted sum is taken



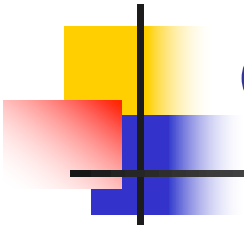
Cont…

- For Example for External output type
  - each unique output that leave system boundary
  - E.g. (Reports , messages to user , data to other applications)
- Simple : few columns
- Average : many columns
- Complex : references many files for production



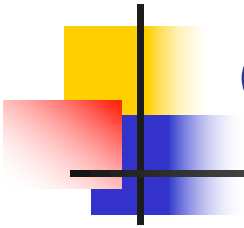
Scheduling and Staffing

- Project Scheduling
  - A project Schedule is at two levels - overall schedule and detailed schedule
  - Overall schedule comprises of major milestones and final date
  - Detailed schedule is the assignment of lowest level tasks to resources



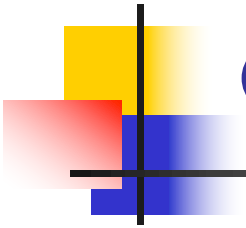
Cont…

- Overall Schedule
  - Depends heavily on the effort estimate
  - For an effort estimate, some flexibility exists depending on resources assigned
    - Eg a 56 PM (person-months) project can be done in 8 months (7 people) or 7 months (8 people)
  - Stretching a schedule is easy; compressing is hard and expensive



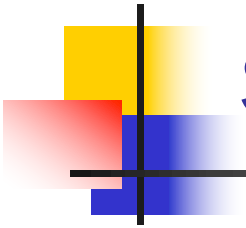
Cont…

- Overall Schedule
  - One method is to estimate schedule S (in months) as a function of effort in PMs (person-months)
  - Can determine the fn through analysis of past data; the function is non linear
  - COCOMO: $S = 2.5\ E^{3.8}$
  - Often this schedule is checked and corrected for the specific project



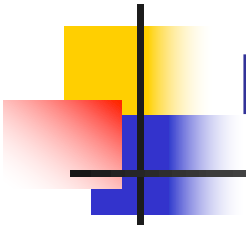
Cont…

- Detailed Scheduling
  - Detailed schedule not done completely in the start - it evolves
  - Can use MS Project for keeping it
  - Detailed Schedule is the most live document for managing the project
  - Any activity to be done must get reflected in the detailed schedule



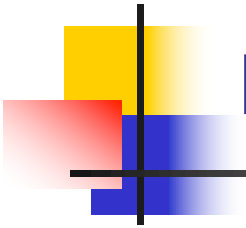
Some points on Scheduling

- Effectively schedule, allocate, use, and replace resources to achieve goals
- Master schedule is the basic tool and main output of planning
- Project control is based on comparing the progress with schedule
- Planning and scheduling are dynamic



Effective Schedule

- Understandable
- Sufficiently detailed
- Highlighting critical tasks
- Flexible
- Based on reliable estimates
- Conform to available resources
- Compatible with other related projects



Developing the Schedule

- Defining objectives
  - Attainable, definitive, quantifiable, with specific duration
- Breaking down the work
- Sequencing the activities
- Estimating the activity costs and durations
- Reconciling with time constraints
- Reconciling with resource constraints
- Reviewing

