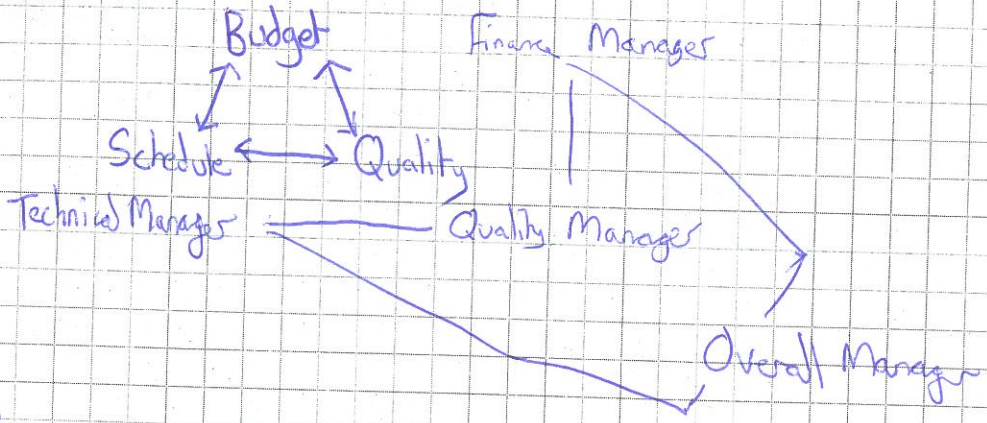
UBJECT: Course 1

DATE:

Software Management Game!



Lecture 2

Total Quality Management (TQM)

Lightweight vs Heavyweight

Agile methods

V shaped model
Detailed plans
Functional specs
Big investment in QA
Formal methods
TQM - CMM for s/w
Prince

Heavyweight QA

International response to the need for quality:

SEI S/W Engineering Institute: Capability Maturity Models

ESI European S/W Institute

OGC Office of Government Commerce: PRINCE projects in controlled environments
PRINCE = Projects In Controlled Environments

SPI S/W Process Improvement ISO9000 SPICE, BOOTSTRAP

ESA European Space Agency S/W Eng standard

MoD QinetiQ

Govt, military, industry all produced standards for software development

Why?
- S/W Pervasive in Society
- Govt is a big customer of software * military, health etc
- Security
- S/W can be an export - economy
- Avoid embarassment for failure
- National infrastructure

SUBJECT: DATE:

IEEE Definitions

Software Quality Assurance

1. A planned and systematic pattern of all actions necessary to provide adequate confidence that an item or product conforms to established requirements

from the customer/other standards — from the customer — not always software, other docs

2. A set of activities designed to evaluate the process by which the products are developed or manufactured.

CMM

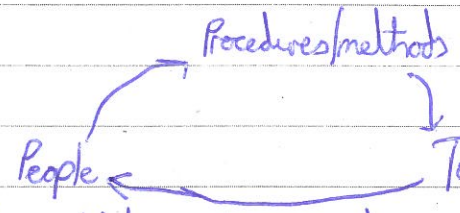
Def 2: TQM/[illegible]-Look at the process, quality of machinery etc

If the process is high quality, the software will be high quality

Difference between software products and factory production.

- S/W error fix cheaper? [illegible]
- [illegible] You make S/W once, it is development
- S/W is development Factory is replication
- In S/W flexibility can be exploited to improve it
- S/W replication is error free
- S/W wears in, Ford Transit wears out.
- Ford Transit is tangible. S/W intangible.

Process



The means by which people, procedures, methods, equipment, and tools are integrated to produce a desired end result.

ECT: G2 DATE:

sts to employ staff
- Equipment
- Training
- Recruitment
- Services
- Holidays
- Sick Pay
- Learning time
- Redundancy pay
- Insurance
- Perks
- Professional Indemnity
- Pensions

varies from company to company
overheads ≡ salary ∴ staff costs = salary x 2

ontracts generally worth 3 or 4 time salaries. Still do it
- Shed risk
- Don't have to hire a team
  → Paying for expertise

N go freelance/contract workers
- Need good CV/portfolio
- Less stable (need to find your own jobs)
- No company benefits (see top), have to pay yourself

L3
IEEE Definition

Development process and maintenance conforms to requirements. Keeping to schedule, budget.

