Risk analysis involving human factors

Kim Rostgaard Christensen

Kongens Lyngby 2012
IMM-BSc-2012-12

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk IMM-BSc-2012-12

Summary (English)

This thesis seeks to identify and determine the significance of human factors in complex safety-critical systems. It is an area with a lot of uncertainty as humans are themselves complex systems.

Most established methods for modelling accidents, have proved themselves too crude for use in resilience engineering, and more sophisticated methods are starting to show. The relative new method FRAM is looked upon more closely in the thesis.

This purpose of this thesis is to document, discuss and apply a modern method for capturing human factors in a safety system - using a real-world accident as an example.

The supervisor on this thesis is Paul Pop (paul.pop@imm.dtu.dk)

Summary (Danish)

Denne afhandling søger at identificere og afgrænse betydningen for menneskelige faktorer i komplekse sikkerheds-kritiske systemer. Dette er en område præget af megen usikkerhed, da mennesker selv er komplekse systemer.

De fleste etablerede uheldsmodelleringsmetoder har vist sig for primitive, til at bidrage til forbedringen og skabelsen af mere modstanddygtige systemer og mere sofistikerede metoder begynder at vinde frem. I denne afhandling, vil udgangspunktet være den relativt nye metode FRAM, der forsøger at tage højde for nogle af svaghederne i de nuværende metoder.

Formålet med denne afhandling er at dokumentere, diskutere og anvende en konkret metode til at afgrænse de menneskelige faktorer i et sikkerhedssystem - gennem et konkret eksempel.

Vejlederen på denne afhandling er Paul Pop (paul.pop@imm.dtu.dk)

Preface

This thesis was prepared at the department of Informatics and Mathematical Modelling at the Technical University of Denmark in fulfilment of the requirements for acquiring a BSc. in Informatics.

It deals with the aspect of human factors in accident models and consists of an introduction to different accident model as they have evolved throughout time - from the early linear models to the modern non-linear complex models.
For the latter, a simple accident will be modelled to help illustrate the usage, and strengthen the discussion of its validity and application.

As the accident being modelled is railway associated, some terminology about railway safety is provided, in the scope it was found necessary.

Lyngby, 01-April-2012

Kim Rostgaard Christensen

Acknowledgements

First, I would like to thank Erik Hollnagel, among others, for all the very useful literature he has published.

I would also like to thank my supervisor at Atkins Denmark; Per Stolze which is, quite frankly, a human encyclopedia on train and railway safety - and has been used as such during the writing of this thesis.

Last, but certainly not least, I would like to thank my lovely girlfriend - who has supported me during the writing of this thesis, and throughout all my over-ambitious projects.

Contents

Chapter 1

Introduction

Accidents have been a part of our world for as long as we have been around, and as technology has progressed and given us the potential to build better and more powerful systems, so has the scale and impact of the accidents. From early day accidents with fire to modern day nuclear power plant incidents.

Risk is a part of our everyday life, as we use trains, aeroplanes or even cross the road. Applying new technology usually involves venturing into uncharted waters - so to speak. And a lot of new lessons are learned; the hard way.

For the past century, process and productivity has gone from simple linear assembly-line, single purpose work to complex changing tasks. This is also highly reflected in the scope/paradigm of the models used to describe the accidents through this century, and a brief overview on the evolution of these can be found in chapter 3.

Systems have also evolved from simple linear controlled systems to complex intercoupled systems. A lot of organizational structure have been built around them, and as safety requirements has gotten more strict the complexity increases (see section ??).

In retrospect, most technological progress has had a few common straightforward goal; to simplify and eliminate the tedious and repetitive tasks, increase

productivity, and ensure the safety of the people using it.
An example could be some factory with an assembly line, that has been automated. This automation now becomes a representation of the previous manual process, that has to be managed from higher level of abstraction, and at some point, a human is controlling the process - or more accurate - a model of the process.

This abstraction and increasing complexity is posing a problem for the human operators of the systems. The details are almost infinite, so only a portion of is can be presented to operators before they experience “information overload”. This puts heavy constraints on the requirements towards the user interface presented to the operators.

But still, this assumes that human performance is constant, which is rarely the case. A lot of factors affect the performance of human operators (see section 2.1), and they have become the most error-prone component in complex socio-technical systems today.

This thesis will look into the challenges faced when the human factor has to be assessed in a safety-critical system. It will present some of the currently used accident/system modelling methods, and a relatively new method that tries to take into account, the human performance variability issue. The latter method is presented along with an example of its usage.

As this thesis extends an internship at risk department dealing with (mostly) railway safety, and the accident modelled in the thesis is a railway incident, there is a few introductory chapters containing a brief introduction to relevant railway concepts and terminology.

1.1 Terminology

This section serves to establish a common terminology.

1.1.1 Accident

Accidents can be defined as unplanned and undesired release of energy that result in losses (either financial, human or ecological). Accidents happen for a number of reasons, and are usually caused by a combination of several unfortunate events rather than a single one. Safety planning during design remedies this, but sometimes the unanticipated happens. This is referred to as Beyond Design-Base Accidents, and are accidents that occurs as a reaction to unanticipated usage or capacity load. In other words, it's what was no taken into account when the system was designed.

1.1.2 Near miss

A near miss is often described as "an unplanned event that did not result in injury, illness, or damage - but had the potential to do so". This is also referred to as "Close call" or "Near Collision" when moving objects are involved.

Near misses are not, as a rule, taken into consideration as an accident. Whether or not the event should examined and treated as an accident depends largely on how close it was to evolve into a real accident.

A recent incident with the Danish IC4 trains resulted in a near miss situation, due to a brake failure. An accident investigation was committed, largely due to the inexplicably of the failure - and the relative young age of the trains. [Hol04] extrapolates the 1:10:30:600 figures giving 1 accident for every 300 near-miss.

A problem with near-misses is they are rarely reported, if not noted by a controlling instance.