Work Breakdown Structure

Work Breakdown Structure

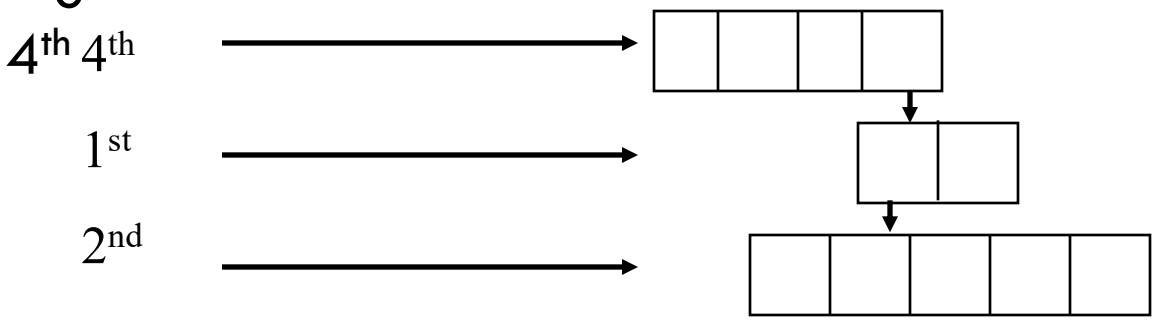
- A WBS is a deliverable-oriented grouping of the work involved in a project that defines the total scope of the project
- A WBS is a foundation document that provides the basis for planning and managing project schedules, costs, resources, and changes
- A WBS shows the breakdown (decomposition) of the major project deliverables into smaller, more manageable components

Work Breakdown Structure

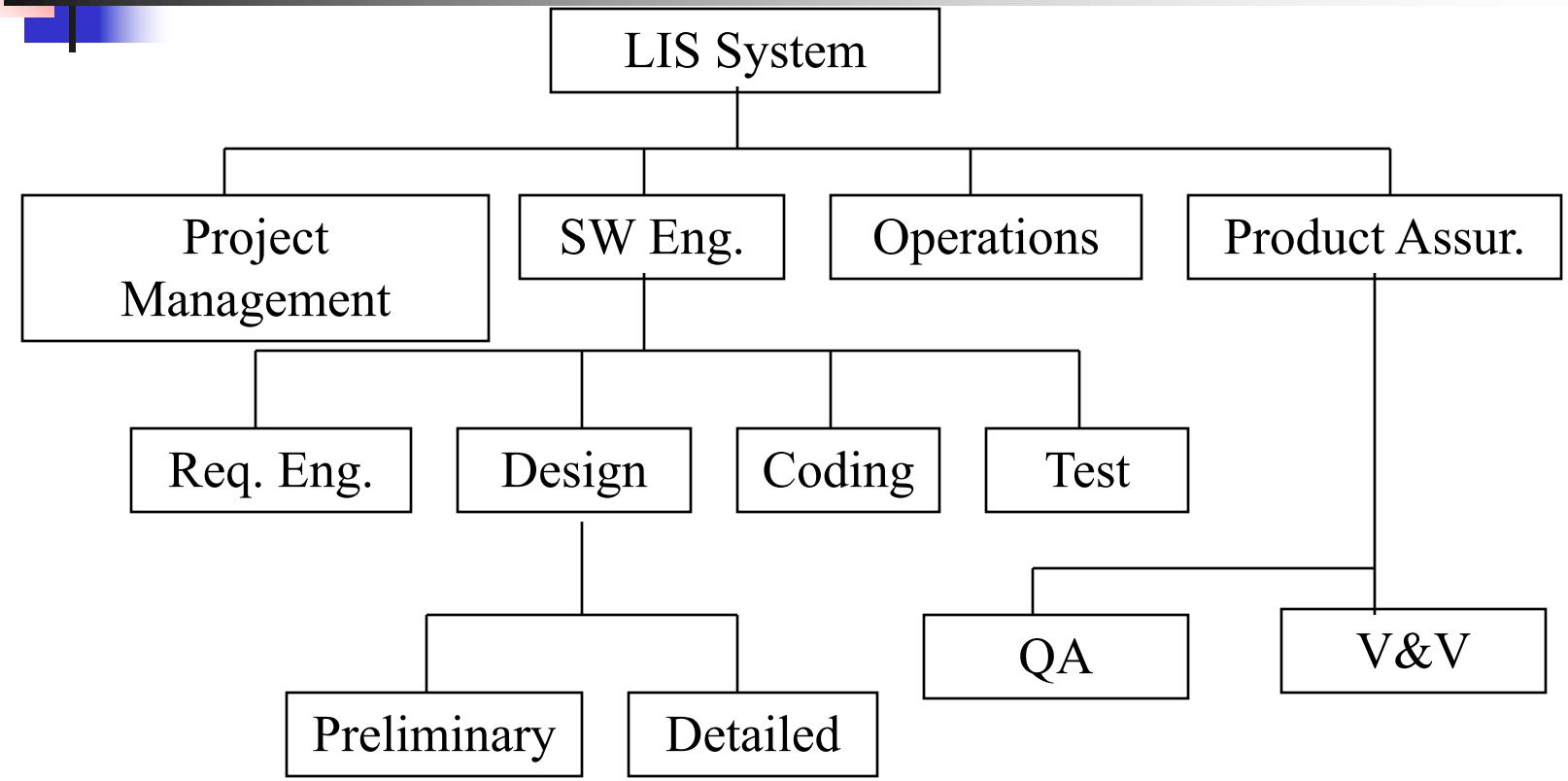
- It gives more accurate estimates, baseline for performance measurement and clear responsibility assignment
- In the context of WBS, work refers to work products or deliverables that are the result of effort
- Deliverables and work at the lowest level of the WBS are called **work packages**
- Work packages are typically the lowest level of work managed by the project manager
- This is also the level at which costs and time can be reliably estimated

