```
C      REAL MAG
       WRITE(26,101)
101    FORMAT(//,1X,'12TH',5X,'MAG',
     1 8X,'PHASE',8X,'ATT')
       WRITE(25,203)
203    FORMAT(1X,'I/P BR,BB,ES')
       READ(25,200)BR,BB,ES
200    FORMAT(3F)
       DO 1 N=1,12
       AR=BR*N/12.0
       AB=BB*N/12.0
       PCOSH=COSH        *COS(AB)
       QCOSH=SI         N(AB)
       PSINH=S           AB)
       QSINH=C           AB)
       BOTTOM=          )**2+(QCOSH+QSINH)**
       ERR=ES          H)/BOTTOM
       ERJ=-ES         H)/BOTT
       CALL MA         MAG,A
1      WRITE( 6,       ANG,
100    FORMA  (/,
```

Risk Assessment

In a software context

Neil Barnatt

Software still goes wrong today

Issues

- Software keeps going wrong
  - Airplanes
  - Trains
  - Financial systems
- Financial pressure plays a role
- Lack of understanding

Improvements

- IDEs have improved
- Architectural understanding improved
- Design patterns have improved
- Testing is more fully applied

POST OFFICE HORIZON IT INQUIRY

Post Office Horizon IT Inquiry

Report

Volume 1

HC 1119

Reputation/legal

Legal
11 ½m A 320 Chertsey Woking
Gatwick M 23, Dartford & M 20 Guildford A 3 M 25
50
50
50
50

Variable speed limit, M25 by N Chadwick, CC BY-SA 2.0 <https://creativecommons.org/licenses/by-sa/2.0>, via Wikimedia Commons

Collective Commons Wikipedia Chris Gunn - 2021 https://www.flickr.com/photos/nasawebbtelescope/51773093465/

Why assess risk?

Moral	The right thing to do. Society expects safety & things to work
Legal	fines (not insurable), custody, civil suits
Financial	fines, lost income, go out of business
Reputation	could go out of business if customers go elsewhere or public forces closure.

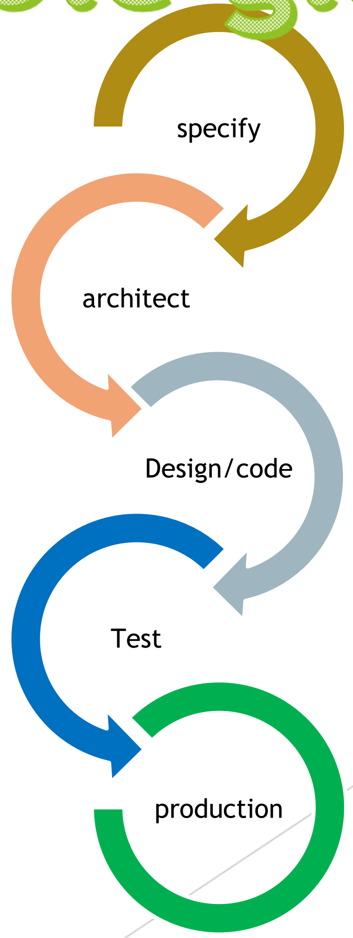
Conceptual software

Software the flexible glue

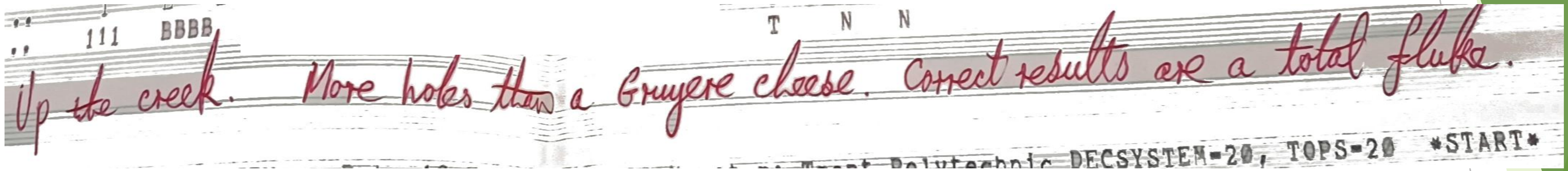
input
- User
- Environment
- Stored data

transform
- Process data in software
- Update current state

activate
- Send transformed input to physical device
- Automatically alter something
- Store the answer



Software error lifecycle

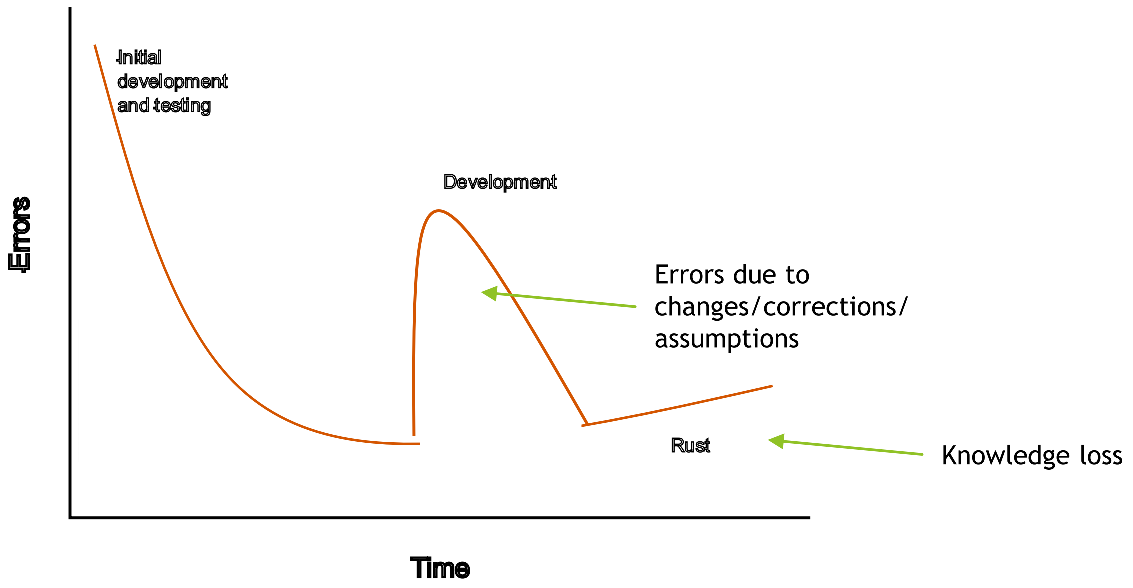


111 BBBB

T N N

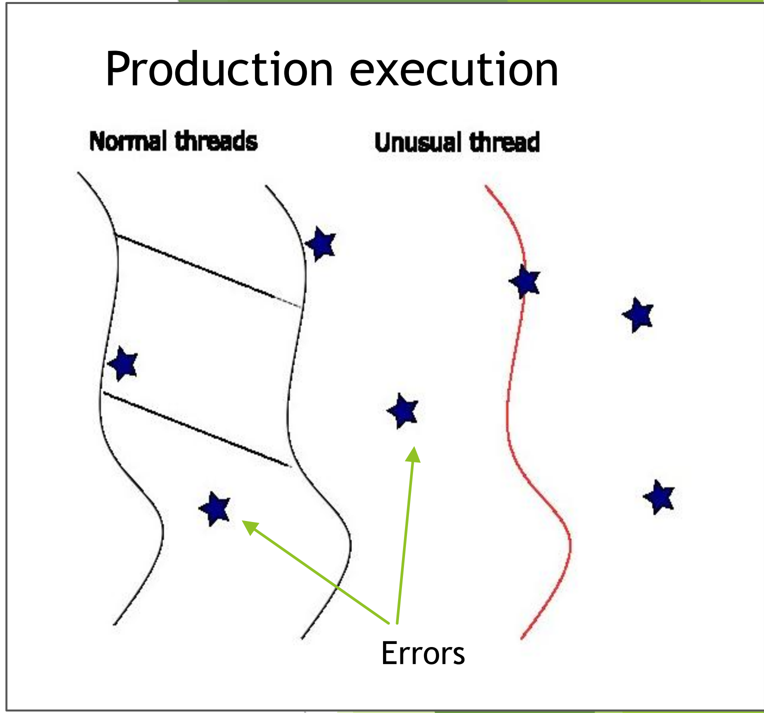
Up the creek. More holes than a Gruyere cheese. Correct results are a total fluke.

