CSIT214/CSIT883
IT Project Management

Risk management

Project management framework (review)

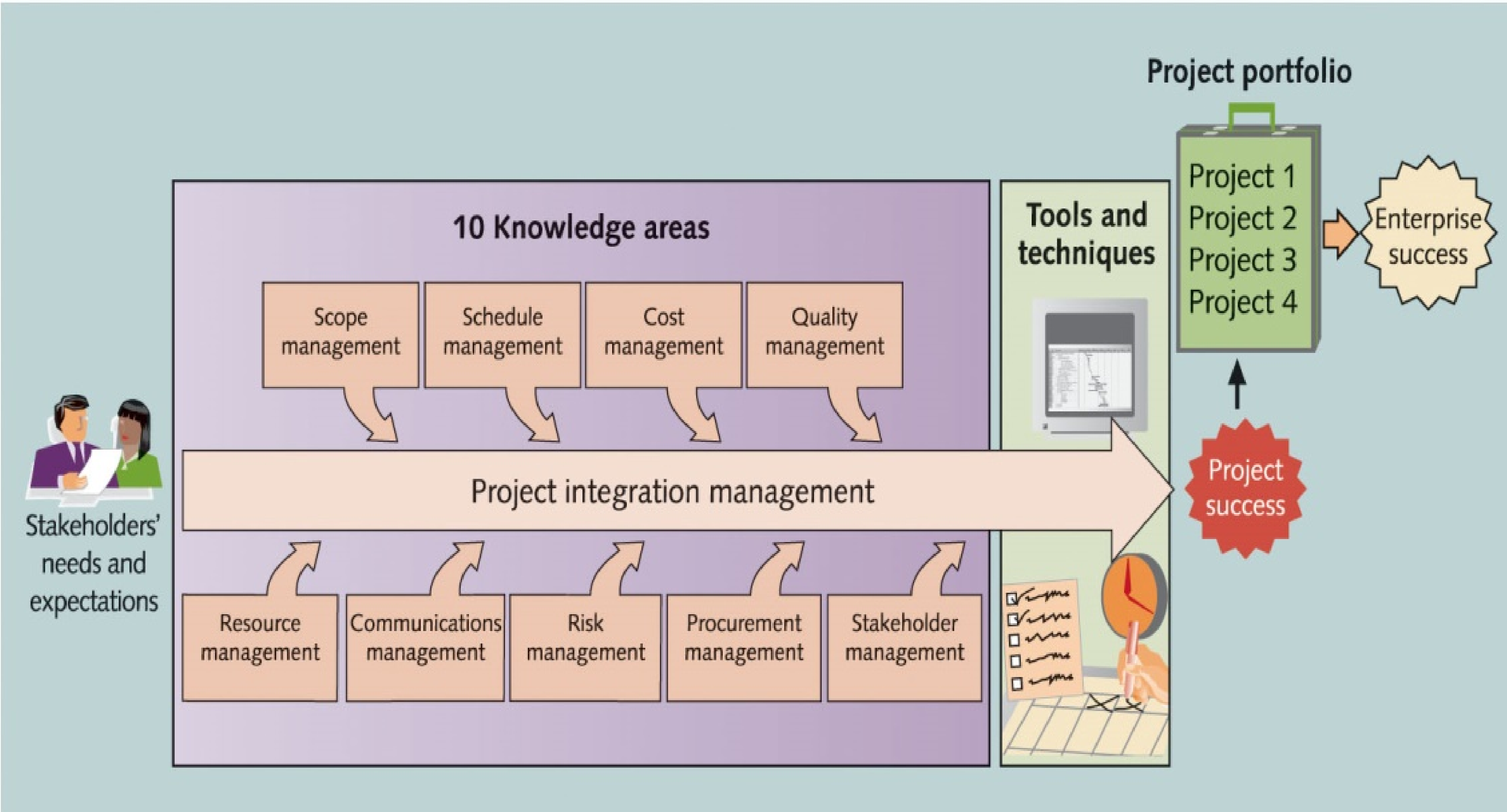


FIGURE 1-2 Project management framework

Risk management

This lecture will touch upon:

- Definition of ‘risk’ and ‘risk management’
- Some ways of categorizing risk
- Risk management
  - Risk identification – what are the risks to a project?
  - Risk analysis – which ones are really serious?
  - Risk planning – what shall we do?
  - Risk monitoring – has the planning worked?
- We will also look at PERT risk and critical chains

Some definitions of risk

'the chance of exposure to the adverse consequences of future events' - PRINCE2

'an uncertain event or condition that, if it occurs, has a positive or negative effect on a project's objectives' - A Guide to the Project Management Body of Knowledge ***(***PM-BOK)

- Risks relate to **possible future** problems, not current ones
- They involve a possible **cause** and its **effect**(s)
  - e.g. developer leaves ---> task delayed
  - E.g. inaccurate estimate of effort or inexperienced staff ---> cost over-run

Note that good risk management may identify situations where an unexpected future event might present an opportunity to be exploited e.g. a task being less difficult than expected.

Quiz

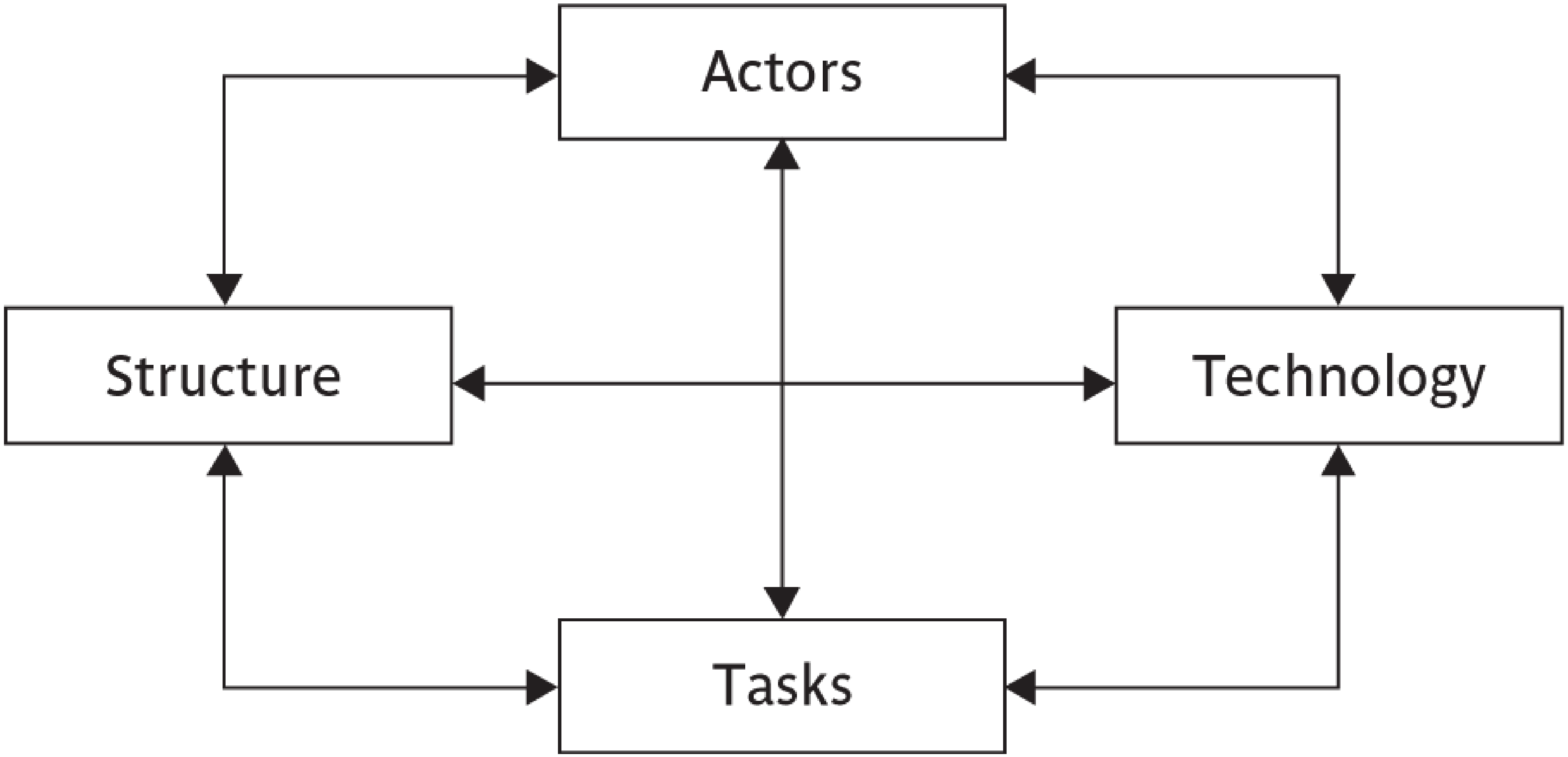
- Match the following causes – a to d – to their possible effects – 1 to 4. The relationships are not necessarily one-to-one. Explain the reasons for each match.
- Causes:

a. staff inexperience;
b. lack of top management commitment;
c. new technology;
d. users uncertain of their requirements