Applying TQM to Software : CMM (Capability Maturity Management)

G3

Contracts

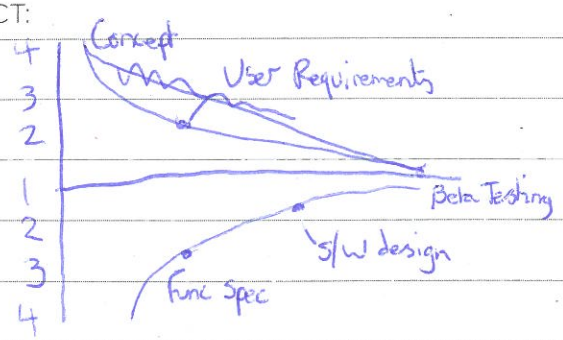
- Fixed Price - better for client, worse for software company (generally)
- Time + Materials - client pays for any setback

however:
- could be a misestimation of price

Can be a combination of these
- eg core features fixed price, extras using T+M

How companies estimate S/W projects
1. Analogy - used past experience for estimates • Spec may grow later
2. Analogy, inexpert • Complexity could be underestimated • Could be done by a salesman
3. Parkinson's Law - work expands to fill the time available. Estimate made according to resources.
4. Price to win - bid low to get the contract. Loss leader. Inaccurate estimate.
5. Top down - build a model. Ask experts about each component
6. Bottom up - S/W modules + staff jobs are known. Experts give detailed estimates. [Time + money]
7. Parametric models - COCOMO. Feed in project size, staff numbers etc.
8. Outline

SUBJECT: DATE:



L4 QA

Make measurements. Apply metrics. (Lots of different metrics for all produced documents/software).

Metrics of an Installed System

G4 Discrete Event Simulator

L5 Metrics of an Installed System

Product Metrics, Process Metrics

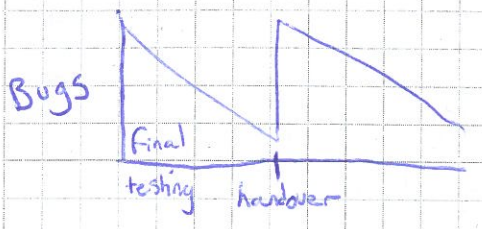
eg Time worked, Lines of Code (output) → productivity, team sizes, days off, Time on work phases

Reliability: the probability that S/W does not fail in a given period of time OR the probability that a given function will perform on demand.
ROCOF (rate of continuous failure): rate at which failures are occuring
MTTR: mean time to repair
MTTF: mean time to failure
MTBF: mean time between failures

$$\text{Availability} = \frac{MTBF}{MTBF + MTTR}$$

ECT: DATE:

vailability: the probability that SW will be functioning at a given future time.



Why?
- New ways of using data
- Combinations of functions
- New platform (H/W and S/W)
- Number of users
- Volume of data
- Date effects (millenium bug)
- Seasonal effects
- Geographic (eg international date line)