1.1.3 Artefact

An artefact is a human made object, or an object with human-applied usage. Artefacts play an important role in everyday life. Most artefacts are created for a specific function or to solve a specific problem, but some are also bi-products of a modernization process.
An example of this, could be the digitalization process of a production plant; a general-purpose computer becomes a special-purpose artefact replacing some manual processes or activities. The limitations of the general-purpose computer still apply though, and new interaction methods will now be constrained by the limitations of the technology, rather than the humans.

1.1.4 Resonance

Resonance is a phenomenon in physics making a system oscillate at a higher amplitude when a force is applied. In physics, this is often depicted as a pendulum in motion, where the applied force is the initial push.

[Hol04] uses the example of a swing set found on playgrounds. When these are set into motion, one can apply force at just the right time, to increase the amplitude of the oscillating function, that represents the swing. Similarly, you can decrease the amplitude by applying force a bit earlier - hereby damping the kinetic energy of the swing.

Resonance can be used to model how large changes in variability can affect and propagate through an entire system.

1.1.5 Railway terminology

Safety has always been a high priority in railway engineering and deployment, and it has thus been a largely contributing industry to safety critical research. Some relevant terminology is covered in this section.

1.1.5.1 Block

A block is a distance of railway that, at any point in time can only be occupied by one train. There are two types of blocks; fixed and moving.

Traditionally, railways are divided into a number of fixed blocks with entry and exit signals. These signals will represent train movement along the block based on a predefined policy. The policy is then again determined from a number of parameters:

- The permitted maximum speed on the line
- The maximum speed and braking characteristics of the different trains occupying the track
- Geological conditions, such as gradients, as these could lead to increase in breaking time.
- Line-of-sight. Being that the signal is optical, the train driver must be able to see it before acting on it.

- The reaction time of the driver

Whereas the maximum speed and geological conditions can be modelled linear - the response time of the driver cannot. And on a line without ATC (see section 1.1.5.3) failure to observe a non-go signal will effectively cancel out all other factors in the model.

Fixed blocks wastes a lot of capacity, as most blocks go largely unused for most of their distance, plus there is a lot of overhead on stopping times.

Moving block address this issue. Instead of having the line divided into a number of fixed blocks, a "safe distance" is defined dynamically based on the current speed and location of the train. This greatly increases the requirement for the technological infrastructure, and the dependability of it.

1.1.5.2 Interlock

An interlock, in railway terminology, is a mechanism that prevents more than one train to be in a given block at a time. A more general term is found in [Lev95].

> Interlocks are commonly used to enforce correct sequencing or to isolate two events in time.

In railway, sequencing is also applied. Usually the sequence "occupied","occupied","not occupied" for two blocks must be asserted to release the block not occupied.

1.1.5.3 ATC

ATC, or Automatic Train Control is a mechanism that allows automatic breaking of trains as the pass a signal at danger. It will signal the train driver that a signal has been passed, and automatically brake the train based on a calculated braking curve.

1.1.5.4 Level crossing

A level crossing, railway/railroad crossing is an intersection between road, designed for regular traffic, and railway tracks. In modern track planning, they are usually avoided as they are a source of both delays and hazards. Instead, bridges are built.

1.1.5.5 Timetable

The primary barrier to provide safety in railway operation is the timetable. It specifies which trains are supposed to be at a certain location at a certain point in time. It is considered the first safety measure in railway operation - and thus required all employees to be in possession of a pocket or wrist watch in order for them to be hired.

1.2 Safety engineering

Safety engineering is a discipline that seeks to design systems that are safe for usage. This is basically, avoiding accidents.

Malicious acts, such as sabotage or terrorism, are considered outside the scope of safety engineering - but is instead treated by a separate field called security engineering. Natural disasters, such as earthquakes and tsunamis, are also typically left out. These are commonly referred to as "Acts of God" as a unified description.

1.3 Resilience Engineering

Resilience engineering can be considered the complimentary to safety engineering; where as safety engineering seeks to build a better and safer system, resilience engineering embraces the fact that errors arise within a system - and tries to limit the impact of these.

Whereas traditional risk management rely largely on lessons learned, and empiric data to provide probabilities; resilience engineering provocatively seeks to create safety though flexibility. Meaning that when a sub-system breaks down it does not necessarily mean the breakdown of the entire system.

Basically it cuts down to the the following question: "If this component breaks down - how will the rest of the system react, and how can I limit the impact." HAZOP (3.5) and FMEA (3.4) are methods that support resilience engineering.

1.3.1 Barriers

Depending on the view, domain or application there may be more than one way of categorizing barriers.[Hol04] defines a barrier as:

> Barriers are hindrances that may either prevent an unwanted event from taking place, or protect against the consequences

- and also defines four categories of barriers; Physical, functional, symbolic and incorporeal. An example is in parentheses.

- Physical barrier: Either prevents an action being carried out, or allowing it to propagate (A wall)
- Functional: Makes an action impossible via preconditions and interlocks. May protect against consequences when activated. (interlock)
- Symbolic: Interpretational barrier (signs, signals alarms)
- Incorporeal: Rely on knowledge and information (rules, restrictions, laws)

As organizational structures and legislation is putting more and more constraints on the security requirements for a system, it has to be included in the modelling from an early stage.

1.4 Accident modelling

Accident modelling is a very useful tool, especially when an accident is very complex. It also introduces a formalism to accident reports.
Accidents models have gone through a number of paradigm shifts briefly discussed here:

1.4.1 Domino model

Accidents are, in classic safety literature, depicted as a series of sequential events - each one is the causing factor of the next. Stopping the event chain from propagating before it ultimately leads to an injury (or damage), will prevent it. This

model is also known as the domino model, due to its close resemblance to a series of dominoes and the way they fall sequential. And safety engineering has been focused on identifying the one event that started it, or putting up a barrier that prevented the chain to complete.
This reasoning is very easy to follow and makes a lot of sense in simple systems. It is easily depicted and hereby easily communicated. There is also a tendency that people think in linear and sequential systems, rather than in complex intercouplings - as these are far easier to comprehend.