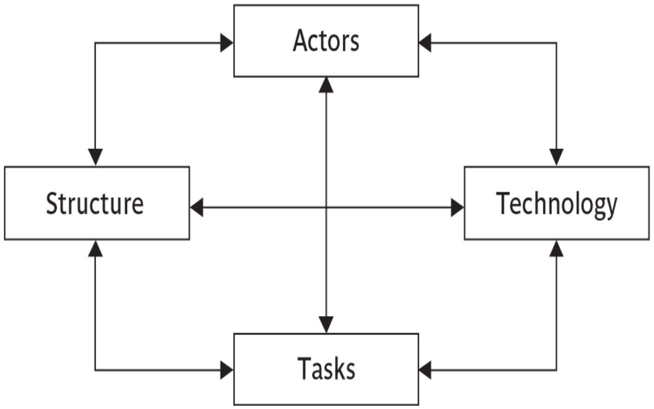
- Effects:

1. testing takes longer than planned;
2. planned effort and time for activities exceeded;
3. project scope increases;
4. time delays in getting changes to plans agreed.

Categories of risk

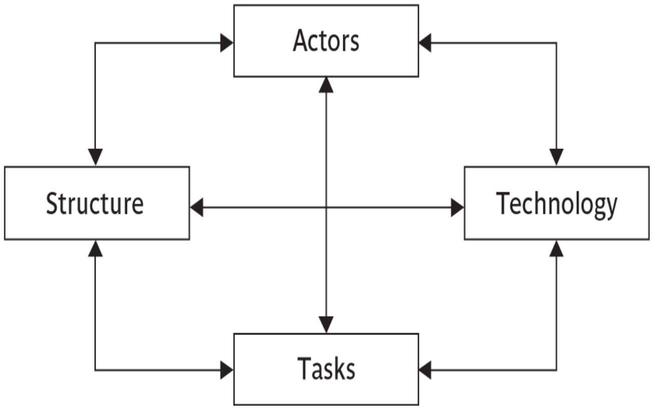


Categories of risk



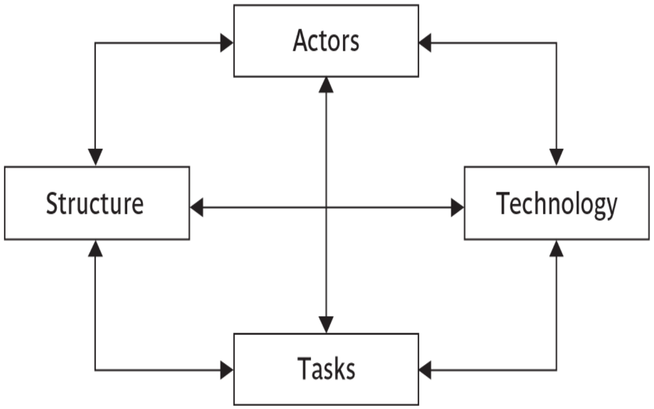
- Actors:
  - relate to all those involved in the project including both developers, users and managers.
  - e.g. a risk could be that high staff turnover leads to information of importance to the project being lost
- Technology:
  - both that used to implement the project and that embedded in the project deliverables
  - E.g. risk could be that the technologies selected are not in fact appropriate.

Categories of risk (cont.)



- Structures:
  - includes management procedures
  - E.g. some team members who need to carry out a particular project task are not informed of this need because they are not part of the project communication network
- Tasks:
  - the work to be carried out.
  - A typical risk is that the amount of effort needed to carry out the task is underestimated.

Categories of risk (cont.)



- Boxes are interlinked: risks often arise from the relationships between factors
  - E.g. between technology and people. If a development technology is novel then the developers might not be experienced in its use
- A risk could belong to more than one area
  - E.g., estimates being wrong could be influenced by problems with structure (over optimism of managers keen to win work).

A framework for dealing with risk

The planning for risk includes these steps:

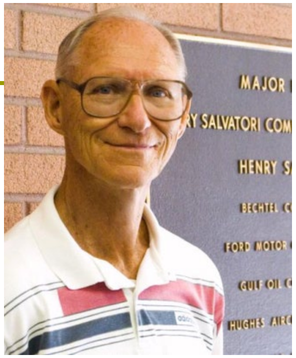
- **Risk identification** – what risks might there be?
- **Risk analysis and prioritization** – which are the most serious risks?
- **Risk planning** – what are we going to do about them?
- **Risk monitoring** – what is the current state of the risk?

Risk identification

Approaches to identifying risks include:

- Use of checklists – usually based on the experience of past projects
- Brainstorming – getting knowledgeable stakeholders together to pool concerns
- Causal mapping – identifying possible chains of cause and effect

Boehm's top 10 software development risks



Barry Boehm surveyed software engineering project leaders to find out the main risks that they had experienced with their projects.

For each risk, some risk reduction techniques has been suggested.

Risk	***Risk reduction techniques***
Personnel shortfalls	Staffing with top talent; job matching; teambuilding; training and career development; early scheduling of key personnel
Unrealistic time and cost estimates	Multiple estimation techniques; design to cost; incremental development; recording and analysis of past projects; standardization of methods
Developing the wrong software functions	Improved software evaluation; formal specification methods; user surveys; prototyping; early user manuals
Developing the wrong user interface	Prototyping; task analysis; user involvement

Boehm’s top ten risk - continued

Risk	***Risk reduction techniques***
Gold plating	Requirements scrubbing, prototyping, design to cost
Late changes to requirements	Change control, incremental development
Shortfalls in externally supplied components	Benchmarking, inspections, formal specifications, contractual agreements, quality controls
Shortfalls in externally performed tasks	Quality assurance procedures, competitive design etc
Real time performance problems	Simulation, prototyping, tuning
Development technically too difficult	Technical analysis, cost-benefit analysis, prototyping , training

Guess what is “God plating”?
-inclusion of features that in fact are unnecessary and which end up never actually being used

Risk analysis prioritization

Risk exposure (RE)

= (potential damage) x (probability of occurrence)

Ideally

Potential damage: a money value

e.g. a project depends on a data centre vulnerable to fire. If a fire occurred, data could be recovered for the cost of $0.5 millions.

Probability: from 0.00 (absolutely no chance) to 1.00 (absolutely certain)

e.g. It is also estimated that 1 in 100 (i.e. 0.01) chance of a fire.

Risk Exposure RE = $0.5m x 0.01 = $5,000

This is crudely analogous to the amount needed for an insurance premium:

- If there were 100 companies chipping in $5,000 each **(risk pooling arrangement)**, there would be enough for the 1 in 100 chance of the fire.

Q & A

Q: What conditions would have to exist for the risk pooling arrangement to work?