Polytechnic DECSYSTEM-20, TOPS-20 *START*



Software errors

- Software and systematic errors
  - When triggered by input conditions they will be executed..time..after time
  - They are latent and unknown
  - The quantity is also unknown
  - Consequence is unknown
- The error density is estimated from the development process or operational experience
  - Gives rise to the idea of Safety Integrity Levels (SIL)
  - However, there is no guarantee it's just a statistical estimate
  - Better processes thought to produce less errors per lines of code



Error association

- Errors tend to be:
  - Associated with complexity
  - Number of parties involved
  - Age of system – sometimes there’s no one left from the original team
  - Frequency of upgrades/fixes/patches $\Rightarrow$ Leads to tension between functionality/quality/safety and cyber security
  - Lack of understanding
  - Lack of testing
  - Interfaces

Problem

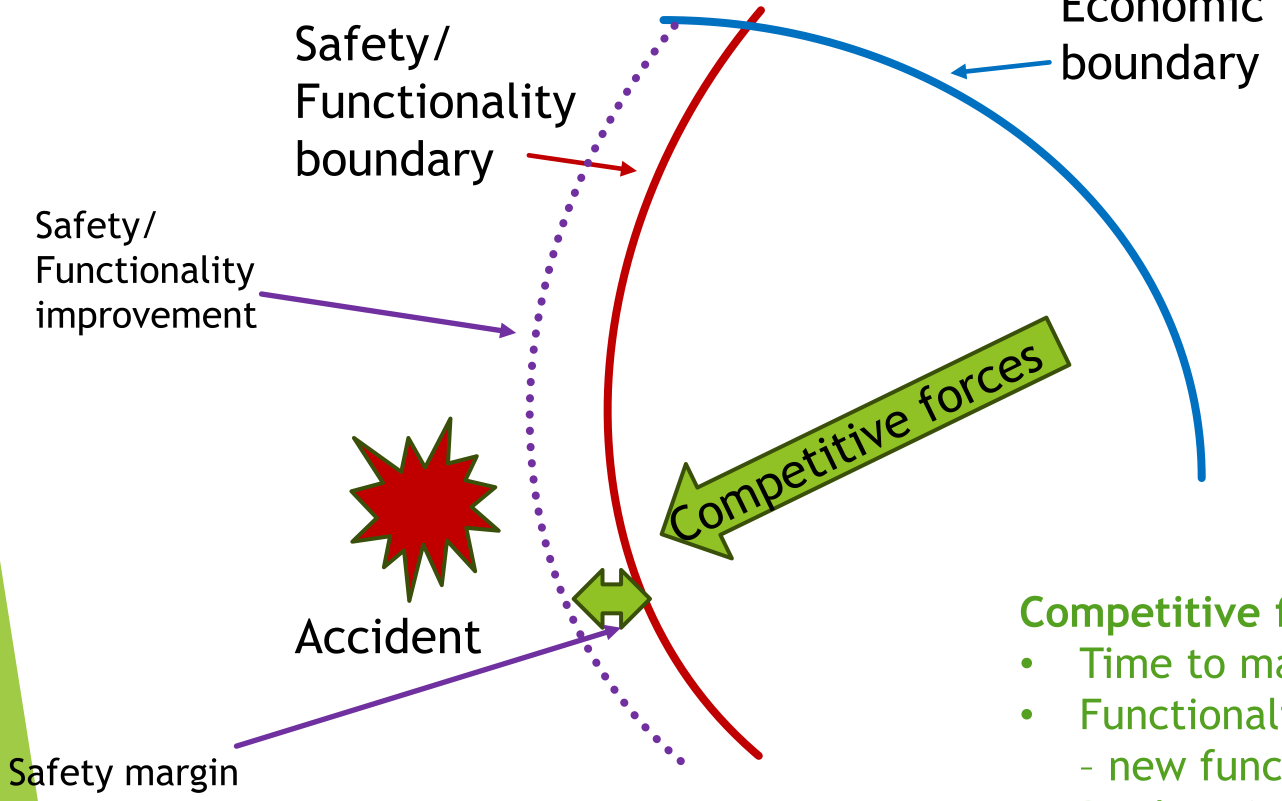
- Software is largely unseen
  - We reason about how it performs
  - What bits are meant to do what
- However, the effects are felt on the product/system as a whole

- Therefore,
  - The primary analysis tends to take place at the system or subsystem level
  - No one cares if the part of the software goes wrong and it has no material effect. But they do care if someone gets injured, they lose money, or reputation

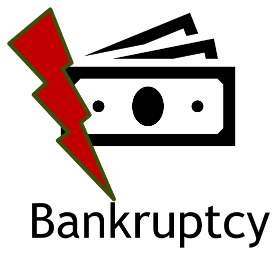
- **A reliable system is not sufficient for safe system.** They are two distinct properties
  - A system can be unreliable and safe
  - A system can be reliable and unsafe
  - Or a system can be reliable and safe

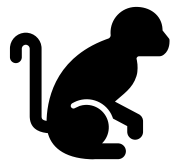
Levenson 2011

The commercial dilemma



Developed from Rasmussen 1997





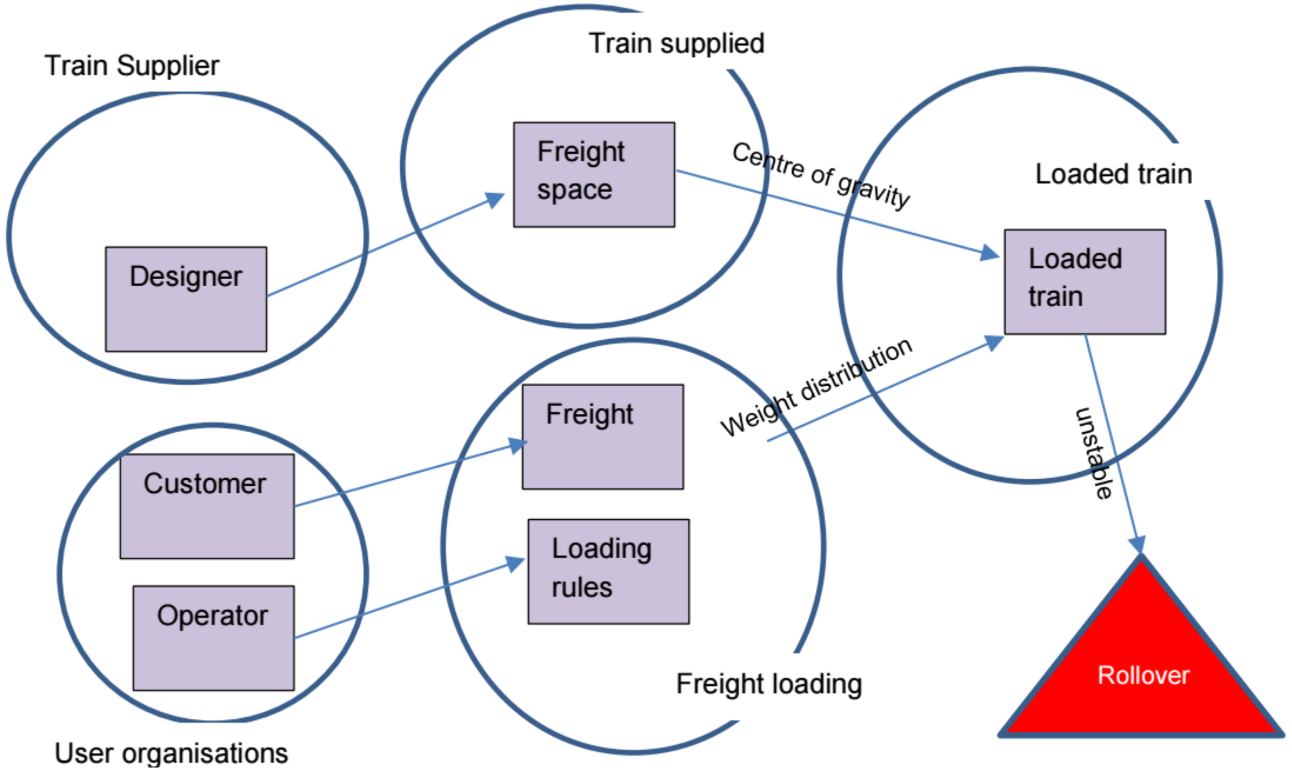
Wilde 1998

Competitive forces

- Time to market
- Functionality – keep relevant – new functions
- Produce it cheaper

Everyone does their bit

CAM-ERD

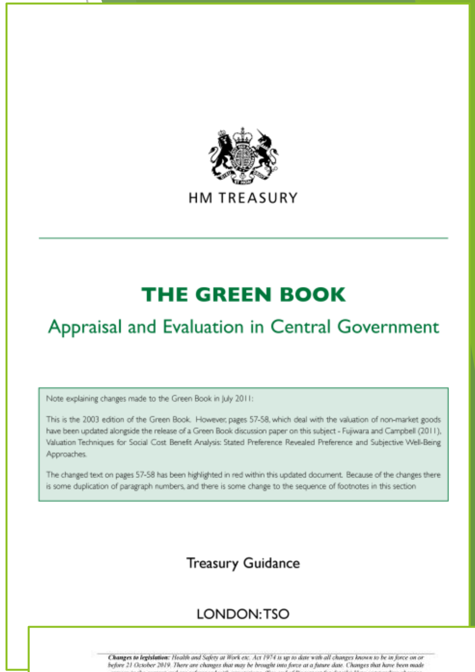


Barnatt 2021 developed from Rasmussen

Safety

- A social concept.
  - An acceptably safe system is one that the level of risk is tolerated by society
  - People decide what is and is not safe – laws and regulation back this up
  - Generally, safety is measured in money because its convenient
    - Value of preventing a fatality is ~£2M in UK rail (based on the Green Book and calculations of economic worth). Note it's a theoretical measure
  - The law does not specify what safety is instead it refers to case law, a concept of gross disproportionality and So Far As Reasonably Practicable as measures
- Proving something is safe is impossible!
  - Would need to prove something is safe in all possible cases
  - A case of the 'All swans are white' falsification
  - Checking something likely to be safe is done by showing its not unsafe.