But, in general it is not recommended to say that a specific event (X) causes another (Y). This implies that X is a precondition to Y, and by eliminating X, Y will no longer happen[Skl02].

1.4.2 Swiss Cheese Model

The Swiss Cheese Model depicts barriers as layers of Swiss cheese with holes in them. When barrier holes align, an hazard is able to "pass though" the holes and manifest itself into an accident.

1.4.3 Complex non-linear models

There has been a paradigm shift in the view on accidents in the later years. Now, accidents are lo longer considered a linear succession of events, but rather a complex combination of events.

Current formal requirements on safety measures in systems engineering, usually focus on the robustness and integrity of single components, rather than on the coupling of these and the system as a whole. But as single technical and organizational components become more robust and resilient, they also tend to get more complex - increasing the requirements to the humans in the system.

Chapter 2

Human factors

The variability of is system is becoming more and more dependent on individual and/or collective performance of humans, and the need for a model that takes these into account, has arisen.

As previously discussed, linear and strongly intercoupled systems are no longer the reality in which we live in. Technological advances has created a self-reinforcing closed loop circuit in which complexity continues to nourish itself (2.2).

To be able to represent and integrate humans as a part of complex larger system, it is necessary to identify and ultimately accept the behaviours and limits of them.

Until recently, humans have been regarded and modelled as machines - and usually as a primitive feedback loop. But studies and empiric data shows that this model is not optimal, as humans acts as feed-forward systems (2.1.2).

During the 1930s and 1940s, behaviourism reduced humans to black box systems and observed response to stimuli, much like how micro-organisms are studied. The problem with this is that human response is largely dependant on current performance, and more importantly, context and environment.

2.1 Human performance

2.1.1 ETTO principle

ETTO is short for Effectiveness-Thoroughness-Trade-Off, and the ETTO principle formalizes the balancing issue in having conflicting requirements or goals. The stop-rule (3.2) is an example of the ETTO principle, as it is usually not possible to do a more in-depth investigation than what time or financial resources allows.

ETTO is something most people do every day without giving it much thought. Cooking for instance, may be subject to a time constraint (dinner time), and thus flavouring the food may become under-prioritized in order to meet the deadline. The saying, I've heard in software engineering circles;

> The product will be Fast, Cheap or Good - pick any two

Seems appropriate here.

ETTO is also commonly applied when insufficient information or knowledge is available; sometimes people fail to acknowledge or follow rules simply because they do not comprehend or know the purpose. This is common in layered management systems, where higher layers of management may enforce procedure rules on workers, in order to control, monitor or optimize processes.

2.1.2 Feed-forward

Human behaviour can be regarded as a feed-forward loop, rather than a feed-back loop. In a feed-back loop, you basically respond to the changes and/or information presented to you at the time of their arrival. In feed-forward systems, however, there is a expectation on what will happen next and responses to action are based on the expected effect among a set of responses.

A good example is driving a car. Minor corrections are done to the direction of the car, sometimes at a very high frequency of several times a second. This has become second nature to regular drivers, but anyone who has observed another person driving a car would have noticed these minor corrections.
The changes done in direction and/or speed of the car are done on the basis on the expectations the driver has to them. In other words, he/she dynamically

alters the total system (car+driver) to reflect the desired outcome of the driver - reaching the destination safely.

2.2 The complexity paradox

As technology advances, systems become more complex, and as their complexity increased, so does their ability to fail in unpredictable ways. This is mainly caused by the incomprehensible size of the entire system as a whole and the number of components used and their inter-dependency. A singe component can fail and propagate through the system undetected and be a contributing factor along with others (e.g. environmental or organizational) to the failure of the entire system.

As these failure are detected, remedial actions are taken - leading to increasing complexity of the system - effectively amplifying the unpredictability.

Computer software is a very good example of a complex system that respond poorly to remedial actions, as resilience is not normally a design goal. A large number of assumptions about values and parameters in a software system can lead to very unpredictable behaviour.

Assume the following: a number in a computer system is represented binary form with a fixed size, e.g. 8 places (bits). When representing a negative number the leftmost bit is '1' - or 'logic high' which leaves only the remaining 7 bits to represent the actual number. The largest signed number we can represent with 8 bits is $2^8 - 1 = 127$. Due to the nature of computer hardware, the number will "wrap around" - much like in a trip meter in an automobile. The problem with the signed representation is at that for an 8 bit representation, the following holds $(2^8 - 1) + 1 = -128$ which is very inaccurate if you expect a positive value.

This is a trivial error, but unfortunately still very common - and will most certainly result in unpredictable behaviour in most systems if not detected.

2.2.1 Cognition

Cognition is popularly referred to as the processes of the mind. It refers to the mental processes that allows memory, abstract problem solving, decision making. The human mind is a complex cognitive system in itself.

The following definition originates from [HW05]; a Cognitive system is

- being goal oriented and based on symbol manipulation
- being adaptive and able to view a problem in more than one way; and
- being able to plan and modify actions based on that knowledge

Cognition is field of study which is best suited for in-field studies and application. This is been widely accepted as "Cognition in the wild".

As people rarely work alone, it makes sense to treat a complete system that involves both humans, organizations and technology as a joint cognitive system. [HW05]

Joint cognitives systems are treated by the relative new cognitive systems engineering field, and distance itself from the classic human-machine interface (HMI). It regards the entire system, including the operator, as a whole - rather than two separate isolated systems.