A:

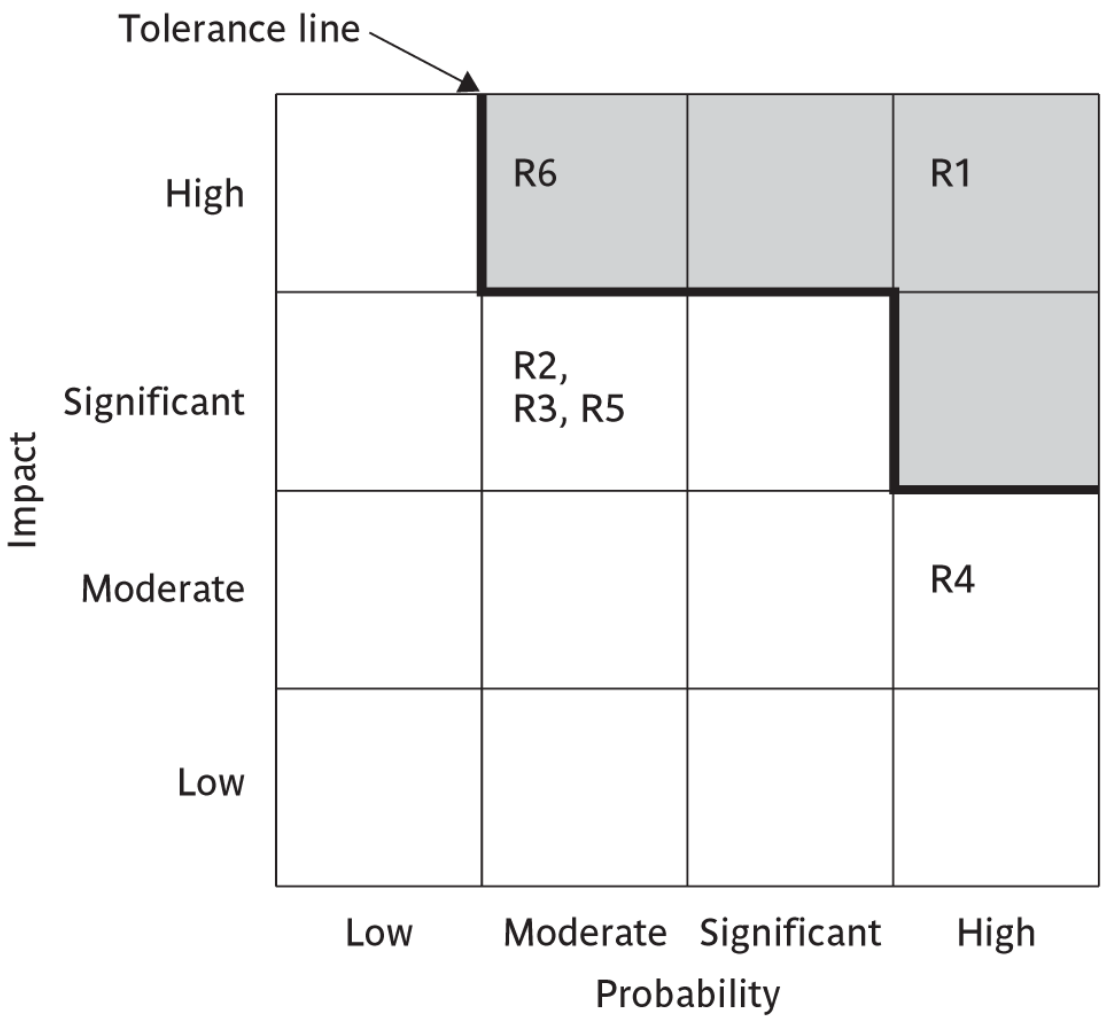
- The **chance is precise**. As this is only an estimate of an average, this could not be guaranteed.
  - Having a larger number of contributors and a larger pool would reduce, but not eliminate, this risk
- The sites would have to be at completely **different locations** so that one site will not affect the others
- Each site has the **same chance**. If the people at a site were aware that the chance was a lot less in their location, they might object to having to subsidize other sites.
- Amount of **damage** caused is always the **same**.

Risk probability: qualitative descriptors

Probability level	*Range*
High	Greater than 50% chance of happening
Significant	30-50% chance of happening
Moderate	10-29% chance of happening
Low	Less than 10% chance of happening

Impact level	*Range*
High	Greater than 30% above budgeted expenditure
Significant	20 to 29% above budgeted expenditure
Moderate	10 to 19% above budgeted expenditure
Low	Within 10% of budgeted expenditure.

Probability impact matrix



- **R1**: Changes to requirements specification during coding.
- **R2**: Specification takes longer than expected.
- **R3**: Significant staff sickness affecting *critical path* activities.
- **R4**: Significant staff sickness affecting *non-critical* path activities.
- **R5**: Module coding takes longer than expected.
- **R6**: Module testing demonstrates errors in design.

In general …

- **Uncertainty** will **lessen** as project progresses
  - more is learnt about user requirements and new technology.
  - This would be reflected in **lower risk probabilities**
- On the other hand, the potential **damage** will tend to **increase**
  - If you type a report without backing it up, as each day adds more text to the report, it also adds to the number of days needed to re-type.
- **Need to frequently <u>reassess</u> risk exposure.**

A framework for dealing with risk (revisit)

The planning for risk includes these steps:

- **Risk identification** – what risks might there be?
- **Risk analysis and prioritization** – which are the most serious risks?
- **Risk planning** – what are we going to do about them?
- **Risk monitoring** – what is the current state of the risk?

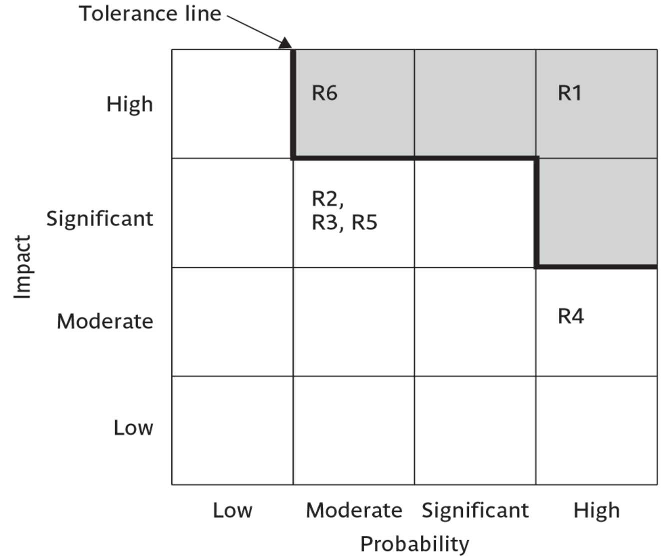
Risk planning

Risks can be dealt with by:

- Risk acceptance
- Risk avoidance
- Risk reduction
- Risk transfer
- Risk mitigation/contingency measures

Risk planning (cont.)

- Risk acceptance:
  - Do nothing ☺.
  - The cost of avoiding the risk may be greater than the actual cost of the damage that might be inflicted.
  - This decision should be taken during risk prioritization.



Risk planning (cont.)

- Risk avoidance
  - avoid the environment or situation in which the risk occurs
  - E.g. If you are worried about sharks, then don’t go into the water.
  - E.g. __________?______________ would avoid a lot of the risks associated with software development
- Risk reduction
  - the risk is accepted but actions are taken to reduce **its likelihood (i.e. chance of occurrence)**
  - e.g. _________?_____________ ought to reduce the risk of incorrect requirements

Risk planning (cont.)

- Risk transfer
  - the risk is transferred to another person or organization.
  - The risk of incorrect development estimates can be transferred by ________________?______
- Risk mitigation
  - tries to **reduce the impact** if the risk does occur
  - e.g. ___________ to allow rapid recovery in the case of data corruption

Deciding on the risk actions ….

- For each actual risk, specific actions have to be planned (out of the five generic responses we have just looked at).
- Whatever countermeasures that are considered, they must be cost-effective.
- But how to the assess cost-effectiveness of a risk reduction action?