CC BY-SA 3.0,
https://commons.wikimedia.org/w/index.php?curid=1243220



HM TREASURY

THE GREEN BOOK

Appraisal and Evaluation in Central Government

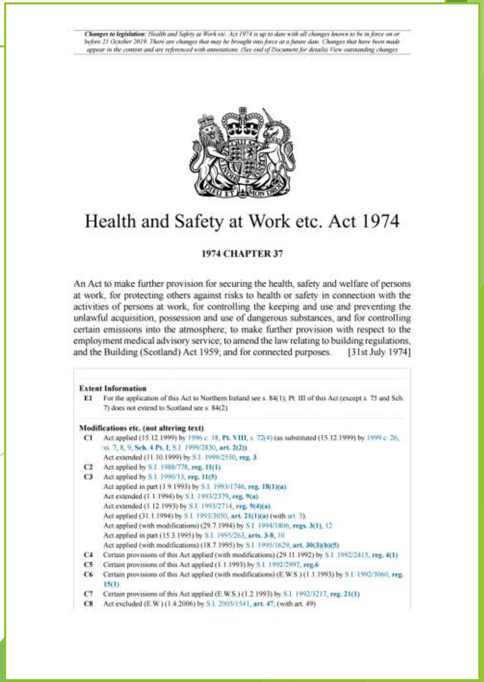
Note explaining changes made to the Green Book in July 2011:

This is the 2003 edition of the Green Book. However, pages 57-58, which deal with the valuation of non-market goods have been updated alongside the release of a Green Book discussion paper on this subject - Fujiwara and Campbell (2011), Valuation Techniques for Social Cost Benefit Analysis: Stated Preference Revealed Preference and Subjective Well-Being Approaches.

The changed text on pages 57-58 has been highlighted in red within this updated document. Because of the changes there is some duplication of paragraph numbers, and there is some change to the sequence of footnotes in this section

Treasury Guidance

LONDON:TSO



Changes to legislation: Health and Safety at Work etc. Act 1974 is up to date with all changes known to be in force on or before 21 October 2019. There are changes that may be brought into force at a future date. Changes that have been made appear in the content and are referenced with annotations. (See end of Document for details) View outstanding changes

Health and Safety at Work etc. Act 1974

1974 CHAPTER 37

An Act to make further provision for securing the health, safety and welfare of persons at work, for protecting others against risks to health or safety in connection with the activities of persons at work, for controlling the keeping and use and preventing the unlawful acquisition, possession and use of dangerous substances, and for controlling certain emissions into the atmosphere; to make further provision with respect to the employment medical advisory service; to amend the law relating to building regulations, and the Building (Scotland) Act 1959; and for connected purposes. [31st July 1974]

Extent Information

E1 For the application of this Act to Northern Ireland see s. 84(1); Pt. III of this Act (except s. 75 and Sch. 7) does not extend to Scotland see s. 84(2)

Modifications etc. (not altering text)

C1 Act applied (15.12.1999) by 1996 c. 18, **Pt. VIII**, s. 72(4) (as substituted (15.12.1999) by 1999 c. 26, ss. 7, 8, 9, **Sch. 4 Pt. I**; S.I. 1999/2830, **art. 2(2)**)
Act extended (11.10.1999) by S.I. 1999/2550, **reg. 3**

C2 Act applied by S.I. 1988/778, **reg. 11(1)**

C3 Act applied by S.I. 1990/13, **reg. 11(5)**
Act applied in part (1.9.1993) by S.I. 1993/1746, **reg. 18(1)(a)**
Act extended (1.1.1994) by S.I. 1993/2379, **reg. 9(a)**
Act extended (1.12.1993) by S.I. 1993/2714, **reg. 9(4)(a)**
Act applied (31.1.1994) by S.I. 1993/3050, **art. 21(1)(a)** (with art. 3)
Act applied (with modifications) (29.7.1994) by S.I. 1994/1806, **regs. 3(1)**, 12
Act applied in part (15.3.1995) by S.I. 1995/263, **arts. 3-8**, 10
Act applied (with modifications) (18.7.1995) by S.I. 1995/1629, **art. 30(3)(b)(5)**

C4 Certain provisions of this Act applied (with modifications) (29.11.1992) by S.I. 1992/2415, **reg. 4(1)**

C5 Certain provisions of this Act applied (1.1.1993) by S.I. 1992/2997, **reg.6**

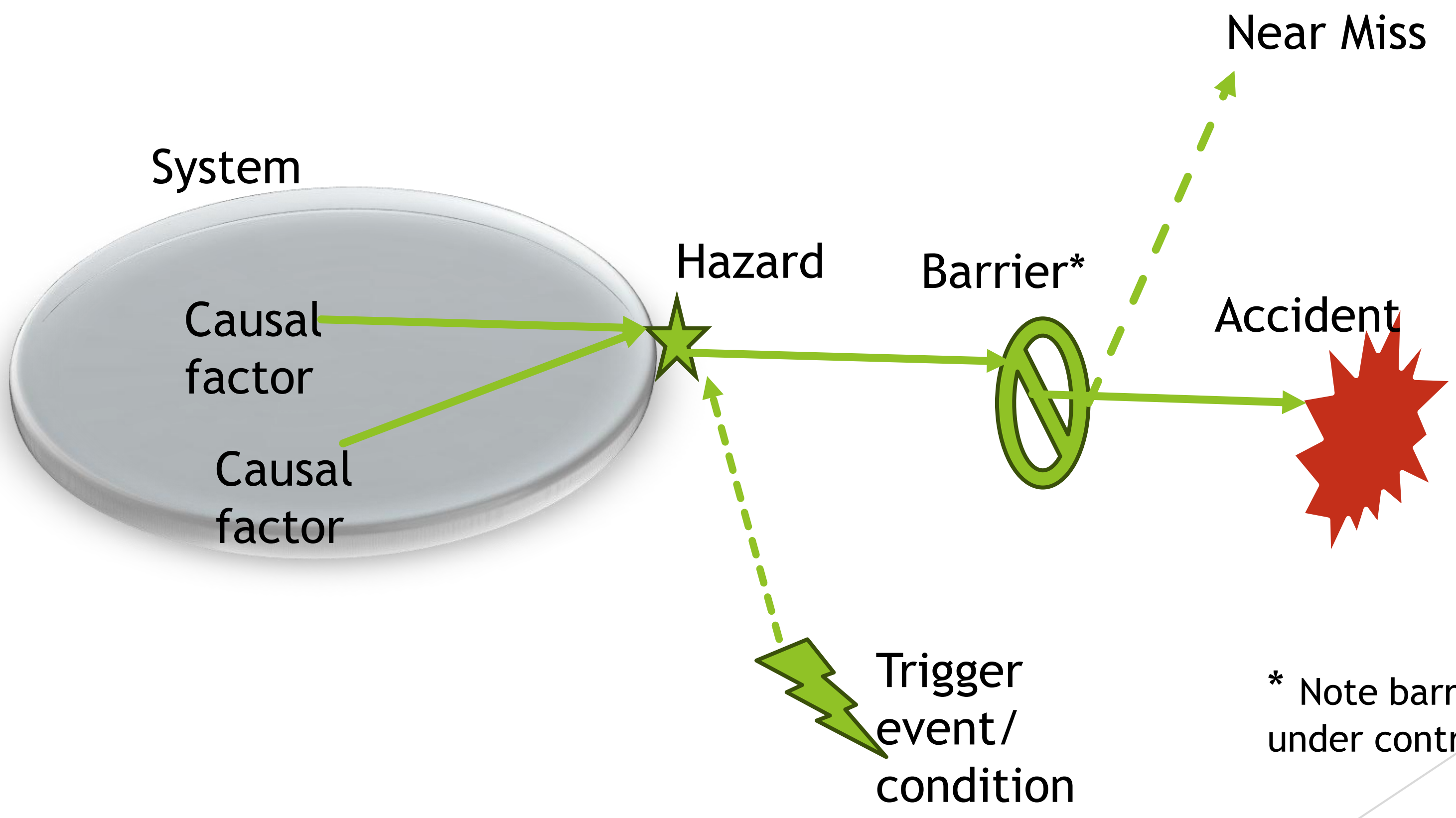
C6 Certain provisions of this Act applied (with modifications) (E.W.S.) (1.1.1993) by S.I. 1992/3060, **reg. 15(1)**

C7 Certain provisions of this Act applied (E.W.S.) (1.2.1993) by S.I. 1992/3217, **reg. 21(1)**

C8 Act excluded (E.W.) (1.4.2006) by S.I. 2005/1541, **art. 47**, (with art. 49)

