07: Managing time (2)

Software Project Management

Philippe Kruchten

Module outline

- Schedule
- Techniques
  - Graphs
  - Critical path
- Other similar techniques
- Applicability in Software development

"Time is *not* money; with money you can put it on the table and you can see it, and if you leave it, it may even accumulate - whereas with time, you can't see it or touch it. It expires at a regular and consistent rate whether you use it or not."

Keith A. Pickavance (2008)

Hofstadter's law: It always takes longer than you expect, even when you take into account Hofstadter's Law.

— Douglas Hofstadter,

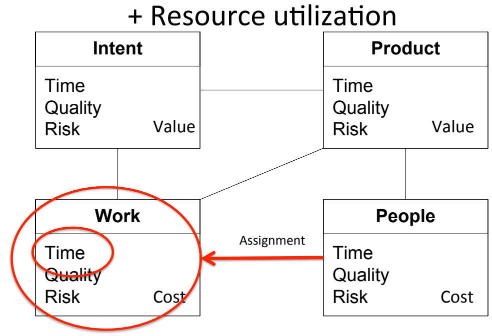
Gödel, Escher, Bach: An Eternal Golden Braid

Acknowledgment

Most material is from Prof. Anthony Finkelstein.

– Used with his permission

Schedule: Work over time + Resource utilization



Resource Scheduling

- Graphical networks and associated calculation techniques assist effective thinking by providing a step-by-step routine for coordinating work assignments and resource utilisation with project objectives
- Control criteria for the evaluation of work progress are established and the most economical means for correcting delays are diagnosed

Network Scheduling

- A network depicts the sequence of activities necessary to complete a project
- Segments of a project are represented by lines connected together to show the interrelationship of operations and resources
- When a duration is associated with each segment, the model shows the time distribution of the total project and its operations, this information can be used to coordinate the application of resources

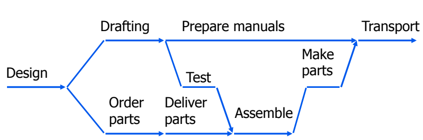
Techniques

- 2 techniques for network scheduling
  - Critical Path Method (CPM)
  - Project Evaluation and Review Technique (PERT)
- Path of critical activities that control the projects duration :
  - Critical Path Scheduling (CPS)
  - CPS is a management control tool for defining, integrating and analysing what must be done to complete a project economically and on time

Fundamentals

- Regardless of the name or the graphical conventions the fundamentals are the same
  - determine a list of necessary activities
  - establish a restriction list that sets the order of activity accomplishment
  - combine the two lists to construct a network

Arrow Diagram



shows operations required to produce custom made machine

1.Activity List

- Break the project down into its component operations to form a complete list of essential activities
  - Without a valid list all subsequent steps are meaningless!
  - An activity is a time-consuming task with a distinct beginning and endpoint
  - Some easily identified characteristic should be associated with each start and finish point

Tips on How to Construct an Activity List

- The way in which activities are defined may be influenced by planning purposes
  - for example if certain types of skilled people are in short supply activities requiring the limited resources should be separated from other operations
- The method or systems development process you are using should help you identify a "first-cut" activity list
  - for example the waterfall model identifies activities (and some sequencing)

2. Checking the Activity List

- A network is a composite picture of an entire undertaking, the activity list needs to be reviewed by suppliers, cooperating departments, subcontractors and anybody whose work impinges on the project

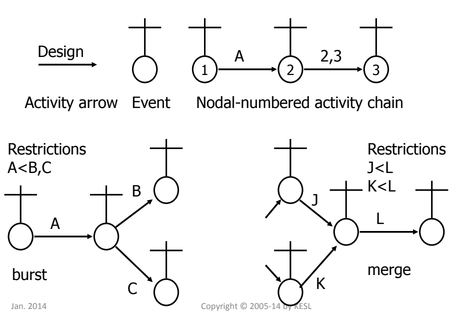
3. Restriction List

- Establishes the precedence of activities
- Uses the rough sequence generally arising from the activity list
- Each activity bracketed by the activities which must immediately precede it, the *prerequisite*, and the activity that must immediately follow it, the *postrequisite*