ritical Path Analysis

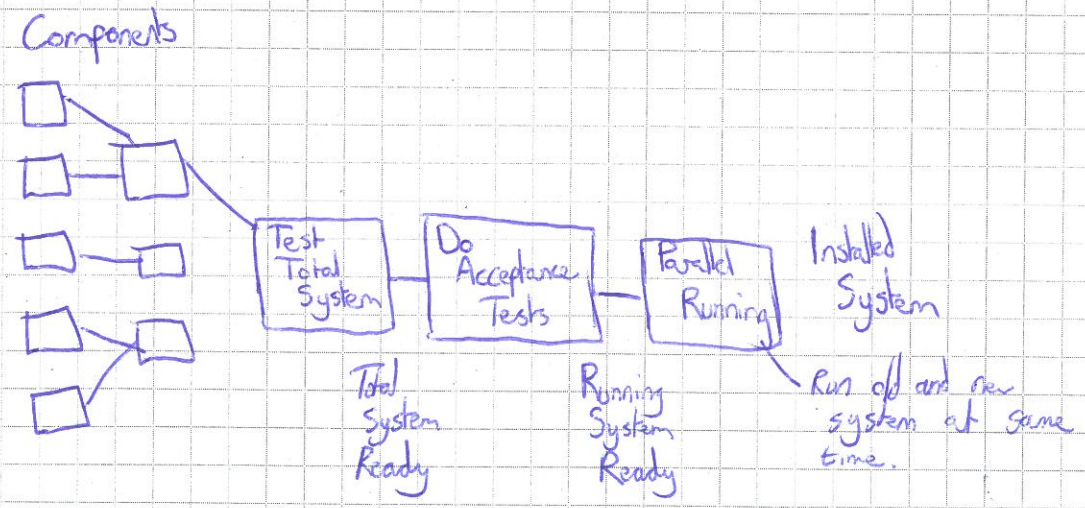
If an activity can be delayed it has x minutes float/slack

A and S/W Projects

- S/W hard to estimate.
- New subtasks/changes in requirements
- Doesn't allow loops → no iteration → no agile methods.
- Suprises
- Development isn't strictly sequential → CPM is sequential.

CPM has an extension to allow overlap

Can use 3 different numbers - optimistic, - reasonable, - pessimistic } weighted average $= \frac{a + 4b + c}{6}$



SUBJECT: DATE:

Reuse and Quality

- You could reuse any deliverable item, eg project plan
- What effect on bugs?

Reuse and Metrics

- Reuse libraries
- Component based S/W engineering
- OO techniques promote reuse

Commercial Reuse

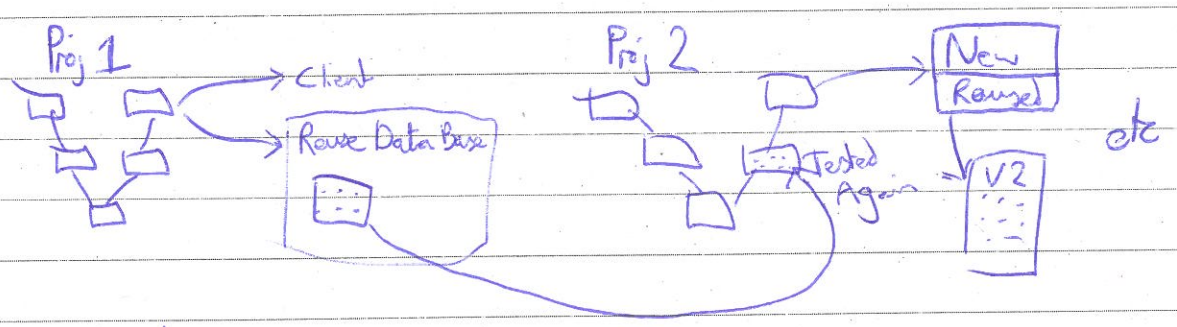
- API's, STL, Java beans etc

Reuse

- Hopefully less bugs
- However you have to find it / adapt it to your needs

Gaffney Model Basis

- If a system consists in part of reused code its quality will probably be higher.
- The reused components have passed through integration and system test again.
- On subsequent reuse this happens again.