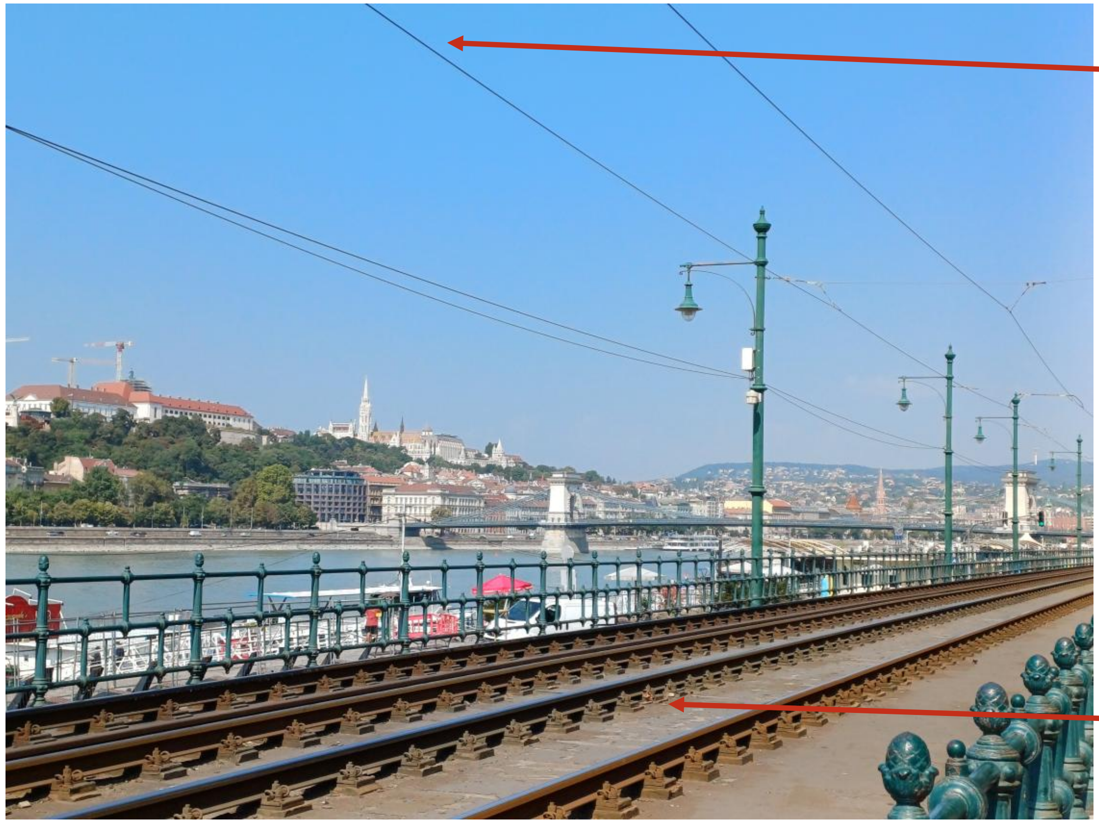
Definitions

Risk is a combination of
likelihood and consequence



* Note barriers are not under control of system

Example



Hazard

Looks benign but looks can be deceptive

Obvious Hazard

Visualisation of Risk

	Insignificant	Marginal	Major	Critical	Catastrophic	
Frequent	Tolerable	Intolerable	Intolerable	Intolerable	Intolerable	<1yr
Probable	Tolerable	Tolerable	Intolerable	Intolerable	Intolerable	<2yrs
Occasional	Tolerable	Tolerable	Tolerable	Tolerable	Intolerable	<5yrs
Rare	Negligible	Negligible	Tolerable	Tolerable	Tolerable	<10yrs
Improbable	Negligible	Negligible	Negligible	Tolerable	Tolerable	<20yrs
Highly Improbable	Negligible	Negligible	Negligible	Tolerable	Tolerable	≥20yrs

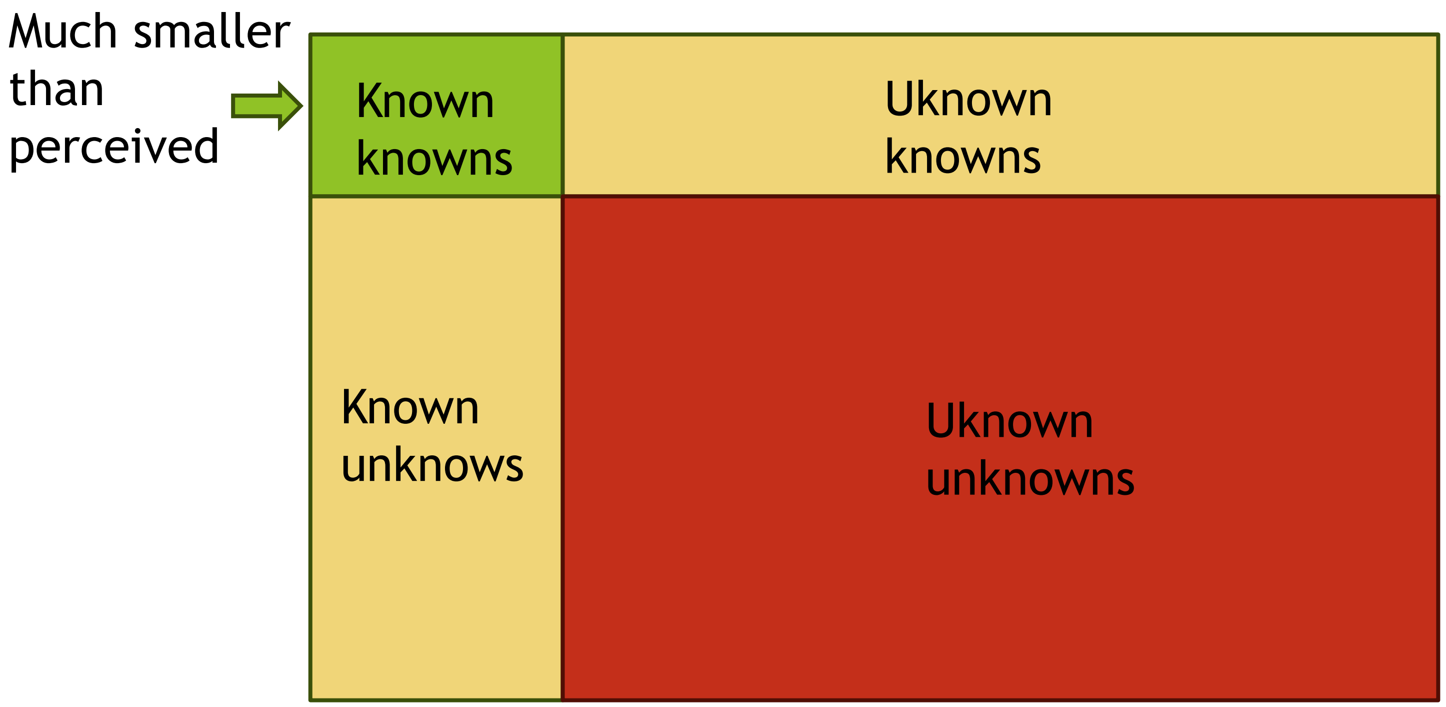
Formulated from EN50126 Cenelec 2017

Risk, Complexity and Understanding

			Risk	
			Complexity →	
			Known	Unknown
Complexity ↓	Understanding	Known	Specification	General industry knowledge
		Unknown	Design/operation outcomes	Unknown unknowns

Adapted from Bowman 1990 & Barnatt 2021

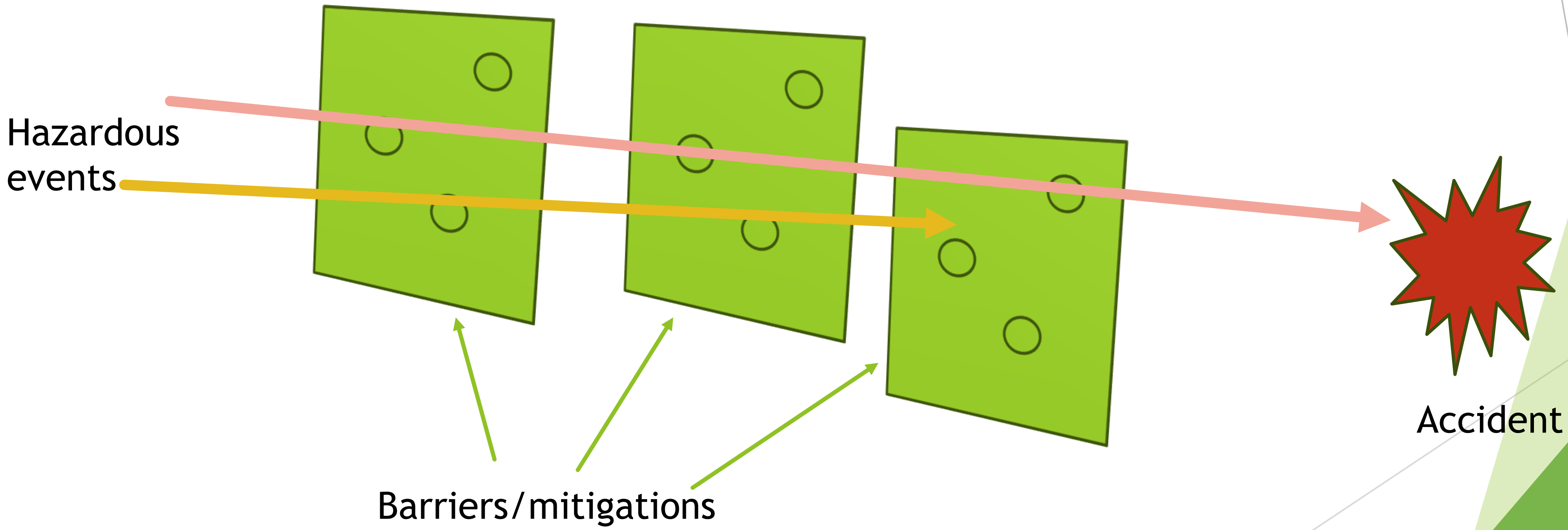
Implication for software



- A customer requirement is likely to be incomplete and fuzzy
- A specification is only an interpreted codification of what is known
- Therefore, there are likely to be holes in design documentation, coding and testing