Risk reduction leverage

Risk reduction leverage =

$(RE_{before} - RE_{after})$/ (cost of risk reduction)

Re_{before} is risk exposure before risk reduction
e.g. 1% chance of a fire causing $200K damage

RE_{after} is risk exposure after risk reduction
e.g. fire alarm costing $500 reduces probability of fire damage to 0.5%

RRL = (1% of $200k)-(0.5% of $200k)/$500 = 2

RRL > 1.00 therefore worth doing

Risk Register

<table>
<tr><td colspan="8">RISK RECORD</td></tr>
<tr><td>Risk id</td><td></td><td colspan="2">Risk title</td><td colspan="4"></td></tr>
<tr><td>Owner</td><td></td><td colspan="2">Date raised</td><td></td><td>Status</td><td colspan="2"></td></tr>
<tr><td colspan="8">Risk description

</td></tr>
<tr><td colspan="8">Impact description

</td></tr>
<tr><td colspan="8">Recommended risk mitigation

</td></tr>
<tr><td colspan="8">Probability/impact values</td></tr>
<tr><td rowspan="2" colspan="2"></td><td rowspan="2">Probability</td><td colspan="5">Impact</td></tr>
<tr><td colspan="2">Cost</td><td colspan="2">Duration</td><td>Quality</td></tr>
<tr><td colspan="2">Pre-mitigation</td><td></td><td colspan="2"></td><td colspan="2"></td><td></td></tr>
<tr><td colspan="2">Post-mitigation</td><td></td><td colspan="2"></td><td colspan="2"></td><td></td></tr>
<tr><td colspan="8">Incident/action history</td></tr>
<tr><td>Date</td><td colspan="2">Incident/action</td><td colspan="2">Actor</td><td colspan="3">Outcome/comment</td></tr>
<tr><td>

</td><td colspan="2"></td><td colspan="2"></td><td colspan="3"></td></tr>
</table>

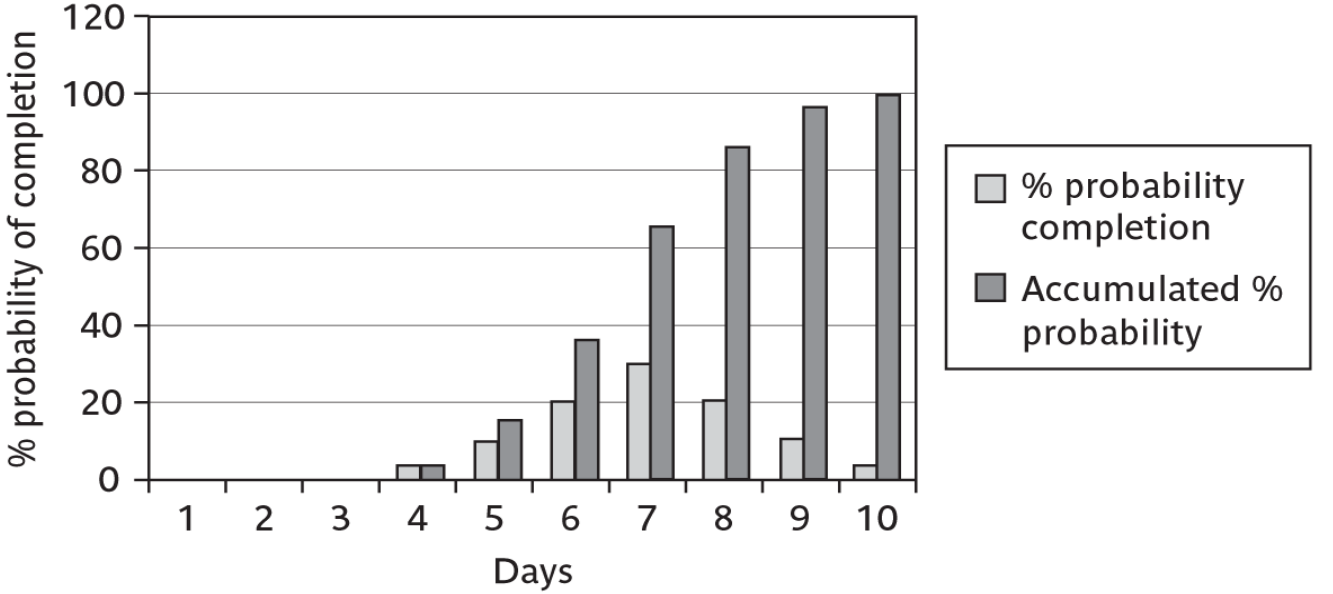
Pen and paper exercise

Jason is a systems analyst who is gathering requirements for an application which will record details of the training undertaken by fire-fighters in the client fire brigade. Details of the training units successfully completed by fire-fighters are to be input to the application by trainers who are themselves senior and active fire-fighters. James needs to interview a trainer to obtain his/her requirements. Because of the senior fire-fighters' other duties, the interview has to be arranged two weeks in advance. There is then a 20% chance of the fire-fighter being unable to attend the interview because of an emergency call-out. Each week that the project is delayed costs the fire brigade approximately $1000.

1. Provide an estimate of the risk exposure (as a financial value) for the risk that the senior fire-fighter might not be able to attend at the times needed.
2. Suggest possible risk mitigation actions.

Evaluating risks to schedule

- Tasks may be completed longer or shorter than what was estimated.



- An estimate of activity duration is realistically a range of values which are clustered around **the most likely value** and are trailing off on either side of this central value.
- How can we take account of this in project planning?

PERT

- **Project Evaluation and Review Technique** (PERT) is a statistical tool used in project management
- Has been applied to very large-scale projects.
- Simplify the planning and scheduling of large and complex projects.
- Incorporate **uncertainty** by making it possible to schedule a project while not knowing precisely the details and durations of all the activities.

Using PERT to evaluate the effects of uncertainty

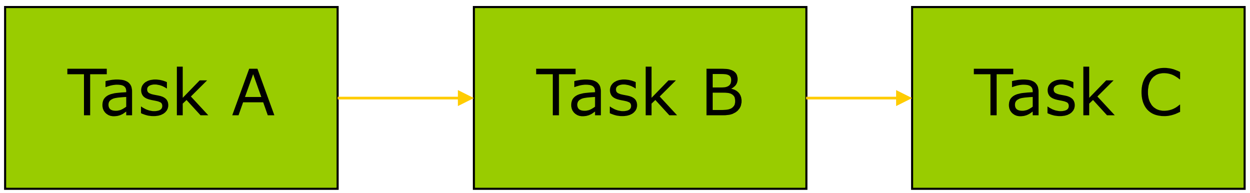
Three estimates are produced for each activity

- *Most likely time (m):* the time we would expect the task to take normally.
  - assuming everything proceeds as normal.
- *Optimistic time (a):* the shortest time that could be realistically be expected.
  - assuming everything proceeds better than is normally expected
- *Pessimistic time (b):* worst possible time
  - assuming everything goes wrong (but excluding major catastrophes)
- **Expected time** (t_e): the best estimate of the time required to accomplish a task, accounting for the fact that things don't always proceed as normal

$$t_e = (a + 4m + b) / 6$$

- **Activity standard deviation: S** = (b-a)/6
  - Can be used to rank activities according to their degree of risk.

A chain of activities



Task	a	m	b	t_e	s
A	10	12	16	?	?
B	8	10	14	?	?
C	20	24	38	?	?

Expected time: t_e = (a + 4m +b) / 6
Activity standard deviation: S = (b-a)/6

A chain of activities

- Carry out a forward pass to compute the expected duration.
- What would be the expected duration of the chain A + B + C?
  - Answer: 12.66 + 10.33 + 25.66 i.e. 48.65
- Also carry out a forward pass to compute the standard deviations.
- What would be the standard deviation for A + B+ C?
  - Answer: square root of $(1^2 + 1^2 + 3^2)$ i.e. 3.32