Work Breakdown Structure

- WBS is a hierarchical representation of a process or product or both (hybrid).
- WBS can be shown in a tree graph or as an indented list
- A decimal numbering to label the elements
  - e.g. 4.1.2 is the 2nd element of the 1st element of the 4th 4th



Tree Graph WBS for LIS

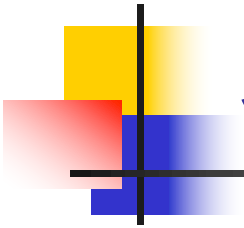


Indented List WBS for LIS

- 0.0 Land Information System (LIS)
- 1.0 Project Management
- 2.0 Software Engineering
  - 2.1 Requirement Engineering
  - 2.2 Design
  - 2.3 Coding
  - 2.4 Test
- 3.0 Operations
- 4.0 Product Assurance
  - 4.1 Quality Assurance
  - 4.2 Verification and Validation

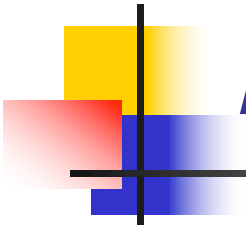
Notes on WBS

- "Rolling wave" approach
  - First top levels
  - Gradual completion
- WBS dictionary
- Make sure about numbering scheme
  - Top-level zero or one, …
- Work package specification for lowest level entries (info, completion, …)