Swiss Cheese – defence in depth

Franz Golhen, Public domain, via Wikimedia Commons



Based on Reason 1997

Software

Positive defences

- Defensive programming
- Check inputs
- Check outputs
- Task monitoring
- watchdogs
- Simplicity

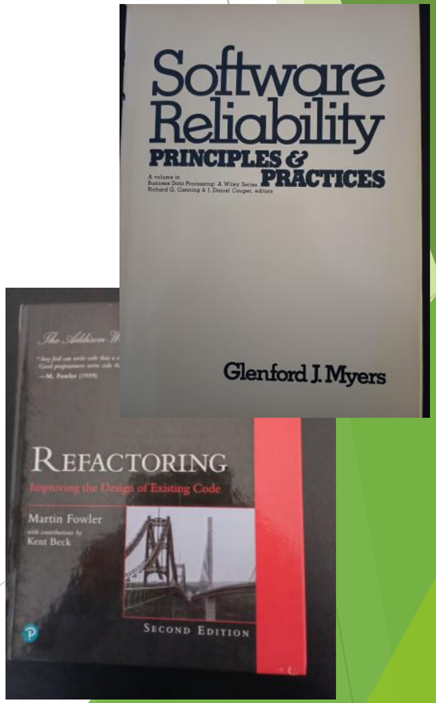
Mutual suspicion

- Keep methods simple
- Adhere to the solid principles
- Loose coupling Data coupling 6 types
- High strength Functional strength 7 types
- Log events
- Testing

Negative drivers

- Required speed
- Complex product requirements
- AI
- Non-deterministic

Myers 1976



Complex systems

- Difficult to understand
- Have emergent behaviours as a result of interactions
- Not possible to test the parts and pronounce on the system safety

- Systems become more complex when intercommunication taken into account

- Software has a stack of dependencies – High level language does not necessarily mean better
  - Hardware – micro code
  - Operating system
  - Tools for creation and maintenance, language nuances
  - Compiler/interpreter correctness
  - User compliance
  - Interfaces
  - libraries

Soup

- Software of uncertain pedigree
- No special process beyond normal commercial requirements
- Windows falls into this category
- The positive is that lots of people use it so errors are more likely to be discovered quickly – lowering the probability density of errors
- It’s cheaper than creating your own
- Convenient

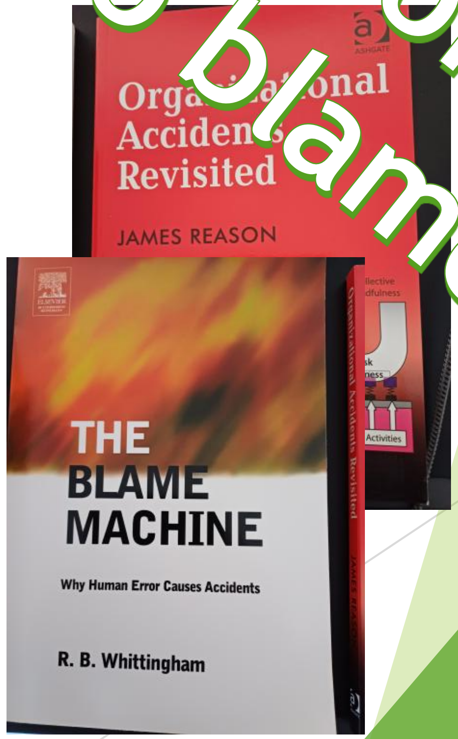
Designed by Freepik Freepik.com

- Defences:
  - Use wrappers to limit connections between SOUP and the application
  - Restrict unused functionality and the influence on other parts of the application
  - Avoid newest versions of the SOUP

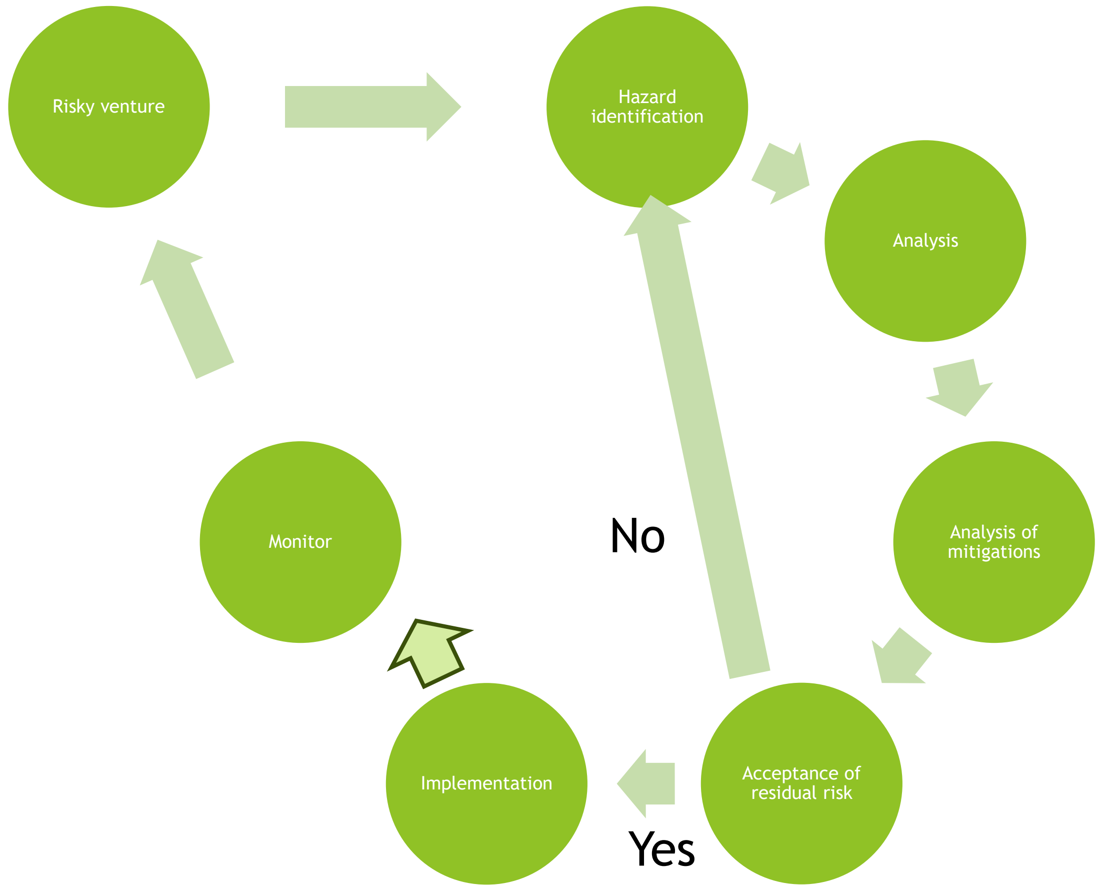
Sociotechnical environment

Users are often NOT to blame

- A System/software is a collaborative endeavour of groups of people to produce an outcome.
- Roles range from law makers to designers and coders and users
- Users are the ultimate recipient of the products.
- Users are human
  - Make mistakes
  - Limited in capacity
  - Lazy – try and find the easiest way to do something
  - Flexible
  - Subject to influence
  - Break rules
  - Easy to blame
- The best a human (when checked) can manage is an error rate of 10^{-5}/hr


Organizational Accidents Revisited
JAMES REASON
THE BLAME MACHINE
Why Human Error Causes Accidents
R. B. Whittingham

Conceptual hazard analysis process cycle



Risk Assessment Objective

E - Eliminate
R - Reduce
I - Isolate
C- Controls (processes)
P - PPE
D - Discipline

Risk Assessment ->Opinion Engineering

- The hazards are often those with a very small likelihood of happening
  - Therefore, little or no statistical data
- The hazards are having to be guessed as the system did not exist before…no experience
- The hazards are latent and unknown

- This leads to ‘experts’ being asked what they think.
- There are formal identification methods
  - HAZOP, SHARD (an examination of data flows)
  - What if – sometimes called SWIFT
  - Day-in-the-life-of

The trouble with complex systems & understanding

- Humans are not capable of deciphering complexity
  - 3 types: Manson 2001

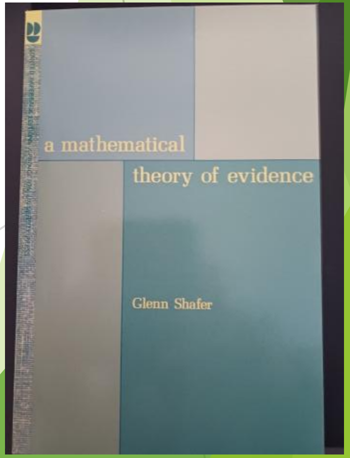
Algorithmic	difficult to create equations to describe
Deterministic	settle on a universal answer; where small changes lead to different outcome
Aggregate	understanding overall effect where behaviours emerge and coupling

Opinions

- Humans are rubbish at estimating probabilities
- Instead answer a substitute question based on feelings, associations and heuristics Kahneman 2011
- A further concept to remember is what is known at the time is all the information to base any judgement on Kahneman 2011

- This leads to the concepts of likelihood, belief and plausibility as the basis for forming relationships of cause and effect in a risk assessment. Shafer 1976
- Therefore the 'expert' may well base the assessment on the strength of belief given a scenario is plausible. … In other words: OPINION!
  - Value of opinion base on credibility, experience and background knowledge of person

- Therefore, completeness and accuracy are a problem!