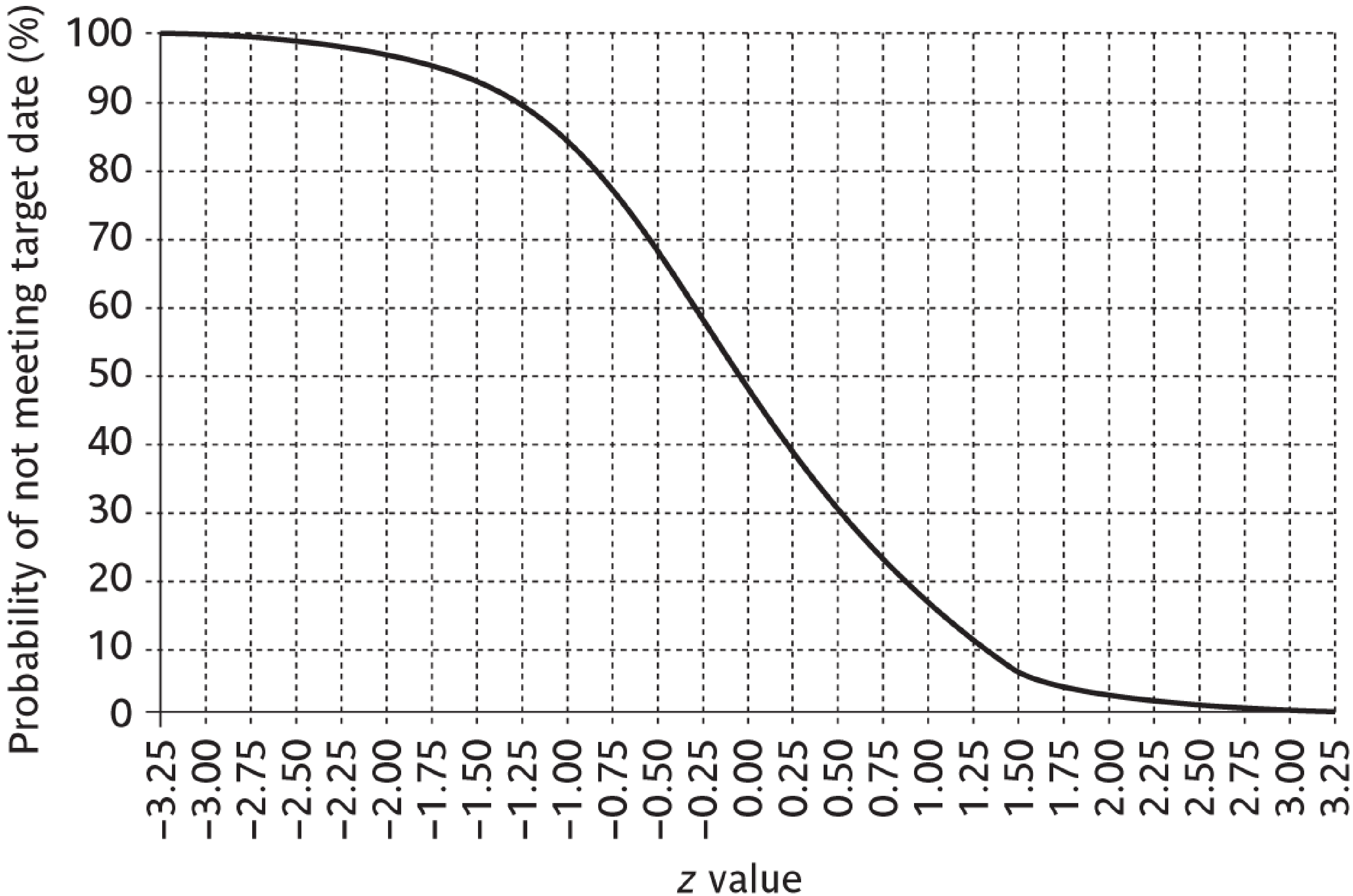
Assessing the likelihood of meeting a target

- Say the target for completing A+B+C was 52 days (T)
- Calculate the z value thus

  $$\mathbf{z = (T - t_e)/s}$$

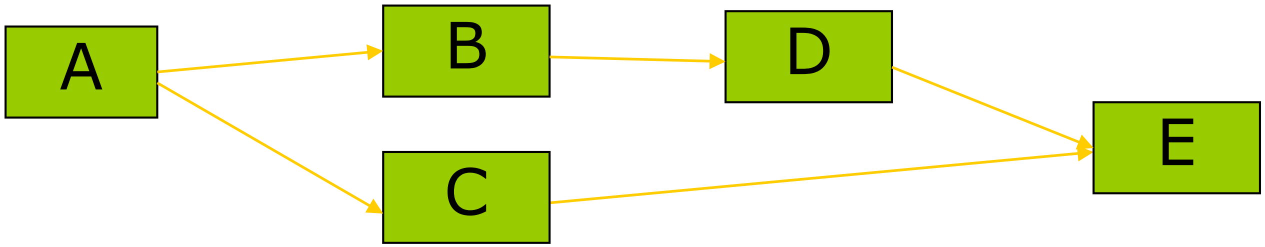
- In this example z = (52-48.65)/3.32 i.e. 1.01
- Look up in table of z values – see next overhead

Graph of z values



There is about a 15% chance of not meeting the target of 52 days.

Pen and paper exercise

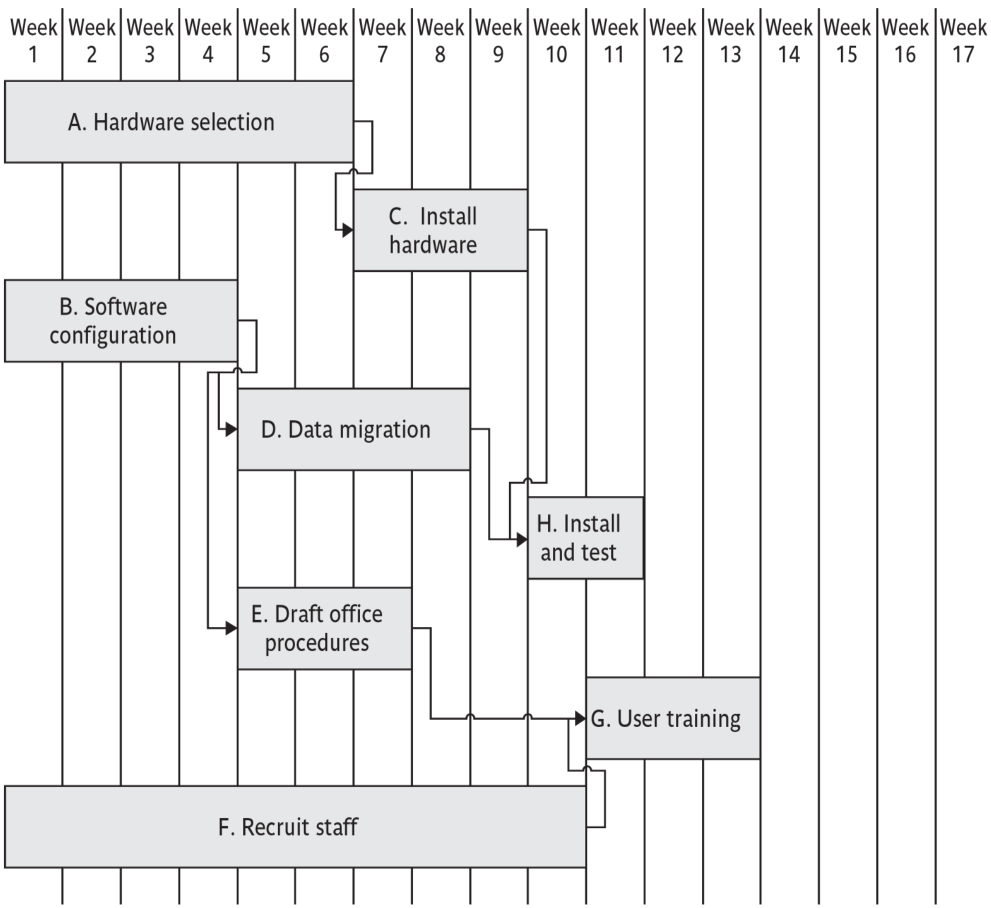


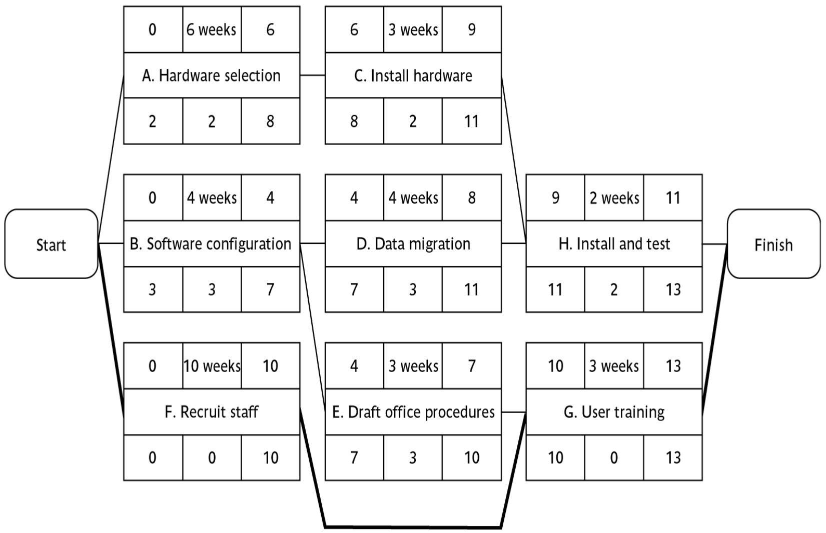
Task	a	m	b	t_e	s
A	4	6	12		
B	6	8	12		
C	10	18	32		
D	30	40	50		
E	7	15	21		

- What is the chance of this project not meeting the target of 80 days?