Example Model

S/W

6 b/kloc	New
2 b/kloc	Reused

50% each

On avg: 4b/kloc

4b/kloc	40% new
2b/kloc	60% reused

Model Mathematics

Dvr: residual bugs in reused code

Dvn: residual bugs in the new code

Dri = Dvn(1-R) + Dvr(R) where R is the proportion of reused code

ECT: DATE:

$4(.4) + 2(.6) = 1.6 + 1.2 = 2.8$ 6/kloc

IODERN
gile Methods

SAIV

WIN-WIN
by Boehm

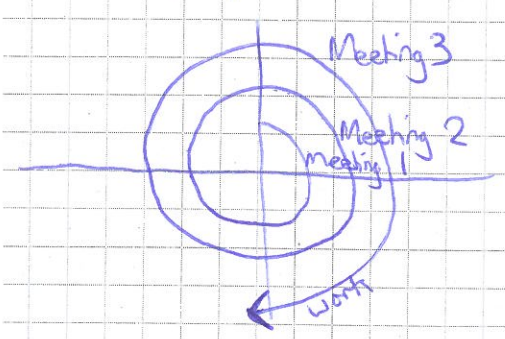
TRADITIONAL
V-SHAPED

3 meetings between all stakeholders > 2 meetings with client (start + end)

akeholdgs : s/w engineer, user, finance, users clients, maintainers of s/w

in-Win Process (Theory W)

in-Win Spiral Model



During Meetings:

Identify ~~up~~ each stakeholders win conditions

Negotiate and agree on what it will do.

Trade offs between parties.

SEE NOTES SERVER

Win-Win Anchor Points

- LCO : Life Cycle Objectives (Functional Spec)
- LCA : Life Cycle Architecture (Part way through)
- IOC : Initial Operational Capability (First Prototype)

Stakeholders commit at these points which drives spiral model.

Risk Management

What risks might cause the S/W Project to go wrong?
We can't eliminate risk and 'minimising' risk is a limited strategy
We accept risk and take the right risks

We don't just react to risk. We are proactive : identify, assess, plan

SUBJECT: DATE:

Development Environment Risks
- Tool support
- Tool quality and documentation
- Tool familiarity
- Best people?
- Staff turnover
- Training
- Enough people?

Estimates?

Lots of users

Level of reuse

Delivery sooner

User reqmts change

Could be 40-60 risks on a big project. Saved in a risk db.

Metrics

Metrics: Probability + Impact (monetary, schedule, quality, reputation, staff morale)

only looking at these!

Estimating Risks
- How probable is it? • What effect would it have? ← Get these correct!
- Consult colleagues to draw up a table

Prioritise! Probability x Impact = Importance

Use the formula carefully. Other aspects could override formula.

eg. • Threshold (can't have X month delay even if lower priority than others)
- Easy to deal with could be done first - bad?
- When the risk occurs - fix sooner risks

ECT: DATE:

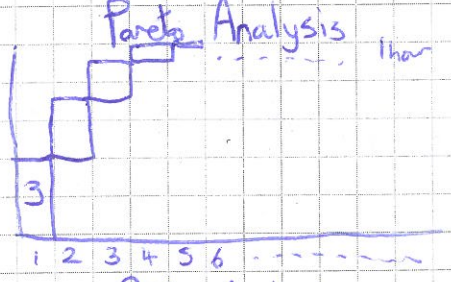
SAIV (Scheduled As Independent Variable)

Schedule dominates over Quality/Price eg Olympics software
Asteroid mining

Trends
- Change is faster
- Schedules are so tight you must release bug-ridden S/W
- Management of 'change' is more important esp. personnel aspects
- Suborn user requirement to cost and schedule

eg if you want to build it in 9 months this is the model to use.

SAIV Approaches
- Build a core capability first
- Prioritise features
- Drop features to meet the schedule



80% of the time covers 20% of the features. Don't do these ones!
Identify areas slowing you down

Extra Optional Features

SAIV and PERT

When you save time on an activity, you save time on the overall schedule only if the activity is on the CP.

CP hard to define because S/W development is complex.

