Systems Engineering

Lecture 11

Risk and Criticality

Dr. Joanna Bryson
Dr. Leon Watts

University of Bath
Department of Computer Science

Learning Outcomes

- After the lecture and doing the reading, you should be able to:
  - Describe the risk management process and define the concept of risk.
  - Identify common risks in a software project.
  - Perform probabilistic risk analysis
  - Explain what is meant by a critical system, and how risk is incorporated into the development of critical systems.
  - Construct a fault tree analysis diagram.

Risk

- Risk: An unwanted event that may occur with negative consequences
  - Risk exposure = $\sum_i$ Probability(event$_i$) × cost(event$_i$)
- Software Engineering considers 3 main forms of risk

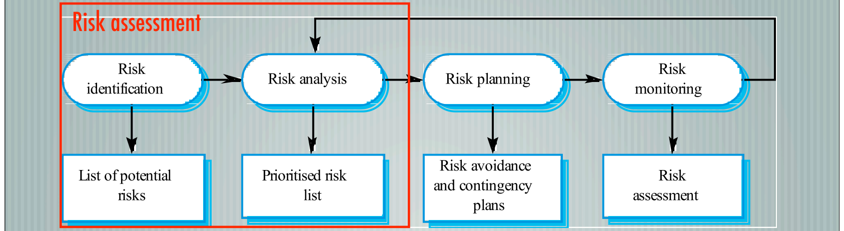
1. Project Risk: cost increase (e.g. schedule slippage)
2. Product Risk: quality degradation.
   - includes Risk of Harm (critical systems).
3. Business Risk: risk to organisation.

Risk Management

- Risk can reflect internal and external factors
- Risk Management:
  - the process of measuring or assessing risk, and then developing strategies to manage the risk.
- Risk Management Process:
  - Risk Assessment
  - Risk Control

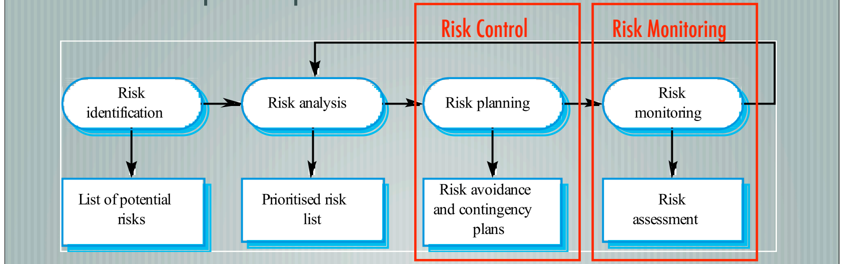
Risk Assessment

- Risk Identification
  - Identify project, product and business risks
- Risk Analysis / Prioritisation
  - Assess the likelihood and consequences of these risks



Risk Control & Monitoring

- Risk planning
  - Draw up plans to avoid or minimise the effects of the risk;
- Risk resolution
  - Implement plans to reduce risk

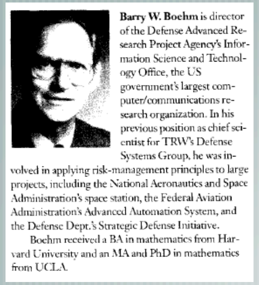


Risk Assessment

- Determine for each risk
  - Loss associated with event (risk impact).
  - Likelihood an event will occur (risk probability).
  - Degree to which we can change the outcome.
- Risk Identification (identify risk events)
  - Case Review / Analogy – read the IT news.
  - Brainstorming
    - Technology risks
    - People risks
    - Organisational risks
    - Requirements risks
    - Estimation risks

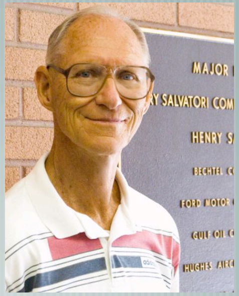
Boehm's Top 10 Risk Items

- Identified through empirical study [Boehm 1991]: Software Risk Management: Principles and Practices, IEEE Software, Vol. 8, Issue 1, pp32-41



Barry W. Boehm is director of the Defense Advanced Research Project Agency's Information Science and Technology Office, the US government's largest computer/communications research organization. In his previous position as chief scientist for TRW's Defense Systems Group, he was involved in applying risk-management principles to large projects, including the National Aeronautics and Space Administration's space station, the Federal Aviation Administration's Advanced Automation System, and the Defense Dept.'s Strategic Defense Initiative.

Boehm received a BA in mathematics from Harvard University and an MA and PhD in mathematics from UCLA.