Critical chain concept

Traditional planning approach





Note: This plan tends to start activities as early as possible – for example see Activities **E and H**.

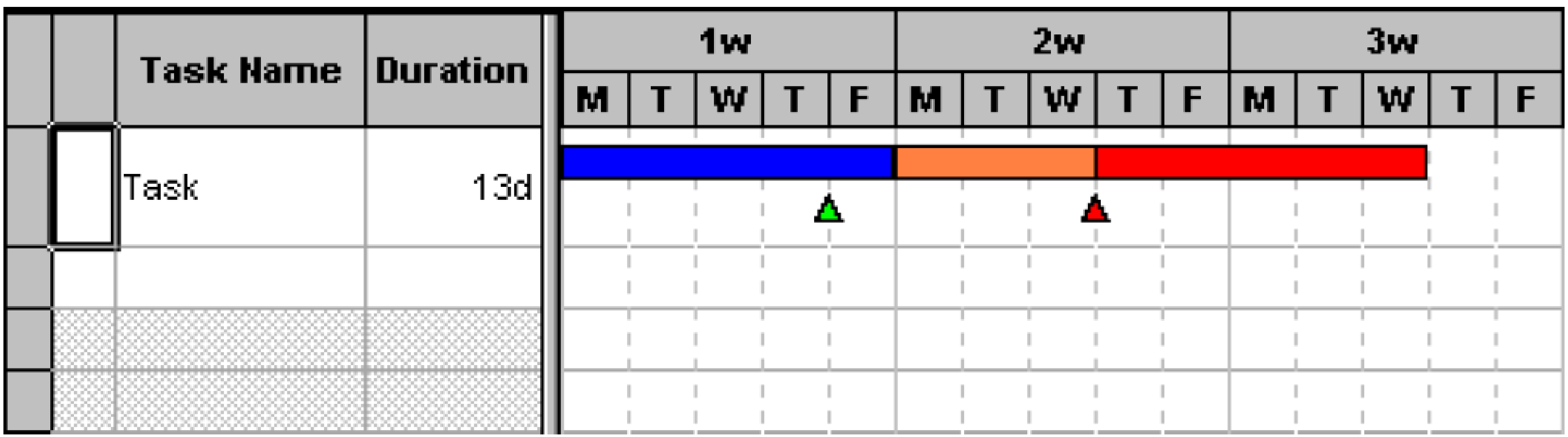
The float on these activities could be used as a buffer if they were late.

Other activities e.g. **F**., in spite of substantial duration, do not have the same kind of cushion. In the case of **F**, the planner would have to make any safety buffer part of the core activity duration.

Critical chain approach

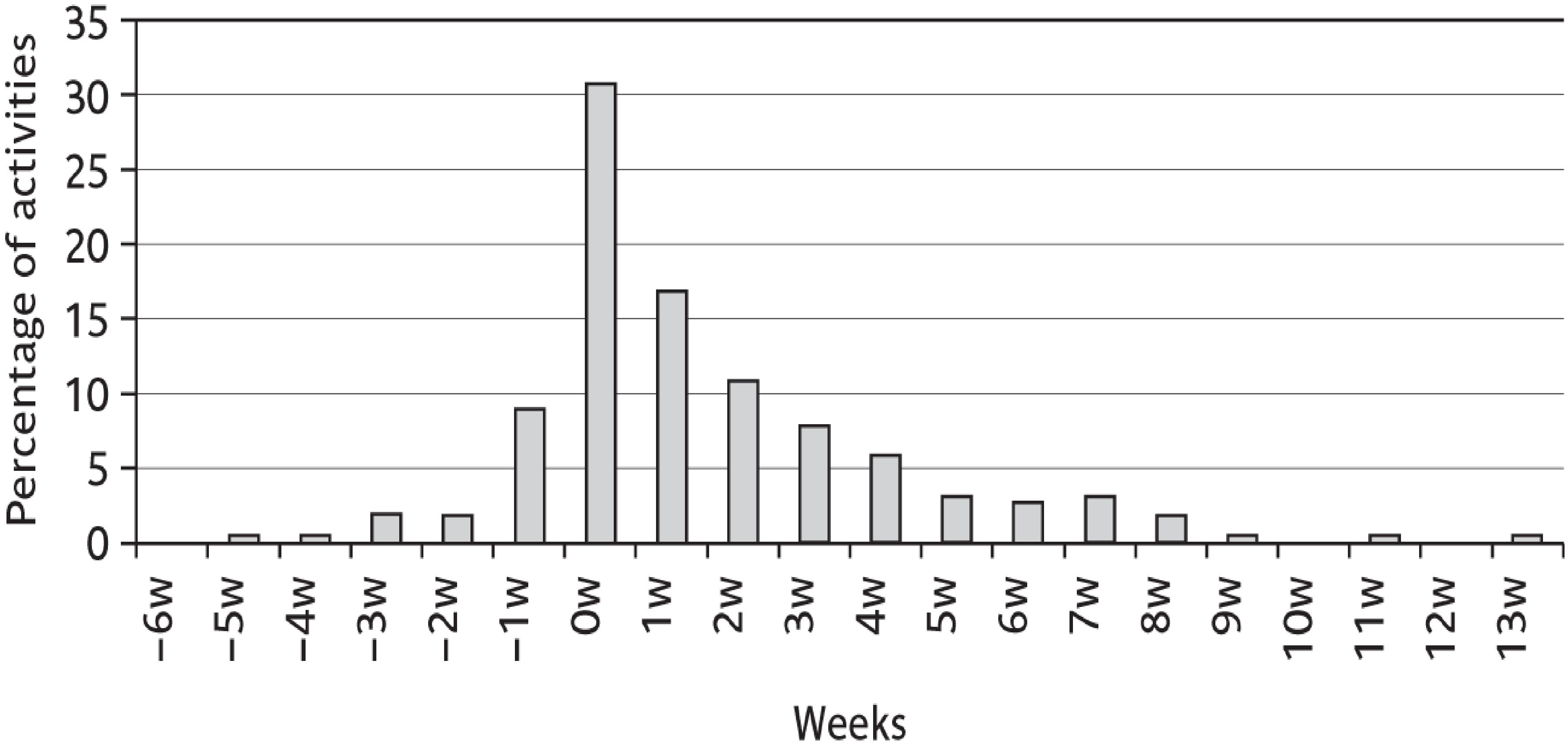
One problem with estimates of task duration:

- Estimators (e.g. Project Manager) add a safety zone to estimate to take account of possible difficulties
  - E.g. Task is estimated to complete in 8 days plus a safety zone of 2 days.
- Developers work to the estimate + safety zone, so time is lost
  - E.g. Developers tend to complete the task in 10 days (rather than 8 days) – "**Student Syndrome**" see http://en.wikipedia.org/wiki/Student_syndrome



- No advantage is taken of opportunities where tasks can finish early – and provide a buffer for later activities