Sequencing Scheduled Activities

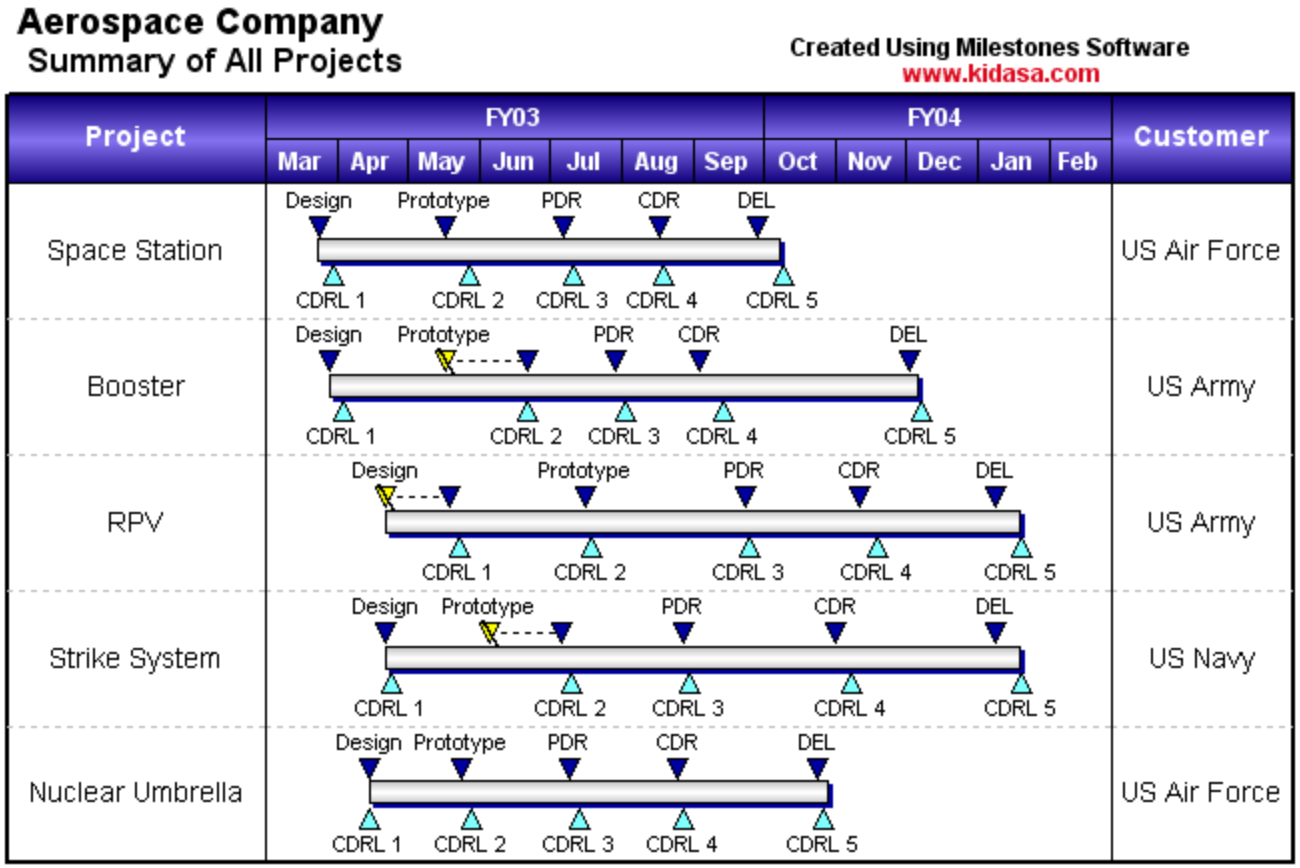
- Interrelationship among activities
- Milestones and Gantt charts are most common
  - Gantt chart also shows the relationship between work load and time
- Precedence networks are used for larger projects
  - Critical Path Method (CPM)
  - Program Evaluation and Review Technique
  - analyze the involved tasks in completing a given project, especially the time needed to complete each task, and identifying the minimum time needed to complete the total project.

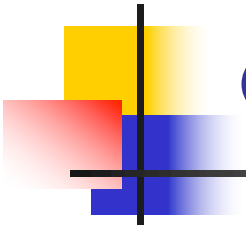


Milestone Chart

- Simplest scheduling method
- Small projects or summary of larger ones
- Ease and minimal cost
- No interrelationships exhibited
- Only completion dates
- Not enough feedback