Hazop - issues

- A real concern is completeness
- A Hazop is designed to prompt participants to suggest hazards and causes through a series of keywords.
- A good session will get many suggestions.

- However, there is a problem with probability analysis

  Edwards 1992 & Pawitan 2001
  - Mathematically probability is required to have 3 properties
    - Probabilities are to be between 0 and 1
    - The underlying model for each probability estimate is constant
    - All probabilities represent a unique event

Hazop - consequence

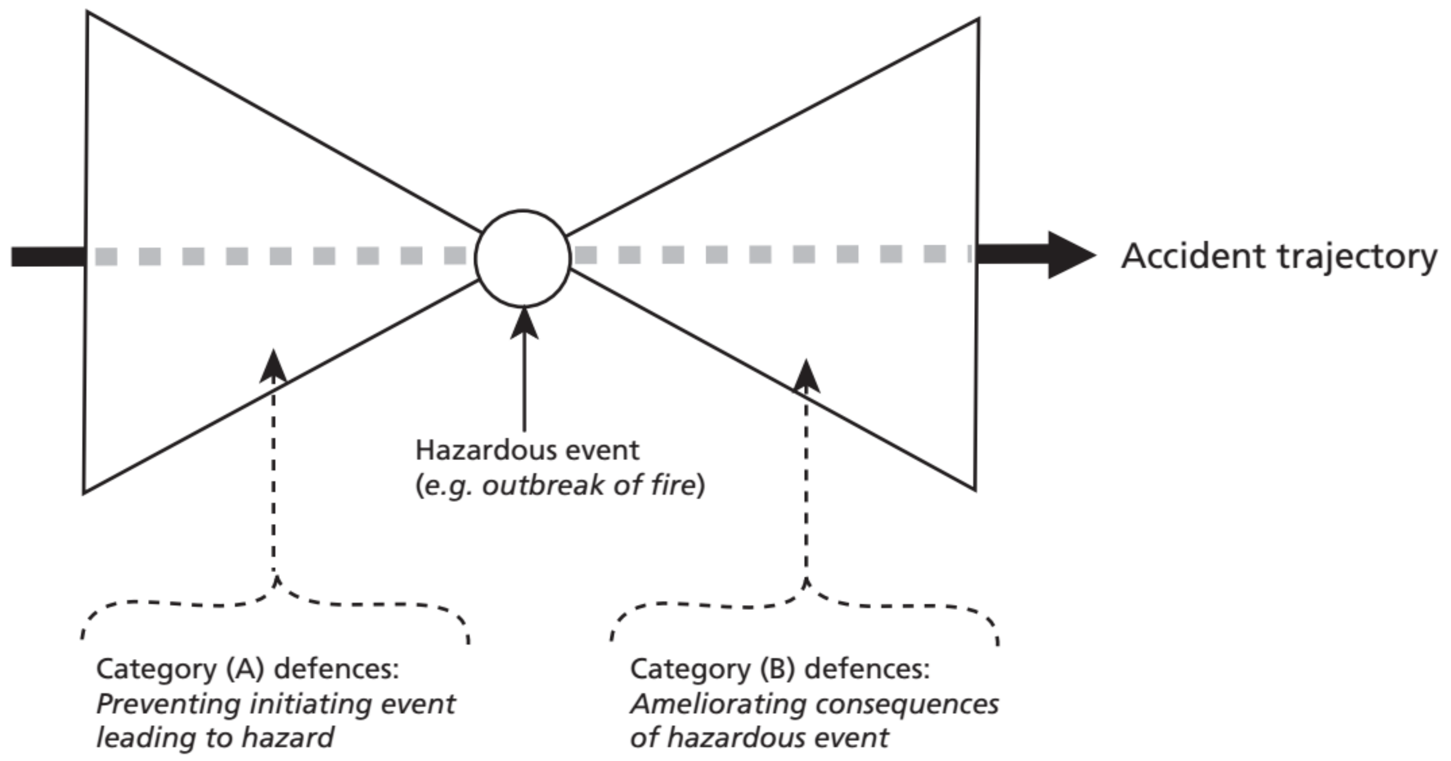
- Because Hazops are opinion based the events are highly likely to be polluted
  - Ie 2 hazards partially describing the same thing in different words
- Consequently, for probability-based analyses (e.g. FTA) the total probability of the top event can exceed 1
  - Can lead to the whole analysis being called into question
- However, it’s better to have this fault to deal with than miss a hazard which later turns out to be critical
- In essence it is more appropriate to refer to likelihood as a more flexible and suitable candidate as a basis of assessment.

Many analysis methods

FMEA & FMECA	- Failure Modes and Effects Analysis (criticality)
STAMP	- System Theoretic Accident Model and Processes
FRAM	- Functional Resonance Analysis Method
CAM	- Composite Analysis Method
Bow-tie	
FTA	- Fault Tree Analysis
ETA	- Event Tree Analysis
FFA	-Functional Failure Analysis
Bayesian networks	– computationally complex with joint probability tables

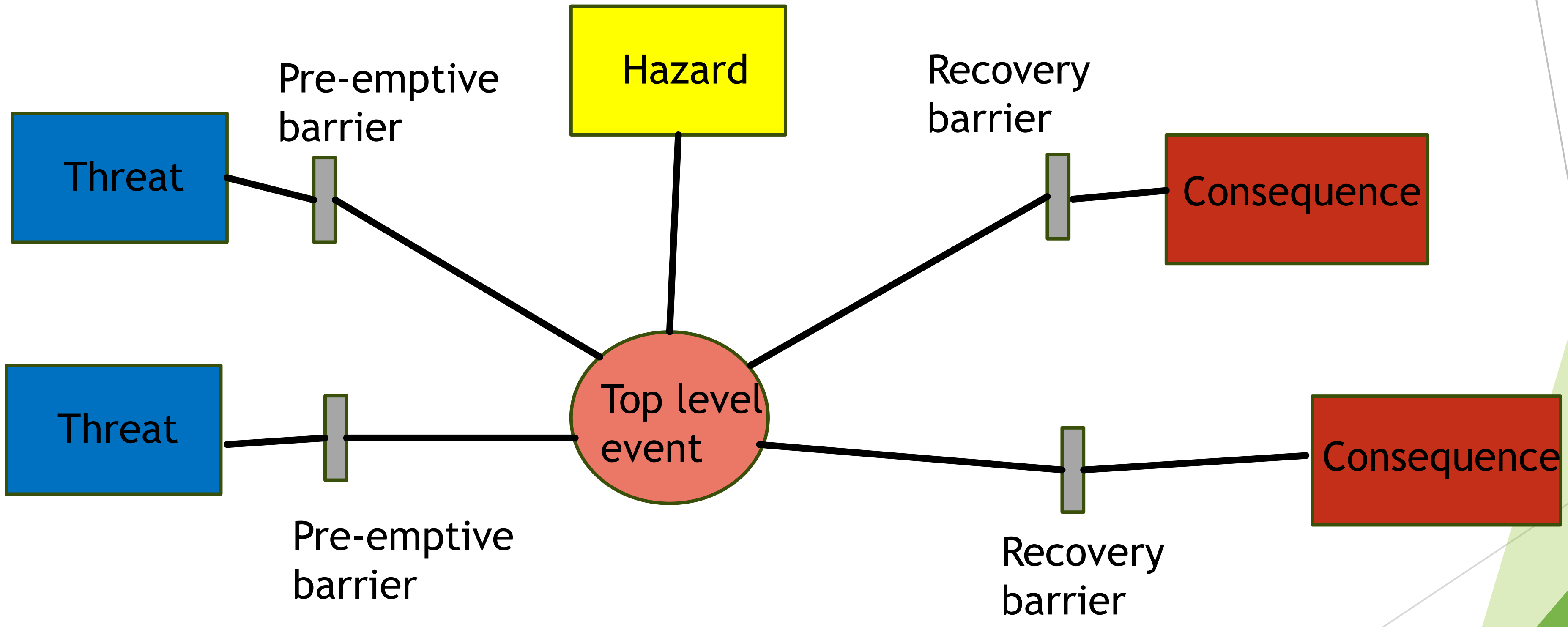
Lots to choose from each has strengths and weaknesses