Critical chain approach

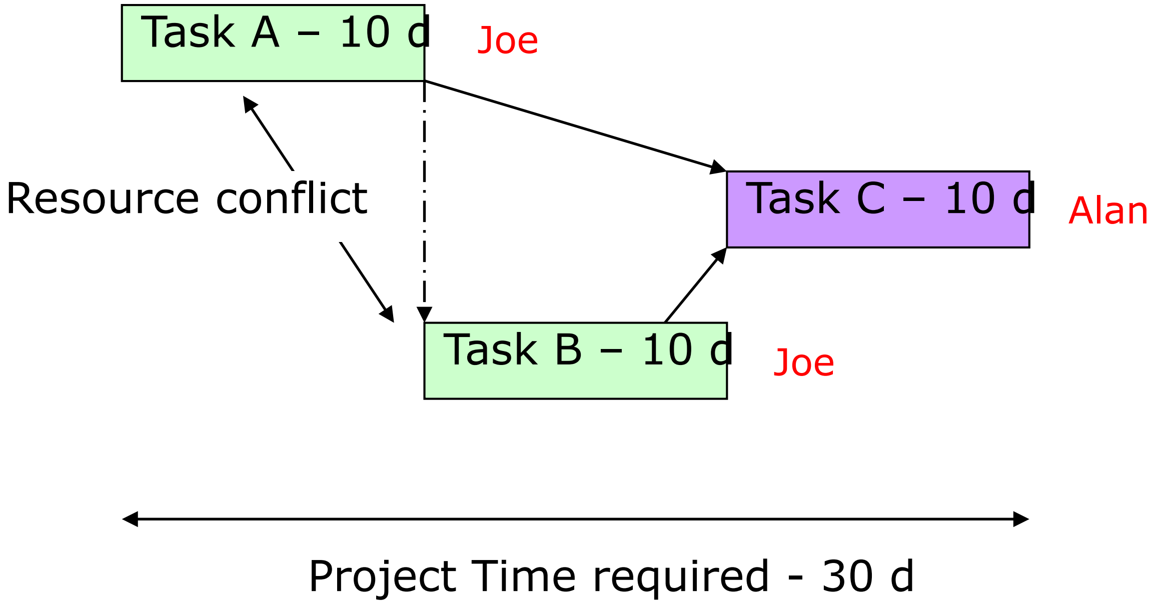


Developers will tend to start activities **as late as** is compatible with meeting the target date as they often have other urgent work to be getting on with in the mean time.

Source: Michiel van Genuchten, "*Why is software late? An empirical study of reasons for delay in software development*", IEEE Transactions in Software Engineering 17(6) 582-90, 1991

Resource dependency

Task C depends on both A and B,
Both A and B need exclusive use of the same resource



Critical chain approach

One answer to this:

1. Ask the estimators for two estimates
   - **Most likely duration**: 50% chance of meeting this
   - **Comfort zone**: additional time needed to have 95% chance

This approach means that the '**safety buffer**' in the estimate for an activity is moved from the individual developer to the project as a whole.

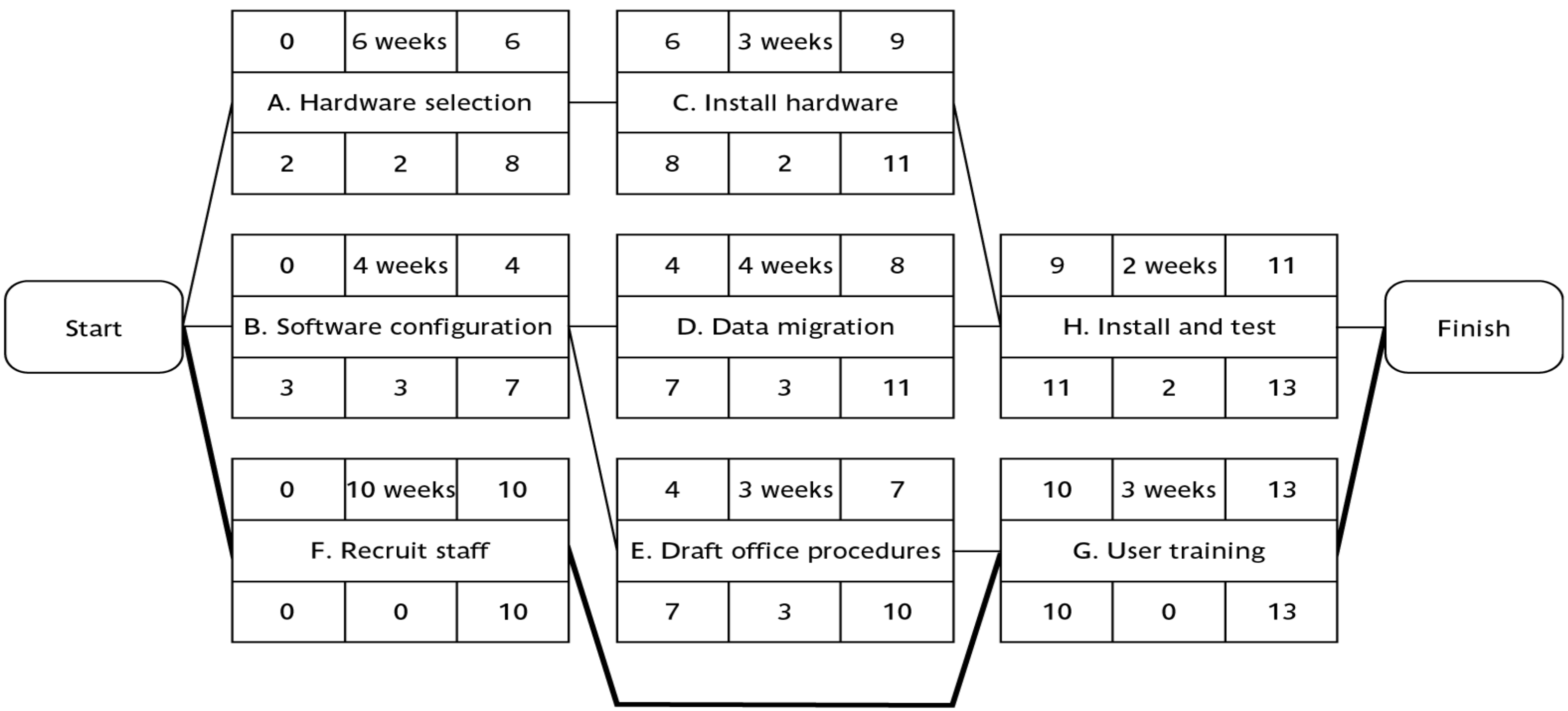
Most likely and comfort zone estimates

Activity	Most likely	Plus comfort zone	Comfort zone
A	6	8	2
B	4	5	1
C	3	3	0
D	4	5	1
E	3	4	1
F	10	15	5
G	3	4	1
H	2	2.5	0.5