Milestone Chart

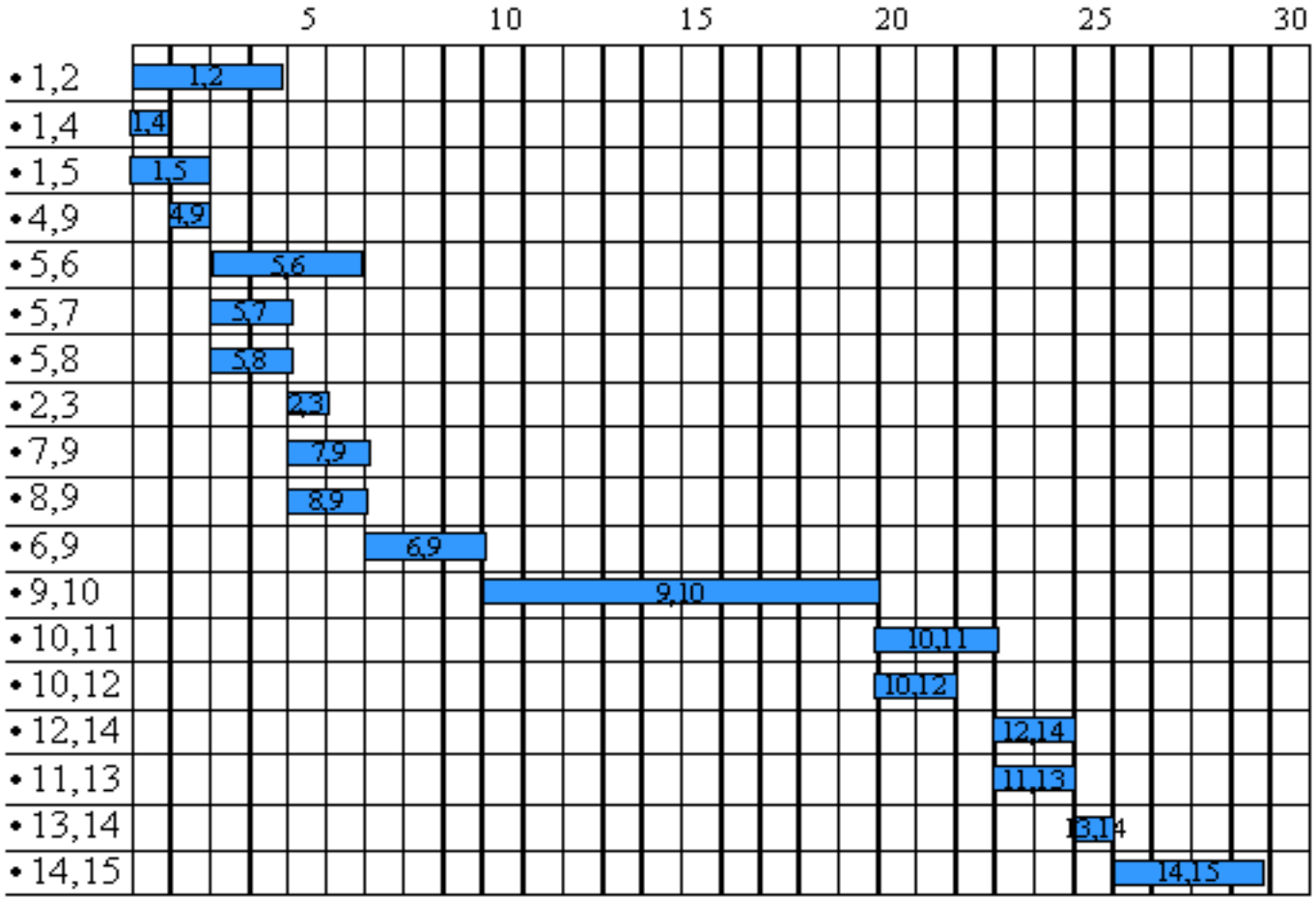




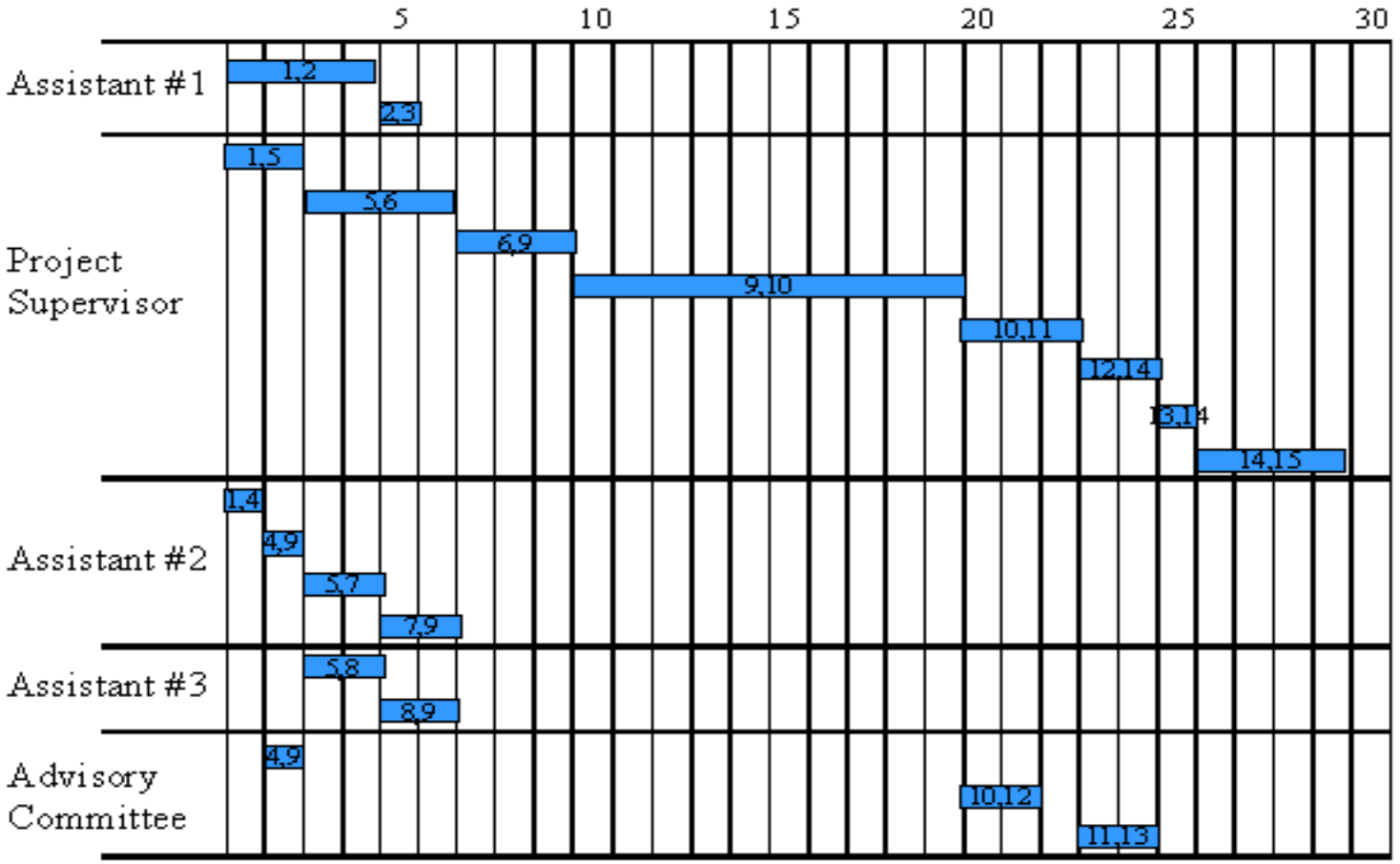
Gantt Chart

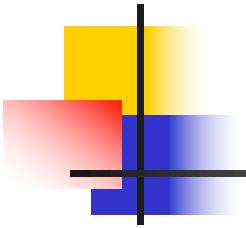
- Gantt or Bar chart used more frequently than others
- Suitable for projects with less than 25 activities
- Graphical display of start/end times
- Shows overlapping activities easily
  - CPM or PERT are translated to Gantt sometimes
- For estimation of resource and budget vs. time