Boehm's (1991) Top 10 Risks

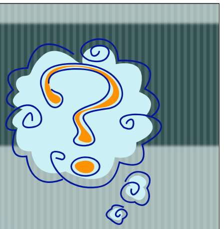
1. Personnel shortfalls
2. Unrealistic schedules and budgets
3. Developing the wrong software functions
4. Developing the wrong user interface
5. Gold plating
6. Continuing stream of requirements changes
7. Shortfalls in externally performed tasks
8. Shortfalls in externally furnished components
9. Real-time performance shortfalls
10. Straining computer science capabilities

Identified through empirical study

Risk Analysis

Risk	Probability	Effects
Organisational financial problems force reductions in the project budget.	Low	Catastrophic
It is impossible to recruit staff with the skills required for the project.	High	Catastrophic
Key staff are ill at critical times in the project.	Moderate	Serious
Software components that should be reused contain defects which limit their functionality.	Moderate	Serious
Changes to requirements that require major design rework are proposed.	Moderate	Serious
The organisation is restructured so that different management are responsible for the project.	High	Serious

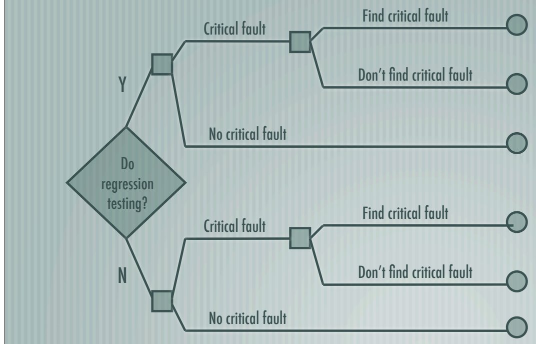
Over to you



- In pairs perform a brief risk assessment (identification and analysis) for your coursework.
  - Pick at least 3 risk events (internal or external) you can foresee.
  - Assess the probability of these occurring.
  - Assess the impact if they do occur.

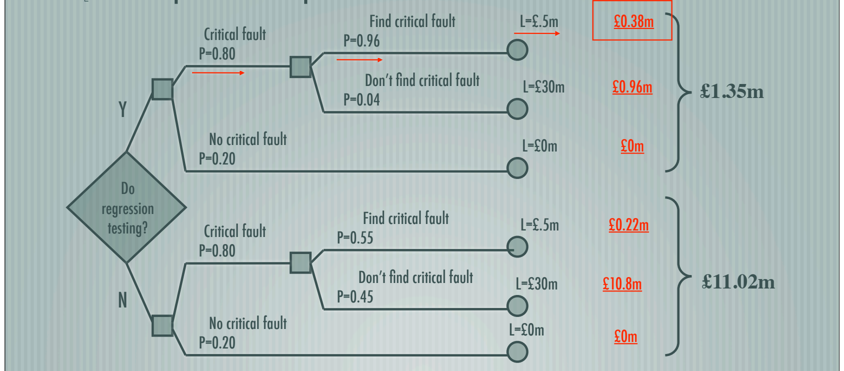
Risk Assessment: Analysis

- Now prioritise risks according to **exposure** (PRA)



Probabilistic Risk Analysis

- Computes risk exposure for various scenarios

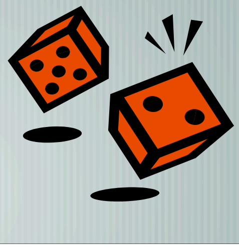


Risk Planning

- Avoidance strategies:
  - reduce the **probability** the risk will arise.
- Minimisation strategies:
  - reduce the **impact** of the risk if it **does** arise.
- Contingency plans:
  - prepared for the worst – have a strategy in place to deal with risks that occur.

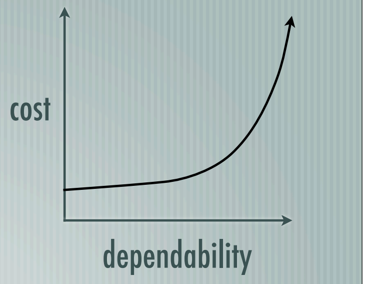
Risk Monitoring

- Risk management is an ongoing, iterative process.
- The risk management process runs hand in hand with your project planning!



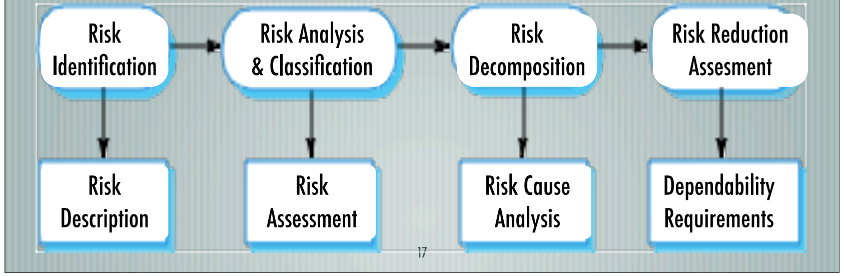
Critical Systems Development

- Three types of critical systems
  1. Safety critical systems
  2. Mission critical systems
  3. Business critical system
- Dependability
  - Availability
  - Reliability
  - Safety
  - Security
- Cost rises exponentially with dependability.



Critical System Specification

- The specification of critical systems should be risk driven
  - Understand risks and identify root causes
  - Define safety requirements to reduce risks
    - Success metrics: should not versus must not.