Activity and Restriction Lists Example

Description	Symbol	Prerequisite	Postrequisite	Restriction List
Design	A		Drafting, Order parts	A<B,C
Order parts	B	Design	Deliver parts	B<D
Drafting	C	Design	Prepare manuals, Make parts	C<E,F
Deliver parts	D	Order parts	Assemble	D<G
Prepare manuals	E	Drafting	Transport	E<I
Make parts	F	Drafting	Assemble	F<G
Assemble	G	Deliver parts, Make parts	Test	G<H
Test	H	Assemble	Transport	H<I

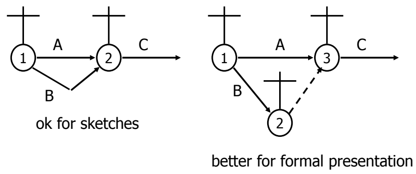
Network Conventions



Dummy

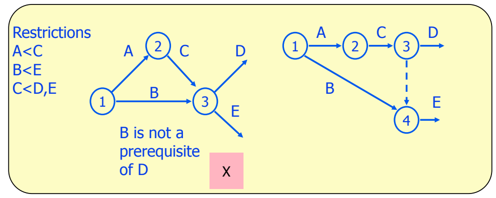
- A dashed line arrow is used in a network to show the dependency of one activity on another
- It is called a *dummy* activity and has all the restrictive properties of regular activities except that it takes zero time
- There are two types of dummies:
  - logic dummies which represent constraint relationships between nodes
  - artificial dummies which assist in numbering and uniquely identifying nodes

Artificial Dummies



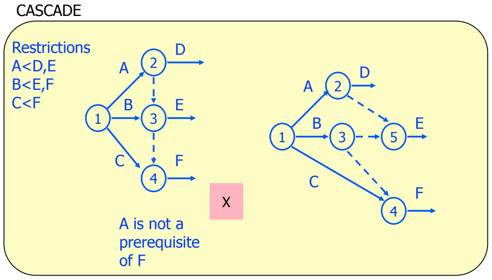
Common Errors I

SPILLOVER



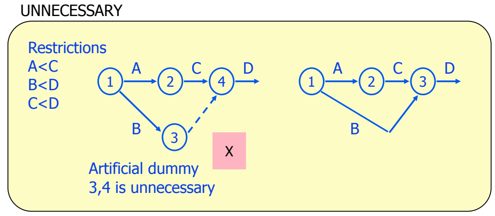
Common Errors II

CASCADE



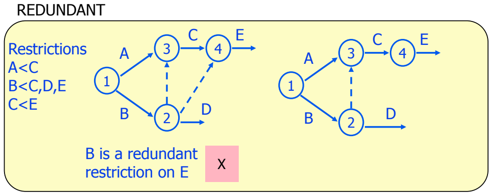
Common Errors III

UNNECESSARY



Common Errors IV

REDUNDANT



Critical Path

- The key to network scheduling is the critical path, the chain of activities that determines the total project duration
- Two approaches are available for estimating activity duration
  - a deterministic approach relies on a single most likely time estimate (this is probably sufficient for software projects)
  - a statistical approach uses a range of possible activity times to determine a single duration

Most Likely Time Estimates

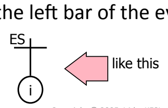
- An estimate of the elapsed time required to accomplish an activity is called the *activity duration*
- An activity duration may be expressed in months, weeks or days, it should not reflect uncontrollable contingencies, estimates for activity duration MUST be derived from objective sources, once activity durations have been identified it may be necessary to revise the network
- The activity duration should be entered besides the appropriate arrow in the network