Bow-tie


Accident trajectory
Hazardous event
(e.g. *outbreak of fire*)
Category (A) defences:
Preventing initiating event leading to hazard
Category (B) defences:
Ameliorating consequences of hazardous event

Hadden-Cave QC 2009

Bow-tie details



Threat= cause of top level event

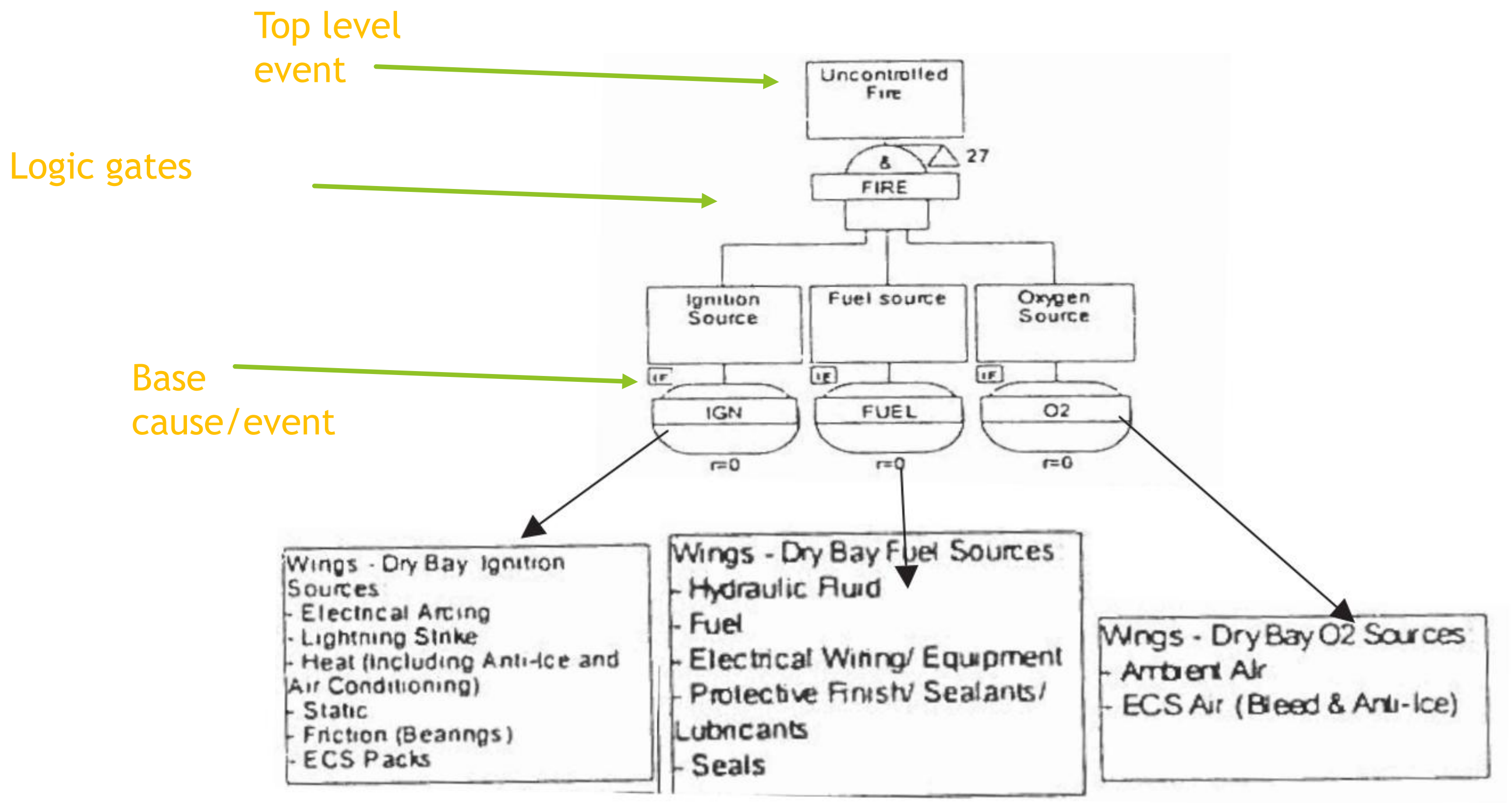
Pre-emptive barrier= interrupt threat

Top level event = Loss of control

Hazard = state

recovery barrier= interrupt event

Fault-tree analysis FTA

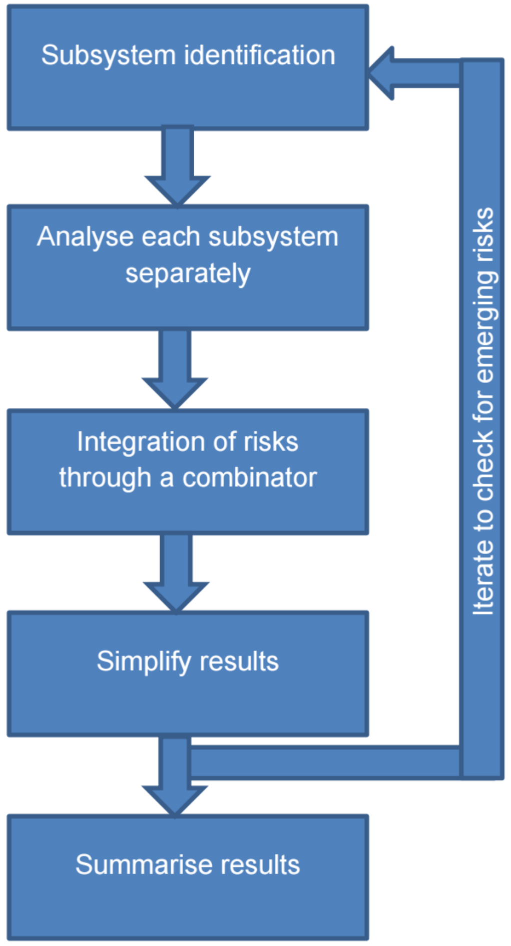


Hadden-Cave QC 2009

Beware of being too precise as there is normally uncertainty in the base events

Analysis method - CAM

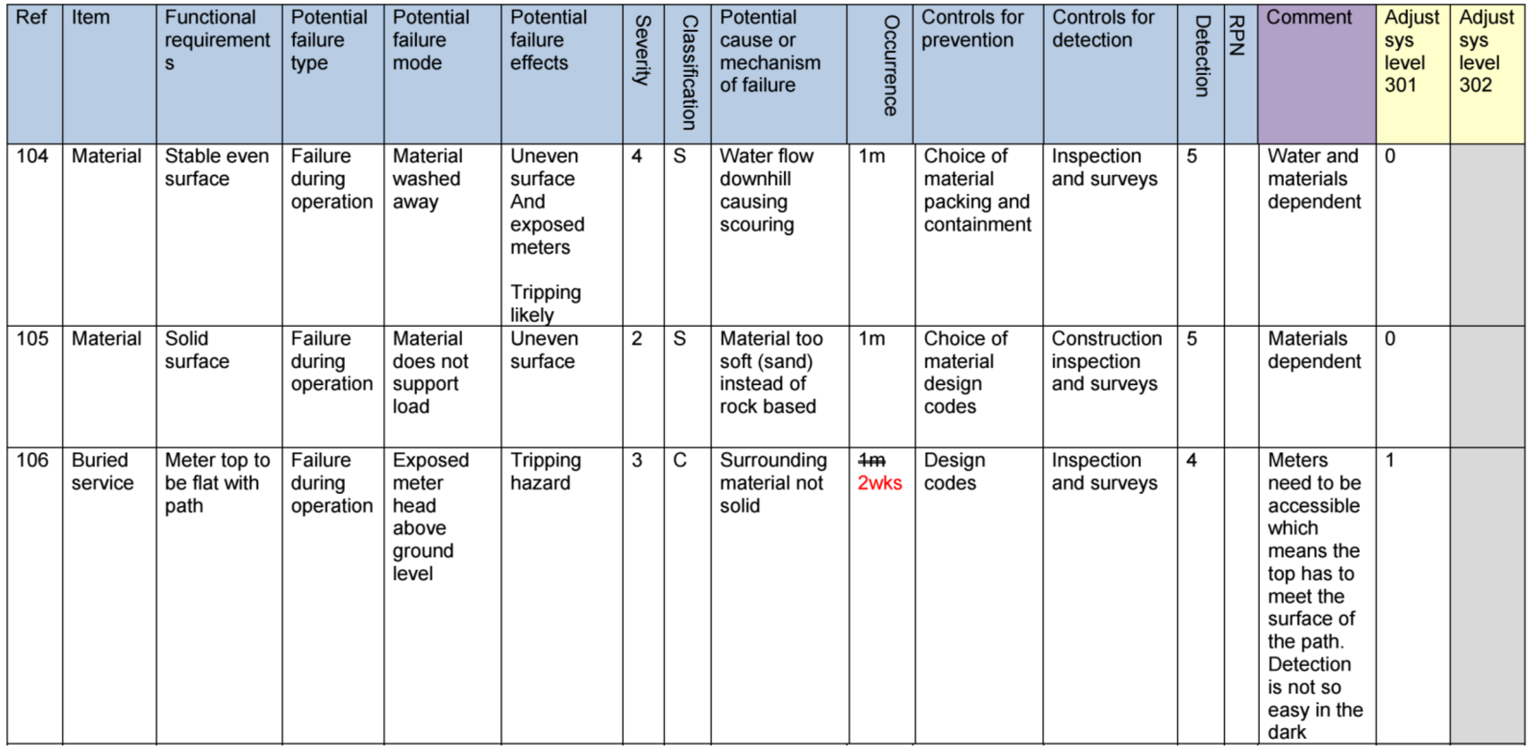
- Composite Assessment Method (CAM)
- Designed to analyse complex systems
- Allows the use of existing techniques and combines the results to an understandable output