Safety Critical Systems

- Risks are categorised as:
  - Intolerable. Must never arise or result in an accident.
  - As Low as Reasonably Practical (ALARP). Must minimise the possibility of risk given cost and schedule constraints.
  - Acceptable. The consequences of the risk are acceptable and no extra costs should be incurred to reduce hazard probability.
- The acceptability of a risk is determined by human, social and political considerations.
- In most societies, the boundaries between the regions are pushed upwards with time i.e. society is less willing to accept risk
- Risk assessment can vary by expert (subjective?).
  - Risks are identified as probable, unlikely, etc. This depends on who is making the assessment.

Where to draw the line?

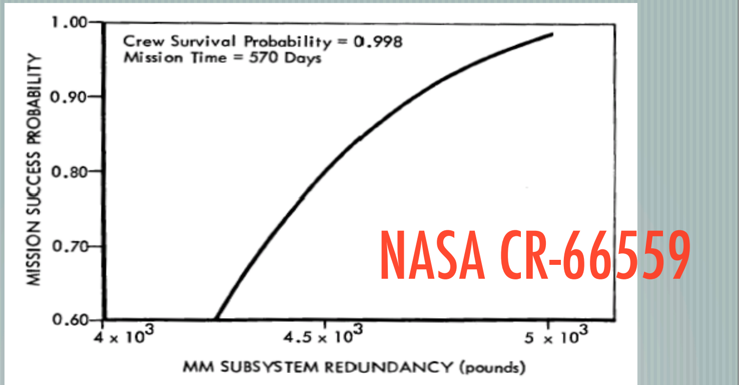


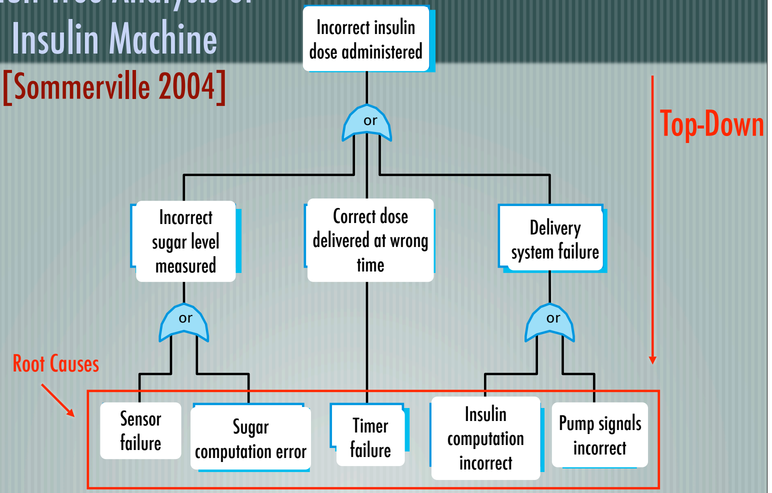
Figure 2.1-11: MM SUBSYSTEMS RELIABILITY

Fault Tree Analysis (FTA)

- Identifying root causes of risk.
  - Developed as part of US MinuteMan Missile program.
- A deductive top-down technique.
- Put the risk or hazard at the root of the tree and identify the system states that could lead to that hazard.
- Where appropriate, link these with ‘and’ or ‘or’ conditions.

Fault Tree Analysis of Insulin Machine

[Sommerville 2004]



Incorrect insulin dose administered

or

Incorrect sugar level measured

Correct dose delivered at wrong time

Delivery system failure

or

or

Top-Down

Root Causes

Sensor failure

Sugar computation error

Timer failure

Insulin computation incorrect

Pump signals incorrect

Critical Systems Design Strategy

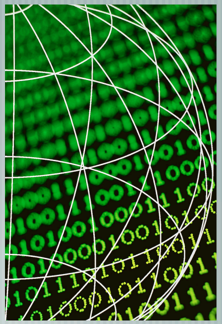
Redundancy

Diversity

- For example, of:
  - Replicated subsystems
  - Replicated internal verification routines

Dependable Programming

- “Dangerous” constructs
  - Goto
  - Floating point numbers
  - Pointers
  - Dynamic memory
  - Concurrency and threads
  - Recursion
  - Unbounded arrays
- Safety enhancing
  - OO Encapsulation
  - Name space management



Fault tolerance (1)

- Run-time fault checking for critical systems
  - Fault free vs. failure free, e.g. RAID.
- Four aspects to fault tolerance
  - Fault detection
  - Damage assessment
  - Fault recovery
  - Fault repair
    - N-version programming, recover blocks, redundant systems, code rewrite(!)

Fault tolerance (2)

- Diversity can be achieved by:
  - Including requirements that different approaches to design be used.
  - Requiring that the implementations should be written in different programming languages.
  - Requiring the use of different tools and development environments.
  - Explicitly requiring different algorithms to be used.

Summary

- After doing some reading, you should be able to:
  - Describe the risk management process and define the concept of risk.
  - Identify common risks in a software project.
  - Perform probabilistic risk analysis
  - Explain what is meant by a critical system, and how risk is incorporated into the development of critical systems.
  - Construct a fault tree analysis diagram.