Boundary Time Calculations

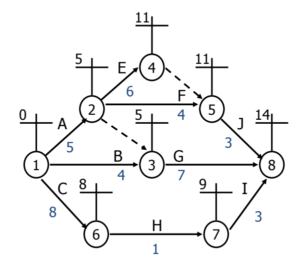
- *earliest start (ES)* the earliest time an activity can begin when all preceding activities are completed as rapidly as possible
- *latest start (LS)* the latest time an activity can be initiated without delaying the minimum project completion time
- *latest finish (LF)* the LS added to the duration (D)
- *total float (TF)* the amount of surplus time allowed in scheduling activities to avoid any interference with any activity on the critical path, the slack between the earliest and latest start times (LS — ES = TF)

Calculating ES

- Make a forward pass through the network adding each activity duration in turn to the ES of the prerequisite activity, dummies are treated exactly the same as other activities
- When a merge is encountered the largest ES + D of the merging activities is the limiting ES for all activities bursting from the event, each limiting ES is recorded on the left bar of the event markers

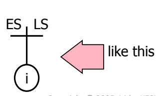


Earliest Start Example



Calculating LS

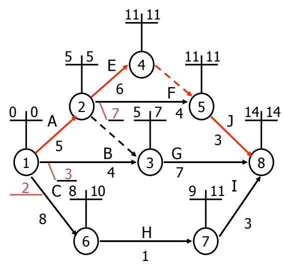
- Basically the reverse of that for ES, make a backward pass through the network subtracting activity durations from the limiting LS at an event
- The limiting LS, the smallest one at a burst event, is entered on the right bar of the cross
- Subsequent LS's are calculated by subtracting activity durations from the LS on the next node cross



Calculating LS (Continued)

- The main difference between LS and ES calculations is that each activity from a common event can have a different LS while all activities starting from the same event have the same ES, to deal with this a *shelf* is added to each activity in a burst that has a larger LF value than the limiting one
- The initial step in LS calculations is to make the right bar of the last cross in the network agree with the left bar, successive subtractions of activity durations from each limiting event should eventually lead to a zero LS for the first network node

Latest Start Example



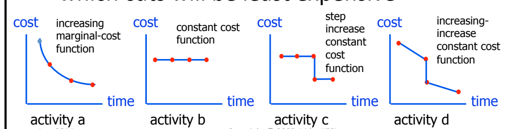
each event with the same entries on the bar is critical

a line connecting these events along activities responsible for the event times is the critical path

Jan. 2014 31

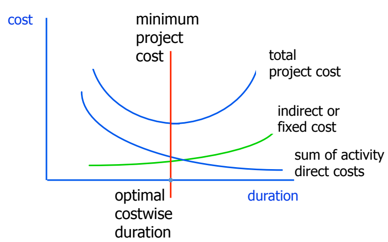
Time-Cost Trade-offs

- An initial network often reveals that a project will take longer than anticipated
- The critical path exposes the group of activities from which cuts should be made to shorten the project but it does not indicate which cuts will be least expensive



Jan. 2014 32

Project Cost Patterns



Jan. 2014 33

Summary: Scheduling

- Project planning and scheduling are essential skills for the software engineer.
- Resource scheduling is a core issue. Critical path scheduling is a simple technique to achieve this, there are many software tools to support it.
- A schedule is only any use if it is realistic and maintained up to date as the project proceeds.

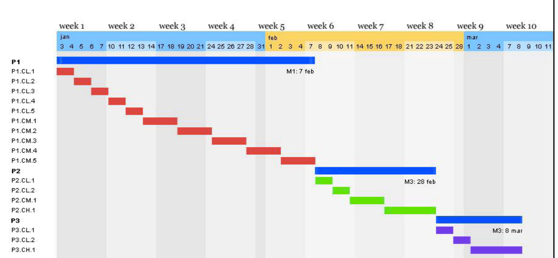
Jan. 2014 34

Other techniques

- Gantt chart
- PERT
- Technique can be enhanced with probabilities and risks

Jan. 2014 35

Gantt chart



Jan. 2014 36

Scheduling in Software development

- Much uncertainty
- Dependencies discovered late
- New tasks discovered late
- Big effort to keep data up-to-date
- OK at a very high level (release planning)

- Needed however when software is a component of a larger system for indentifying dependencies and integration points, and slippage
- Needed for programs and portfolio management