2.2.2 Circadian rhythm

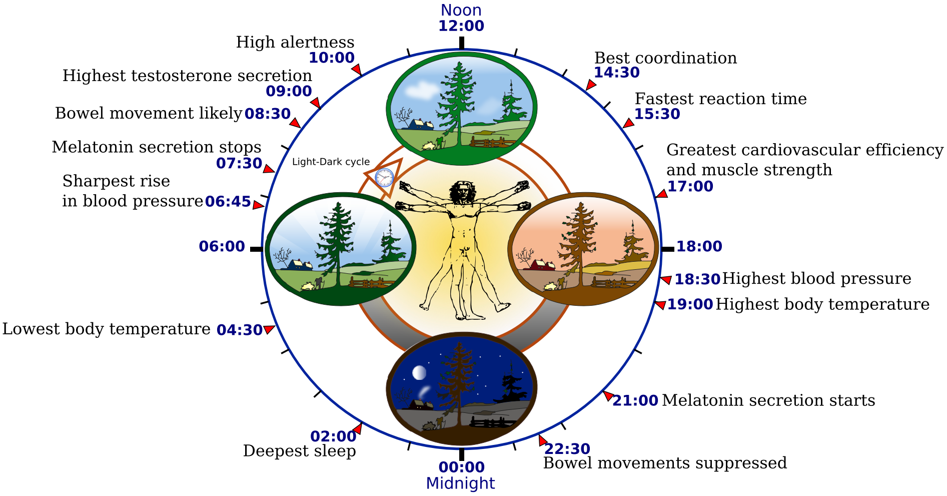

Noon
12:00
High alertness
10:00
Highest testosterone secretion
09:00
Bowel movement likely 08:30
Melatonin secretion stops
07:30
Sharpest rise
in blood pressure 06:45
06:00
Lowest body temperature 04:30
02:00
Deepest sleep
00:00
Midnight
22:30
Bowel movements suppressed
21:00 Melatonin secretion starts
19:00 Highest body temperature
18:30 Highest blood pressure
18:00
Greatest cardiovascular efficiency
and muscle strength
17:00
Fastest reaction time
15:30
Best coordination
14:30
Light-Dark cycle

Figure 2.1: Human biological clock (Licence: GFDL)

The circadian rhythm is a the natural daily rhythm that is found in, humans, plants, other mammals alike. It has tremendous impact on the performance of an individual and is thus non-negligible when discussing human factors in a system. See figure 2.1, for a visual presentation of the human biological clock.

Circadian rhythm has a large impact on, for example aeroplane pilots flying across time zones, and thus loses their natural sense of daylight. This leads to fatigue, decreased responsiveness and performance ([MBD10]).

2.3 User interfaces

In the recent years, as both cognitive psychology and technical progress has advanced, user interfaces is beginning to receive more attention.

[Nor02] presents the human action cycle manifested into the following seven stages of action. The following ordered list presents these steps in specific relation to user interface design - from the user perspective.

1. Form a goal: What does the user want to achieve?
2. Translating the goal into a task or a set of unordered tasks: Which actions are needed to reach the goal?
3. Order the tasks:
4. Executing the action sequence:
5. Perceiving what happened:
6. Interpreting the outcome according to the users' expectations:
7. Evaluating what happened against what was intended

Usability engineering is a field in growth, and has been given a great deal of attention the last five years. A lot the psychological data and lessons learned could be of very high use the design of human-machine interfaces for safety-critical applications.

2.4 Dynamic reconfiguration

One of the things humans do very differently, and much better, than machines, is adapt. Humans are able to learn from experience and apply new knowledge.

Machines however, are usually only designed for one purpose, and reconfiguration is usually not an option, unless it is really necessary - e.g. when situation of emergency arises. Modern end-user consumer products are usually designed with a dynamic reconfiguration strategy in mind - where applicable.

A market where dynamic in-field reconfiguration is widely used, is the growing market for smart phones. These are typically cellular phones with additional functionality, such as being able to install third party software. The makers of these products have acknowledged the fact that their systems, due to increasing constraints on time-to-market window, will have to be shipped before extensive testing has been performed.
But by enabling their system to be reconfigured in-field, they will be able to fix errors in their product, that have been identified by the users.

Chapter 3

Accident modelling

Accident models seek to explain the unexplainable and introduce a formalism into the accident reporting. [Hol04] discusses a number of accident models in detail, and finds - among other things - the following:

- Graphical representation is a big challenge. Not only when it comes to communicating the findings, but also when the cause of the accident has to be traced. Boxes and sequential also models tends to lead to boxed and sequential thinking.
- Complex models are not easily represented graphically, nor are they easy to communicate without loss of information quality, or correctness.
- Organizational structure is often overlooked and neglected in accident models

Historically, this has not always been so. Most accident models focus almost entirely on the closed-loops within the systems.

3.1 Moving up the abstraction ladder

When deploying systems for controlling physical processes, there is a large risk of alienating the people working with the process. Especially if the have not been in touch with the actual process itself, but only with the abstract representation.

A number of unintended constraints will inevitably appear when trying to represent a real system from a model. Especially those of symbols and display screen real estate. Whereas when you are present at the machine itself, you can actually see what is going on.[HW05]

3.2 Root Cause Analysis

Before starting a Root Cause Analysis, or RCA, a stop rule is usually defined. A stop rule is the point where you do no dig any further. An analogy from [Hol04] shows a RCA as a tree where the single leaf can be considered an event. The cause is then traced back to the roots of the tree to the root, but usually not any further. A root event can typically be traced further back, and fans out to a number of contributing factors.

3.3 Fault tree analysis

Fault tree analysis, or FTA, uses a graphical representation distinguishing between events, gates and transfers. It uses the common set of logic gates (AND, OR, NOT, XOR) as well a few specialized additions.

3.4 FMEA

Failure Mode and Effect Analysis assumes views the system as a number of individual components, and by implying the failure of one of these components, the effect is sought determined. This can be very useful in detecting single-points-of-failure components and perform remedial actions on these. FMEA can also be applied on a functional level, rather than on component level - depending on domain and context.

When the failure modes are identified, each is systematically quantified by the following three measurements.

3.4.1 Occurrence

The purpose of this step is to determine the frequency of the failure. This is usually usually based around historical or empirical numbers. Each failure mode is given a rating between 1-10 based on the definitions given in table 3.1.