'RAD Opportunity Tree' a tool to help identify and cut the CP
eg Motorola round-the-world testing
(Test software 24 hours a day by doing it in different places)

pwg

SUBJECT: DATE:

Safety Life-Cycle Model for S/W Development (Quality > Time/Schedule)

Like the traditional phase oriented model But

- IDENTIFY HAZARDS
- ASSESS THE RISKS - legislation
  - standards
  - regulation by the safety and regulatory authorities

1 in 10^4 failure probability

(1 in 10^3/10^4 for 'normal' S/W)

The Safety Integrity Level of the S/W System

More safety critical → more rigour (formal methods, extra testing etc)

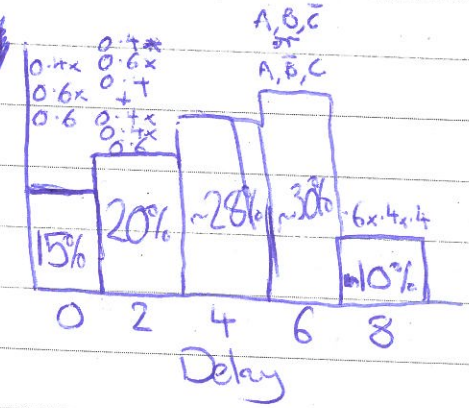
Rank between 1 (very high) to 5 (normal)

Risk Mgmt

Combined effect of Risks

Probability Calculations or Monte Carlo Simulation

	Prob	Delay
Risk A	0.6	4 months
Risk B	0.4	2 months
Risk C	0.4	2 months



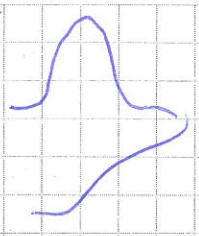
Look back at the risks and mitigate them

Cost-Benefit Trade-off = is it worth it to mitigate the risks

SUBJECT: DATE:

Monte Carlo Simulation

Using Previous Risks/Figures



Risk A 6 digits/10
Risk B 4 digits/10
Risk C 4 digits/10

Run repeated simulations.

Better than probability calculation when lots of risks

Have a Risk Table

Do a Risk Management plan. An RMMM plan.

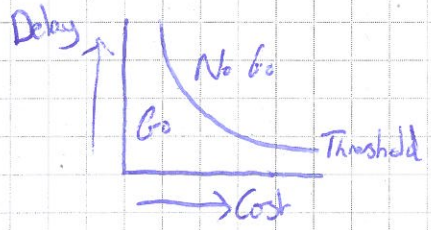
Mitigate, Monitor, Manage

RMMM incurs ~~was~~ extra costs so you must do a cost/benefit tradeoff.

30-40 risks on a larger project. Develop and pursue Rmmm for these risks.

Revise the risk table regularly.

Go/No Go Decisions If delay × cost > threshold you might cancel a project



Risk

£500,000 or gamble for £1 million

99:1	2 ~~gamblers~~ non-gamblers
90:10	10 ~~gamblers~~ non-gamblers
60:40	1 gambler

Prob × Impact

At 60:40 $1 \times 500{,}000 = £500{,}000$
$\cdot 6 \times 1{,}000{,}000 = £600{,}000$ this is mathematically better,

but the £500,000 has minimum risk exposure. Managers like the minimise risk.

If the impact is big, this alters the choice. eg £ billion company would take the gamble.

If the change goes past a threshold, eg delay, £ then risk prioritisation changes

SUBJECT: DATE:

Real-Time Question - Don't answer the question.
Talk about formal/semi-formal methods used to analyse report/plan strategy.
CPA / Cocomo etc

Advantages for free lance contractor
- More money, more freedom (pick + choose what you want to do, place, type of work etc)

What is the safety critical lifecycle of S/W Development
Hazard analysis eg Nuclear v. Hazardous 1 ↑
2
3
4
5 More hazardous
Video Games

More hazardous → more testing
V shape model
Formal methods
} More rigour

Safety critical lifecycle: do hazard analysis then choose what to use

3 reasons to subsidise S/W quality (govt) even though we have free market
- Govt are big users of S/W, tax, pension, military, health etc
  so they want high quality
- S/W is national infrastructure (similar to roads), everyone uses it so they want to ensure a high national standard
- Failure is embarassing (eg failure of an NHS system)
- Export economy (IT), defense, media, telecoms

Rule of Three → (Not Examinable) ← Artificial Insemination
Jelinski + Moranda

Will choose best 3 of your answers