Subsystem identification
Analyse each subsystem separately
Integration of risks through a combinator
Simplify results
Summarise results
Iterate to check for emerging risks

Barnatt 2021

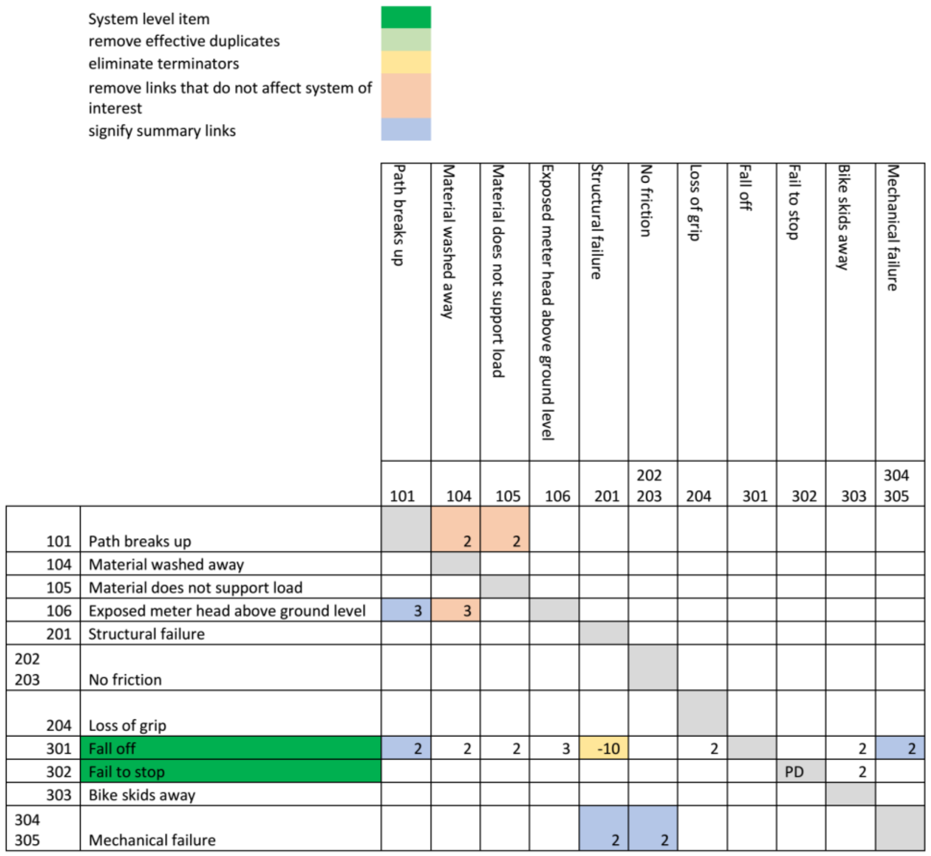
FMEA - example



Ref	Item	Functional requirements	Potential failure type	Potential failure mode	Potential failure effects	Severity	Classification	Potential cause or mechanism of failure	Occurrence	Controls for prevention	Controls for detection	Detection	RPN	Comment	Adjust sys level 301	Adjust sys level 302
104	Material	Stable even surface	Failure during operation	Material washed away	Uneven surface And exposed meters Tripping likely	4	S	Water flow downhill causing scouring	1m	Choice of material packing and containment	Inspection and surveys	5		Water and materials dependent	0	
105	Material	Solid surface	Failure during operation	Material does not support load	Uneven surface	2	S	Material too soft (sand) instead of rock based	1m	Choice of material design codes	Construction inspection and surveys	5		Materials dependent	0	
106	Buried service	Meter top to be flat with path	Failure during operation	Exposed meter head above ground level	Tripping hazard	3	C	Surrounding material not solid	~~1m~~ 2wks	Design codes	Inspection and surveys	4		Meters need to be accessible which means the top has to meet the surface of the path. Detection is not so easy in the dark	1	

Barnatt 2021

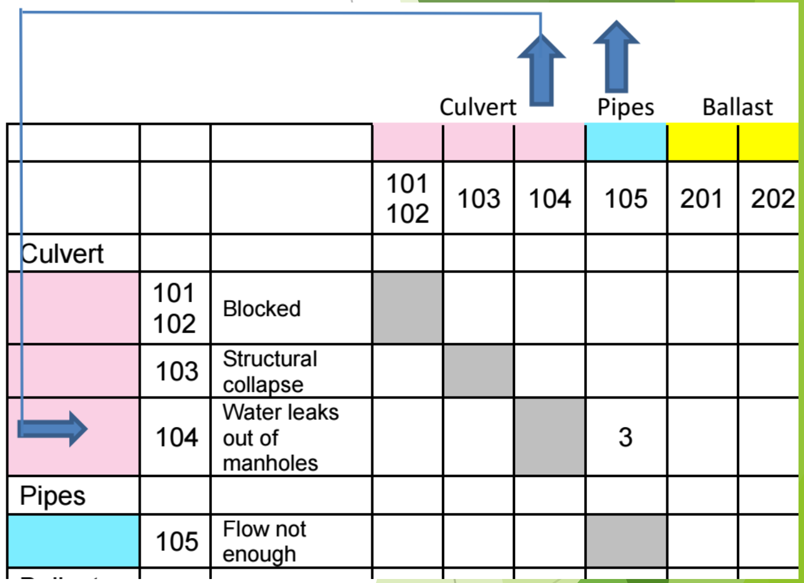
Rationalised – CAM-C example



System level item	🟩
remove effective duplicates	🟩
eliminate terminators	🟨
remove links that do not affect system of interest	🟧
signify summary links	🟦

		Path breaks up	Material washed away	Material does not support load	Exposed meter head above ground level	Structural failure	No friction	Loss of grip	Fall off	Fail to stop	Bike skids away	Mechanical failure
		101	104	105	106	201	202 203	204	301	302	303	304 305
101	Path breaks up		2	2								
104	Material washed away											
105	Material does not support load											
106	Exposed meter head above ground level	3	3									
201	Structural failure											
202 203	No friction											
204	Loss of grip											
301	Fall off	2	2	2	3	-10		2			2	2
302	Fail to stop									PD	2	
303	Bike skids away											
304 305	Mechanical failure					2	2					

Barnatt 2021



			Culvert			Pipes	Ballast	
			101 102	103	104	105	201	202
Culvert								
	101 102	Blocked						
	103	Structural collapse						
	104	Water leaks out of manholes				3		
Pipes								
	105	Flow not enough						

Barnatt 2021

Cause-consequence analysis - example

Ref	Title	Hazard	Cause	Description	Consequence scenario	Consequence description	Control	Evaluation type	Likelihood	Consequence	Risk
301-104 (R1)	Fall off	Unbalanced	Path material washed away	Heavy rain washes material away and leaves uneven surface	Rider loses balance and falls off	grazed leg	Path maintenance	Risk Estimation	Occasional	Marginal	Tolerable
301-105 (R2)	Fall off	Unbalanced	Path material not suitable	Path rutted	Rider loses balance and falls off	grazed leg	Path design	Risk Estimation	Occasional	Marginal	Tolerable
301-106 (R3)	Fall off	Unbalanced	Exposed meter head	Meter head is above path and is hit by bike wheel	Rider loses balance and falls off	Broken leg	Path design and maintenance	Risk Estimation	Probable	Major	Intolerable
301-204 (R4)	Fall off	Unbalanced	Loss of grip on tyres	Tyres lose grip on the surface	Rider loses balance and falls off	grazed leg	Experience	Risk Estimation	Rare	Marginal	Negligible

Risk matrix - example

	Insignificant	Marginal	Major	Critical	Catastrophic
Frequent	🟨	🟥	🟥	🟥	🟥
Probable	🟨	🟨	🟥 R3	🟥	🟥
Occasional	🟨	🟨 R1, R2, R5, R7	🟨	🟨	🟥
Rare	🟩	🟩 R4	🟨	🟨	🟨
Improbable	🟩	🟩	🟩	🟨	🟨
Highly Improbable	🟩	🟩	🟩 R6	🟨	🟨

Barnatt 2021

Final remarks

Checklists are the enemy of understanding

- Software almost certainly has errors
- The effect on the outside world is what counts
- Defence in depth is essential to prevent software errors doing nasty things
- Humans are often pointed to as being at fault, but are they? They could be heroes!
- Assessment is difficult and often comes down to opinions
- There are many risk assessment methods each with strengths and weaknesses

- Understanding the hazards and the resulting risks is critical

Else there is no value in risk assessments

Questions

Pixaby.com