Rating	Meaning
1	No known occurrences on similar products or processes
2-3	Low (relatively few failures)
4-6	Moderate (occasional failures)
7-8	High (repeated failures)
9-10	Very high (failure is almost inevitable)

Table 3.1: FMEA occurrence categories

3.4.2 Severity

This step serves to determine the severity of the failure mode - or the effect. If, for example, it is something that cause minor glitches to observant user(s), it will most likely be categorized as "No effect" - or 1. If on the other hand, it causes injury to the users it is considered hazardous and will be on a 9 or 10.

Rating	Meaning
1	No effect
2	Very minor
3	Minor
4-6	Moderate
7-8	High
9-10	Very high and hazardous

Table 3.2: FMEA severity categories

3.4.3 Detection

The next step is to identify how detectable the failure mode is.

Rating	Meaning
1	Certain - fault will be caught on test
2	Almost Certain
3	High
4-6	Moderate
7-8	Low
9-10	Fault will be passed to customer undetected

Table 3.3: FMEA detectability categories

3.4.4 Risk priority number (RPN)

When the previous three steps are completed, a worksheet is typically produced, which serves as a basis for calculating risk priority numbers - or RPN. It is calculated in with the following formula.

$$RPN = O \cdot S \cdot D \tag{3.1}$$

Where O is the occurrence frequency factor, S is the severity factor and D is the detectability factor.
The activities for FMEA is covered in extensive detail in [MS80].

FMEA has a "Big brother" method that includes criticality in the analysis. It is called FMCEA - or Failure Mode Criticality and Effect Analysis.

3.5 HAZOP

HAZOP is a short form of HAZard and OPerability study. It uses a set of guide words to create a systematic review of a system. It is originally developed to analyse chemical process systems, but is now used for a variety of systems - including software. The method itself, and the list of guide words are standardized[1].

HAZOP works by first defining components and interfaces - and how they are interconnected. Due to its origin in chemical processing industry, it focuses on the flow between thesis components via the connections. This however, has proven to be a relevant general model for other field - such as computer science.

[1] BS: IEC61882:2002 Hazard and operability studies (HAZOP studies)

Here the flow is not a material, but information - or a electrical signal.
This has added four additional guide words to HAZOP; early, late, before and after. Upon identification of interconnections of components, a systematic process is started by taking every relevant guide word for each connection and record any findings.

Guide Word	Meaning	Example
No or Not	Complete negation of the design intent	No result or reply when expected
More	Quantitative increase	Information/material flow rate too high
Less	Quantitative decrease	Information/material flow rate too low
As well as	Qualitative modification/increase	Extra product/events in addition to expected
Part of	Qualitative modification/decrease	Incomplete sequence/activity
Reverse	Logical opposite of the design intent	Reverse flow of traffic, material or current
Other than	Complete substitution	Other result/outcome than expected
Early	Relative to the clock time	Signal too early in reference to system clock
Late	Relative to the clock time	Signal too early in reference to system clock (deadline miss)
Before	Relating to order or sequence	Signal arrives earlier in a sequence than intended
After	Relating to order or sequence	Signal arrives later in a sequence than intended

Table 3.4: HAZOP guide words

The data in table 3.4 is based on the similar table found in [Sto96].

3.5.1 Discussion

HAZOP is a strong and formalized method for doing systematic assessment of a complete system. However, it focuses primarily on the couplings between components, and thus fails to capture any propagation there may arise. This can be remedied by the use of fault trees (3.3)

3.6 STAMP

System-Theoretic Accident Model and Process regards the safety problem as a control problem. It seeks to define, and on the remedial side, enforce safety constraints.
It consists of three basic constructs; safety constraints, hierarchical safety control structure and process models.

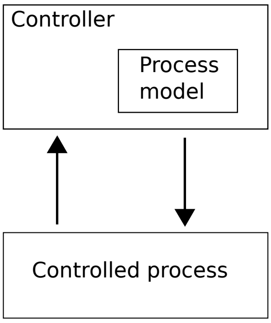


Figure 3.1: Process model as seen by STAMP

Process models are internal representations of a controlled process held by a controlling instance. This is depicted in figure 3.1. Safety violations can occur when the process model are not in correspondence with the actual system. Communication (control lines) are modelled by up- and downstream arrows.
STAMP also seeks to model the control structure and tries to captures the individual contribution of each level technical, managerial, organizational and regulatory. It also embraces

For a full presentation of STAMP see [Lev12].

These, and other analysis are covered in-depth by [Skl04].

3.7 Functional Resonance Analytic Model

FRAM is designed for systems which both include human and organizational factors. It seeks to avoid the unintended effects of other graphical representations - e.g sequential thinking. It represents a system as a set of interconnected functions. Each function is represented as a FRAM node - or as the creator informally states; a hexagonal snowflake. An example is shown i figure 3.2.

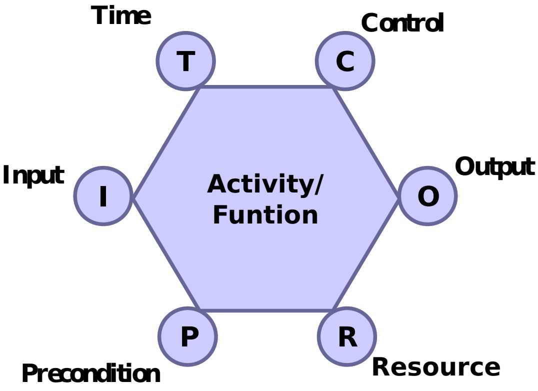


Figure 3.2: FRAM node

Each of these nodes consists of a six edge connection points - one output and five inputs. These model how functions are inter-coupled. A brief explanation of each connector follows.

- Time: Time constraints. Can be real-time or schedule constraints.
- Precondition: A connected function must supply output to this input before the function can start.
- Control: Implies "controlled by", and specifies input from supervising function. Can be plans, procedures, guidelines or other functions.
- Input: That which is used or transformed to produce the output. Links to previous functions.
- Output: The basic output of this function - or what it produces. Connects to input of other nodes.
- Resource: Resources consumed by this function (examples are; matter, energy, hardware, software, manpower, information)

Unused connectors can explicitly be marked as Not Applicable (N/A).