Gantt Example -1



Gantt Example -2

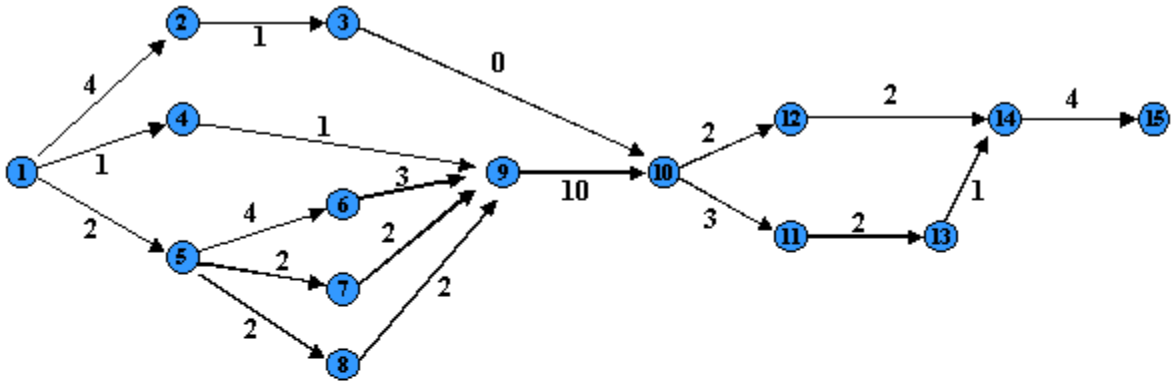




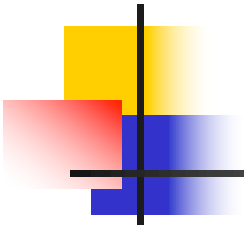
Precedence Networks

- Network is a graphical representation of sequential relationship between:
  - Activities
  - Events

CPM/PERT Graph

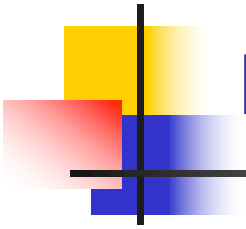


- 1,2 Literature review
- 2,3 Develop reaction eval. instrument
- 1,4 Strike committee
- 4,9 Meet with advisory committee
- 1,5 Develop needs questionnaire
- 5,6 1st division needs assessment
- 5,7 2nd division needs assessment
- 5,8 3rd division needs assessment
- 6,9 Analyze 1st division results
- 7,9 Analyze 2nd division results
- 8,9 Analyze 3rd division results
- 9,10 Develop 1st draft of curriculum
- 3,10 Dummy activity
- 10,11 Develop behavior eval. instrument
- 11,13 Advisory committee review (eval.)
- 13,14 Evaluations revisions
- 10,12 Advisory committee review (curr.)
- 12,14 Curriculum revisions
- 14,15 Pilot studies



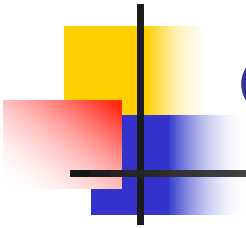
Developing Plan

- On typical software Project,
  - Development of the project plans tends to occur in multiple passes
- On larger projects,
  - Different segments of the plan may be assigned to different people for development
  - As a project manager – to integrate the various plans into a complete and integrated set of planning documents
- On smaller projects,
  - Likely, PM develop most or all sections of the plan
  - Possibly with reviews and comments provided by topic area experts



Developing Preliminary Plans

- In developing the project plan
  - Regularly be revising , adjusting and revisiting the various sections as more of the project details are discussed, documented, and communicated to others.
- On each pass through the plan, you will
  - Add missing material
  - Revise existing material
  - Identify and resolve internal inconsistencies, and
  - Identify and prioritize remaining planning activities



Cont…

- As you add content and detail to the plan
  - Near-term details will be easier to capture than more distant ones.
  - Eg. The parts of the plan covering the next three to six months will typically include more detail than the parts covering months twelve through eighteen