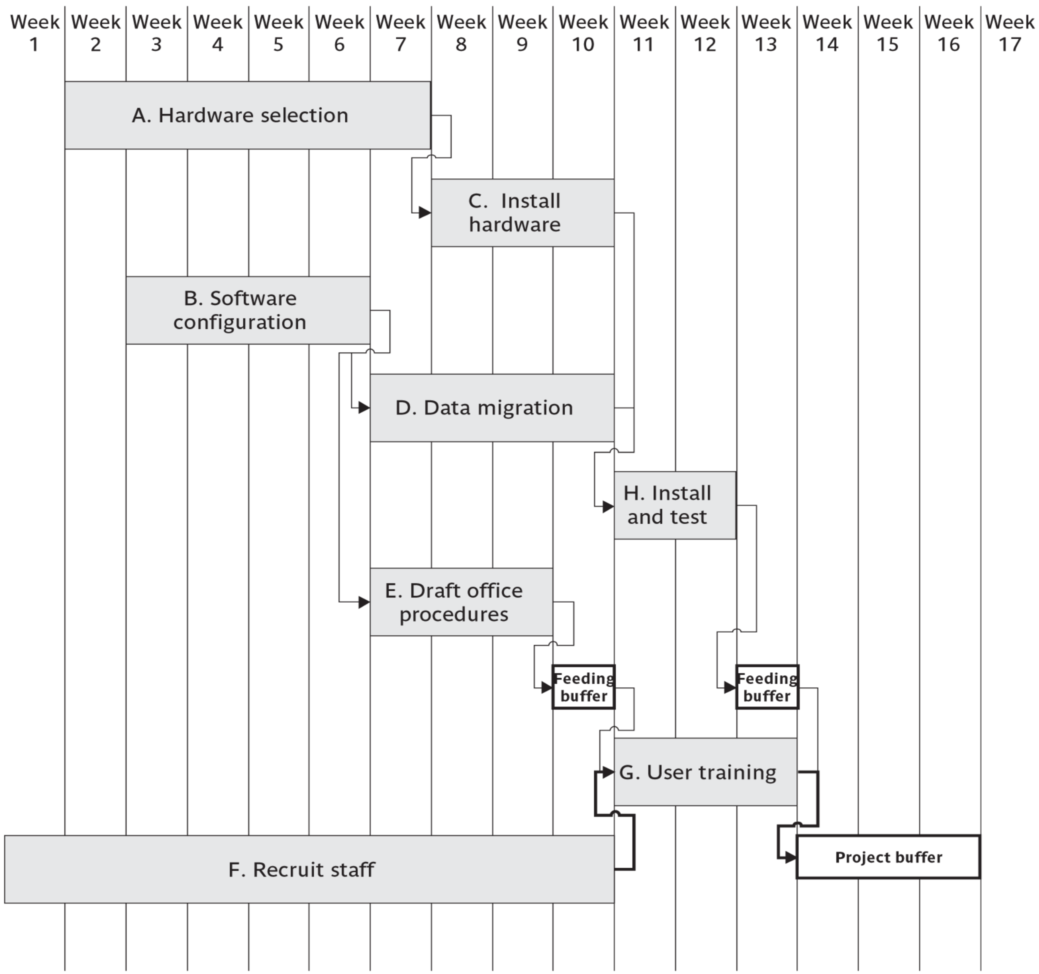
TABLE 7.8 Most likely and comfort zone estimates (days)

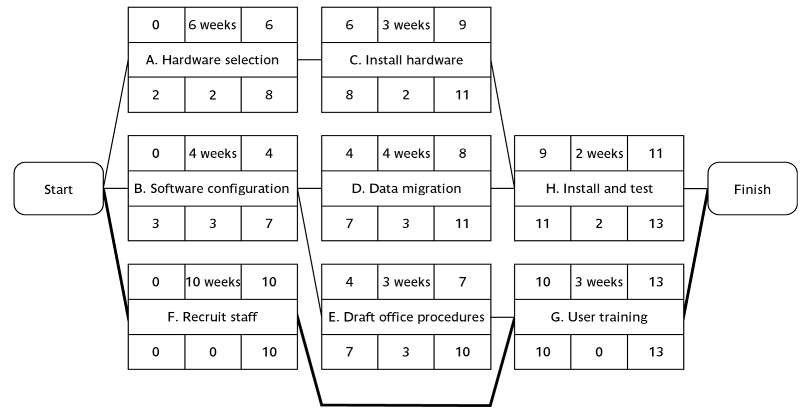
Critical chain approach (cont.)

2. Schedule all activities suing **most likely values** and starting all activities on **latest start dates.**



Scheduling
Gantt chart





Each activity is scheduled to start **as late as possible**

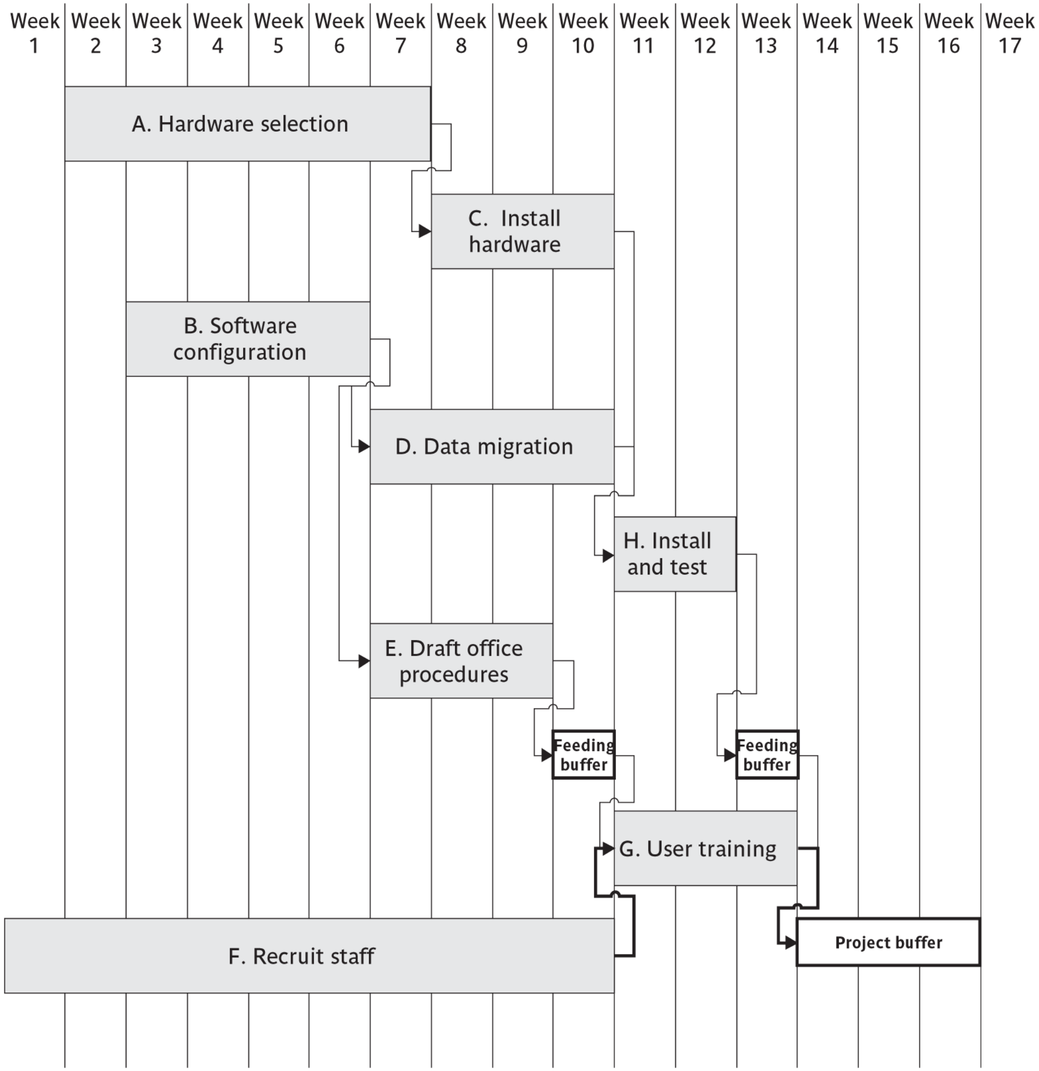
Critical chain - continued

3. Identify the **critical chain** – same a critical path but resource constraints also taken into account
   - The critical chain is the **longest chain** of activities in the project, taking into account both task and **resource dependencies**.
     - A resource dependency is where one activity **has to wait for a resource** (e.g. a human developer) which is being used by another activity to become available.
4. Put a **project buffer** at the end of the *critical chain* with duration **50%** of sum of **comfort zones** of the activities on the **critical chain**.

Critical chain -continued

5. Where **subsidiary** chains of activities feed into critical chain, add **feeding buffer**
6. Set the duration of feeding buffer to be 50% of sum of comfort zones of activities in the feeding chain
   - Where there are **parallel** chains, take the **longest** of the comfort zone total.

Plan employing the critical chain approach



Executing the critical chain-based plan

- No **chain** of tasks is started earlier than scheduled, but once it has started, it is finished **as soon as possible**
- This means the activity following the current one starts **as soon as** the current one is completed, even if this is early – **the relay race principle**

Executing the critical chain-based plan

Buffers are divided into three zones:

- Green: the first 33%. No action required if the project completion dates moves into this zone.
- Amber : the next 33%. An action plan is formulated if the project moves into this zone.
- Red : last 33%. An action plan is executed if the project penetrates into this zone.