FRAM analysis then consists of four basic steps:

3.7.1 Identify essential system functions

The objective of the first step is to identify essential system functions, and characterize each of them by the six basic aspects specified in figure 3.2. This can be done in a table, and converted to hexagonal objects later on.

3.7.2 Characterize the context dependent variability of each node

The next step is to characterize the context dependent variability of the scenario as a whole, and for each node. This is done from a list from a list of common performance conditions - or CPCs.

CPC	Category
Resource availability	H-T
Training and experience (competence)	H
Quality of communications	H-T
Quality of human-machine interfaces	T
Access to procedures and methods	H
Working conditions	H-T
Number of simultaneous objectives	H-O
Time available	H
Circadian rhythm	H
Quality of team collaboration	H
Quality of organizational support	O

Table 3.5: Different CPC's and their category context

Every condition is categorized; stable or variable but adequate, stable or variable but inadequate or unpredictable. Focus should be on whether they have positive or negative impacts on performance. A small characterization should also be provided.

After identifying the CPCs, the variability must be determined in a qualitative way in terms of stability, predictability, sufficiency, and boundaries of performance.

When applied on an accident analysis, the analysis focuses on comparing the observed and the normal performance.

3.7.3 Define functional resonance between nodes

The third step will link the functions and define the functional resonance. The purpose of the couplings between the nodes is to determine the potential for functional variability. This step is aimed at locating the unpredictable or inadequate couplings and where variability can be an issue.

3.7.4 Identify damping factors

The final step will be be the remedial one, where identification of variability barriers - or damping factors - will take place. For a more in-depth description on barriers see section 1.3.1.

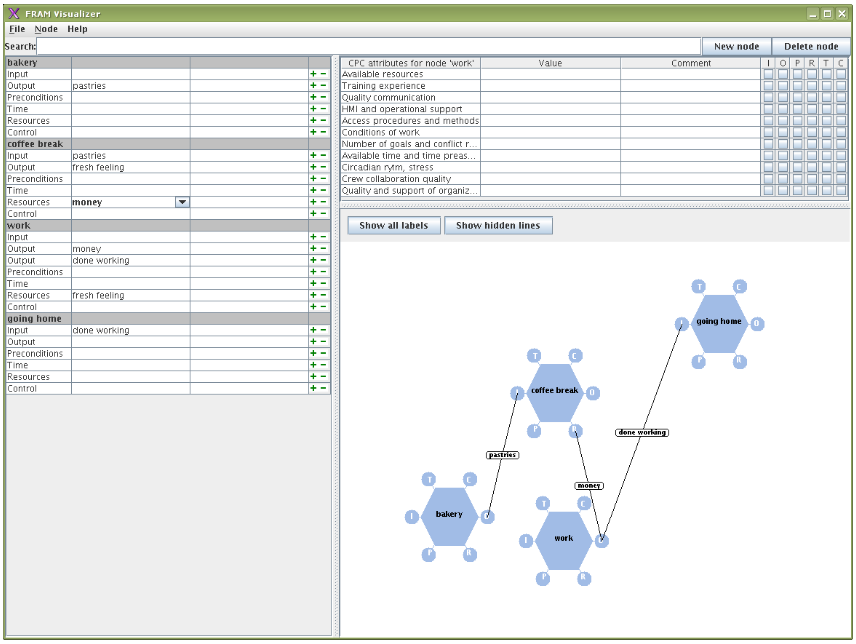
Usually one or more of these four classes of barriers are deployed;

- Monitoring: a management layer barrier that can provide early warning signals to higher levels of management. Usually implemented with the data already available.
- Detection: is the technological approach to monitoring, but still requires a manual reasoning and/or interpretation.
- Dispersion: involves creating a new physical barrier that prevents propagation - e.g. sprinkler system and airbags. Measures done here are meant to increase the internal resilience of the system.
- Correction: can be everything from revised legislations to replacing technological components. It can also involve firing an employee identified as a contributing or causing factor.

3.8 Retrospective FRAM

Although FRAM is intended to be used as a pre-deployment tool to enhance the resilience of a system (or sub-system), its modelling characteristics enables it to be applied retrospective on an accident or near-miss event.

The FRAM analysis now involves another step prior to the original four;



bakery			
Input			+ −
Output	pastries		+ −
Preconditions			+ −
Time			+ −
Resources			+ −
Control			+ −
coffee break			
Input	pastries		+ −
Output	fresh feeling		+ −
Preconditions			+ −
Time			+ −
Resources	**money**		+ −
Control			+ −
work			
Input			+ −
Output	money		+ −
Output	done working		+ −
Preconditions			+ −
Time			+ −
Resources	fresh feeling		+ −
Control			+ −
going home			
Input	done working		+ −
Output			+ −
Preconditions			+ −
Time			+ −
Resources			+ −
Control			+ −

CPC attributes for node 'work'	Value	Comment	I	O	P	R	T	C
Available resources			□	□	□	□	□	□
Training experience			□	□	□	□	□	□
Quality communication			□	□	□	□	□	□
HMI and operational support			□	□	□	□	□	□
Access procedures and methods			□	□	□	□	□	□
Conditions of work			□	□	□	□	□	□
Number of goals and conflict r...			□	□	□	□	□	□
Available time and time preas...			□	□	□	□	□	□
Circadian rytm, stress			□	□	□	□	□	□
Crew collaboration quality			□	□	□	□	□	□
Quality and support of organiz...			□	□	□	□	□	□

Figure 3.3: FRAM Visualizer interface

Define the purpose of modelling and describe the situation being analysed. Either an event that has occurred (incident/accident) or a possible future scenario (risk).

3.9 FRAM tools

A special-purpose piece of computer software exists to visualize the FRAM models and conveniently describe the functions and couplings using textual input and graphical representation. It can be seen in figure 3.3.

Chapter 4

Example accident model

Using FRAM retrospectively will, hopefully, identify some of the critically constrained couplings between functions in a system. Applying this knowledge, it will be possible not only to build safer, but also more resilient systems.

This chapter will take a relevant near-miss and model it with the procedure suggested by FRAM.

4.1 Near miss at Train crossing

Modelling near miss incidents is not very common, though very rewarding in terms of drawing experience from them. A more elaborate explanation of what a near miss is, see section 1.1.2

4.1.1 Background

At Grenåbanen on Tuesday the 26th of March 2010, at 14:40, an ambulance was intentionally led over railway crossing that should have been secured. This situation led to a near-miss, and luckily no one was harmed.

4.1.1.1 Official accident report

Train RV 4940 in transit from Grenå towards Aarhus was signalled that crossing 128a was secured.

Shortly after, the train driver realized that 2 railway employees were located on the track. The driver did not do anything further as he assumed they would move when they saw the train.

As the train approached the crossing, an ambulance with siren signal entered the crossing - from the road side. The train driver used the emergency brake, hereby avoiding collision with the ambulance. According the the train driver, the collision was imminent.

The 2 railway employees reported that, they thought they would be able to assist the ambulance in reaching its destination faster, by leading it into the crossing before the train arrived, but misjudged the situation.

The document "udrykningsbekendtgørelsen" (the official notice regarding emergency) states that the driver of an emergency vehicle, must at all times abide signals or other instructions, at railway crossings

Following the initial investigations and evaluations of the data available - the accident investigation committee reached the following conclusion; further studies would not necessarily lead to preventative recommendations, or result in findings leading to significant improvements in railway safety.

With reference to Danish railway legislation, the accident investigation committee decided not to perform further studies.

4.1.2 Assumptions

As there are only crossing bars in the driving direction - it is assumed that the crossing bars were in place as the event took place. Figure 4.1 illustrates the assumed path of the ambulance and the placement of the crossing bars. The path taken by the ambulance indicates that it had to slow down, maybe even significantly. This adds to the total variability of the pseudo-barrier which here is time.

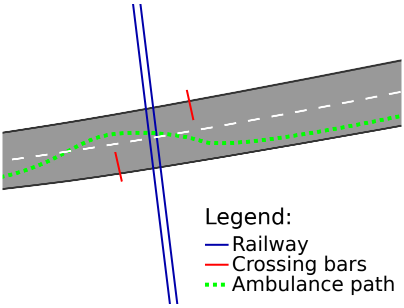


Figure 4.1: Assumed ambulance path (train approaching from the north)

4.1.3 FRAM model

4.1.3.1 Functions

It is assumed that the two employees manually initiated the securing of the crossing. The function/activity is represented in table 4.1. To initiate the securing, the car drivers must first be notified of the pending closing of crossing bars. This must be done some time prior to the train arrival, as response time on clearing is relativity high.

Function	Initiate securing of crossing
Input	
Output	Signal the car drivers
Precondition	Incoming train
Resource	
Time	Safe time before train arrives
Control	Regulations

Table 4.1: FRAM table of the initiating activity

When the road side is cleared, the crossing bars are lowered - effectively blocking it. It is essential that the cars have left the crossing. This function is modelled in table 4.2.

Table 4.3 models the function that allows train passage. As the crossing is interlocked, it must be secured for train passage before the train will be able to enter it. This is usually enforced by both a signal, and ATC (1.1.5.3).

When the train has passed, the crossing will be able to unblock the road side

Function	Block passage from road side
Input	Signal the car drivers
Output	Lower the crossing bars
Precondition	
Resource	
Time	Time sufficient to clear crossing
Control	

Table 4.2: FRAM table of the blocking function

Function	Allow passage from train side
Input	Train has enters crossing
Output	Train has left crossing
Precondition	Crossing bars lowered
Resource	
Time	Must pass within blocking-time window
Control	ATC

Table 4.3: FRAM table representing the activity of allowing train passage

after a safety delay. Some crossings will automatically open after a time-out has occurred, even if no train has passed - of course while blocking the railway. Table 4.4.

Function	Unblock road side
Input	Train has left crossing
Output	Unblock road side passage
Precondition	
Resource	
Time	Safety delay
Control	Time-out

Table 4.4: FRAM table representing the function of unblocking the road side

This graphical representation shows quite clearly that time is an essential aspect of every activity and function - hence every small variability in a function will resonate and propagate onto the next.

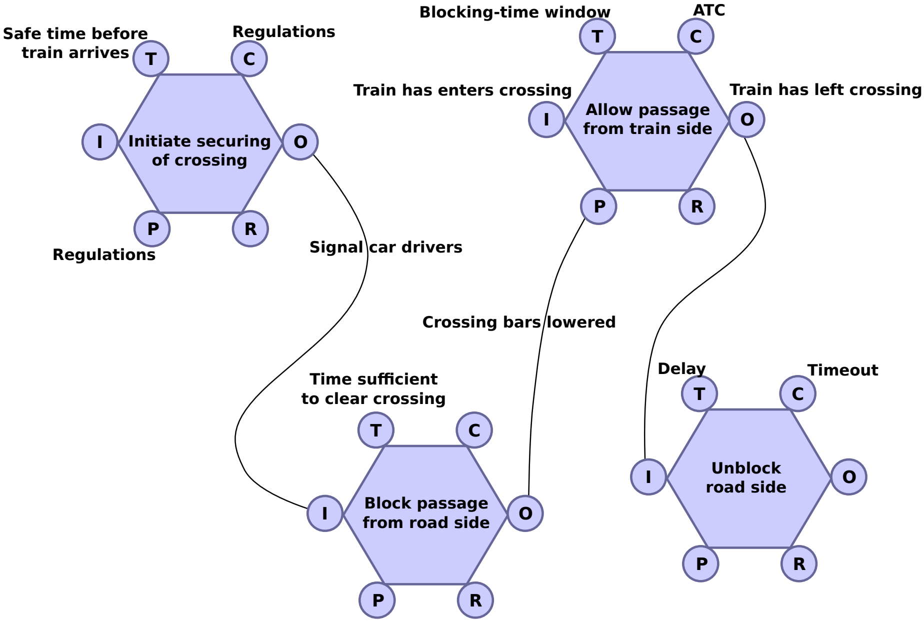

Safe time before train arrives
Regulations
T
C
I
Initiate securing of crossing
O
P
R
Regulations
Blocking-time window
ATC
T
C
Train has enters crossing
I
Allow passage from train side
O
Train has left crossing
P
R
Signal car drivers
Crossing bars lowered
Time sufficient to clear crossing
T
C
I
Block passage from road side
O
P
R
Delay
Timeout
T
C
I
Unblock road side
O
P
R

Figure 4.2: Graphical FRAM representation

4.1.4 Barriers

There are a number of barriers in place here:

- A physical barrier that is intercoupled with a incorporeal barrier; the crossing bars that depend on driving in the right side of the road (following traffic laws).
- Time - the essential barrier. Safety functions depend largely on having the time to complete their cycle.
- Two incorporeal barriers; “udrykningsbekendtgørelsen” and the railway legislation.
- The ATC (1.1.5.3)

4.1.5 Common performance conditions

The common performance conditions identified in table 4.5 was found very similar for each independent node, and thus will not be repeated for each of these.

Common Performance Conditions	Characterisation	Rating
Resource availability	Staff	Adequate
Training and experience (competence)	Adequate	Adequate
Quality of communications	Time constraint	Inadequate
Quality of human-machine interfaces	-	-
Access to procedures and methods	Clear instructions	Adequate
Working conditions	Repetitive	Unpredictable
Number of simultaneous objectives	Fixed without slack	Inadequate
Time available	Time constraint	Unpredictable
Circadian rhythm	Short shifts	Adequate
Quality of team collaboration	-	-
Quality of organizational support	Independent	Adequate

Table 4.5: Overall performance conditions of the scenario

The FRAM analysis specify that every variability should be specified in a quantitative way. However, since the amount of data is very limited, such a quantification would, in this case, be based around wild guesses, and not serve a constructive purpose.

4.1.6 Functional resonance

Applying the data from table 4.5 result in a strengthening of our assumption that timing skews will resonate throughout the system entirely. A performance hit, which was exactly what happened, will cause a ripple, and potentially a hazardous situation.

4.2 Discussion

The near-miss here is beyond the scope of the intended operation/design of ATC(1.1.5.3) and is a perfect example of high variability in a system. Everything works as intended, until the railway workers are affected by a disturbance that ultimately leads to two time constraint pressures; one from the Ambulance, and one from the approaching train.

On the remedial side, there could be a gain by adding a second set of crossing bars, meaning that both driving lanes of the road will be blocked. This will prevent this barrier to be overridden entirely. Adding a second set of bars would introduce the hazard of “trapping” cars inside the secured crossing, but could be avoided by delaying the lowering of the second set of bars for a short time after the first set.

Chapter 5

Conclusion

Usability engineering is a field where cognitive systems engineering could benefit from. A lot the research done here, share the same traits as “cognition in wild”, and can thus be used for optimizing human-machine interfaces.

FRAM and STAMP share the same goal and have identified some of the same issues; organizational structure and its impact on design and safety and the human performance issue. The graphical aspect of FRAM is more formal than STAMP, and it appears that the notation form of FRAM could be integrated into STAMP relatively simple, the reverse seems improbable though.

The evolution of accident modelling methods seems inevitable, as they merely seek to comprehend the increasing complexity of the world around us. There is no empiric data on the new accident model’s adoption today, but chances are, that they will get a good foothold within the next 5-10 years.

From a personal perspective, having only an academic background on both linear and non-linear accident models, it is very difficult to be able to identify the relevant details in a specific situation. I can only imagine that a practitioner will find FRAM a valuable tool for explaining the unexplainable.

List of Figures

Bibliography

[Hol04] E. Hollnagel. *Barriers and accident prevention.* Ashgate Pub Ltd, 2004.

[HW83] E. Hollnagel and D.D. Woods. Cognitive systems engineering: New wine in new bottles. *International Journal of Man-Machine Studies*, 18(6):583–600, 1983.

[HW05] E. Hollnagel and D.D. Woods. *Joint cognitive systems: Foundations of cognitive systems engineering.* CRC Press, 2005.

[Lev95] N.G. Leveson. *Safeware: system safety and computers (Chapter 16).* ACM, 1995.

[Lev12] N.G. Leveson. *Engineering a safer world: Systems thinking applied to safety (Chapter 4).* MIT Press (MA), 2012.

[LRH09] J. Lundberg, C. Rollenhagen, and E. Hollnagel. What-you-look-for-is-what-you-find - the consequences of underlying accident models in eight accident investigation manuals. *Safety Science*, 47(10):1297–1311, 2009.

[MBD10] MM Mallis, S. Banks, and DF Dinges. Aircrew fatigue, sleep need and circadian rhythmicity. *Human factors in aviation*, 2:401–436, 2010.

[MS80] (U.S). Military Standard. Mil-std-1629a. *Procedures for Performing a Failure Mode, Effect and Criticality Analysis*, 1980.

[Nor02] D.A. Norman. *The design of everyday things.* Basic books, 2002.

[Skl02] S. Sklet. Methods for accident investigation. *Trondheim: Gnist Tapir*, 2002.

[Skl04] Snorre Sklet. Comparison of some selected methods for accident investigation. *Journal of Hazardous Materials*, 111(1–3):29 – 37, 2004. <ce:title>A Selection of Papers from the JRC/ESReDA Seminar on Safety Investigation Accidents, Petten, The Netherlands, 12-13 May, 2003</ce:title>.

[Sto96] N.R. Storey. *Safety critical computer systems (Chapters 3 and 4)*. Addison-Wesley Longman Publishing Co., Inc., 1996.