Human Error, Safety and Systems Development

Edited by
Chris W. Johnson
Philippe Palanque

HUMAN ERROR, SAFETY AND SYSTEMS DEVELOPMENT

IFIP – The International Federation for Information Processing

IFIP was founded in 1960 under the auspices of UNESCO, following the First World Computer Congress held in Paris the previous year. An umbrella organization for societies working in information processing, IFIP's aim is two-fold: to support information processing within its member countries and to encourage technology transfer to developing nations. As its mission statement clearly states,

> *IFIP's mission is to be the leading, truly international, apolitical organization which encourages and assists in the development, exploitation and application of information technology for the benefit of all people.*

IFIP is a non-profitmaking organization, run almost solely by 2500 volunteers. It operates through a number of technical committees, which organize events and publications. IFIP's events range from an international congress to local seminars, but the most important are:

- The IFIP World Computer Congress, held every second year;
- Open conferences;
- Working conferences.

The flagship event is the IFIP World Computer Congress, at which both invited and contributed papers are presented. Contributed papers are rigorously refereed and the rejection rate is high.

As with the Congress, participation in the open conferences is open to all and papers may be invited or submitted. Again, submitted papers are stringently refereed.

The working conferences are structured differently. They are usually run by a working group and attendance is small and by invitation only. Their purpose is to create an atmosphere conducive to innovation and development. Refereeing is less rigorous and papers are subjected to extensive group discussion.

Publications arising from IFIP events vary. The papers presented at the IFIP World Computer Congress and at open conferences are published as conference proceedings, while the results of the working conferences are often published as collections of selected and edited papers.

Any national society whose primary activity is in information may apply to become a full member of IFIP, although full membership is restricted to one society per country. Full members are entitled to vote at the annual General Assembly, National societies preferring a less committed involvement may apply for associate or corresponding membership. Associate members enjoy the same benefits as full members, but without voting rights. Corresponding members are not represented in IFIP bodies. Affiliated membership is open to non-national societies, and individual and honorary membership schemes are also offered.

HUMAN ERROR, SAFETY AND SYSTEMS DEVELOPMENT

IFIP 18th World Computer Congress
TC13 / WC13.5 7th Working Conference on
Human Error, Safety and Systems Development
22–27 August 2004
Toulouse, France

Edited by

Chris W. Johnson
University of Glasgow, Scotland

Philippe Palanque
Université Paul Sabatier, France

KLUWER ACADEMIC PUBLISHERS
NEW YORK, BOSTON, DORDRECHT, LONDON, MOSCOW

eBook ISBN: 1-4020-8153-7
Print ISBN: 1-4020-8152-9

Created in the United States of America

Visit Springer's eBookstore at: http://www.ebooks.kluweronline.com
and the Springer Global Website Online at: http://www.springeronline.com

Contents

Contributing Authors

Yamine Aït-Ameur
LISI/ENSMA, France

Hans H. K. Andersen
Risø National Laboratory, Denmark

Nasrine Bagheri
University of Toronto, Cognitive Engineering Laboratory, Canada

C. Baron
LESIA-INSA, Toulouse, France

Sandra Basnyat
LIIHS-IRIT, Université Paul Sabatier, France

Guy Boy
Eurisco International, Toulouse, France

Benoit Breholée
ONERA-CERT-DTIM, France

Michael Cebulla
Technische Universität Berlin, Fakultät IV, Institut für Softwaretechnik und theoretische Informatik, Germany

Eugène Chouraqui
LSIS, Marseille, France

Patrick Girard
LISI/ENSMA, France

Louis Goossens
Delft University of Technology, The Netherlands

Angela Miguel
Dept. of Computer Science, University of York, UK

Gilles Motet
LESIA-INSA Toulouse, France

Jari Nisula
Operational Monitoring & Human Factors, Flight Operations Support, Airbus, France

Philippe Palanque
LIIHS-IRIT, Université Paul Sabatier, France

Fabio Paternò
ISTI-CNR, Pisa, Italy

Mats Persson
National Defence College, Sweeden

Veronika Prinzo
Federal Aviation Administration Civil Aerospace Medical Institute, USA

Georgios Rigas
National Defence College, Sweeden

Carmen Santoro
ISTI-CNR, Pisa, Italy

Jens-Uwe Schroeder
World Maritime University, Malmö, Sweden

Bastiaan A. Schupp
Dept. of Computer Science, University of York, UK

Shamus Smith
Dept. of Computer Science, University of York, UK

Elisabeth Strunk
University of Virginia, USA

Maria F. Q. V. Turnell
LIHM - Laboratorio de Interfaces Homem Maquina da Universidade Federal de Campina Grande, Brazil

Peter Wright
Dept. of Computer Science, University of York, UK

This page intentionally left blank

Preface

The papers in this collection address the problem of developing systems that support human interaction with complex, safety-critical applications. The last thirty years have seen a significant reduction in the accident rates across many different industries. Given these achievements, why do we need further research in this area?

There is little room for complacency. For example, it has been difficult to sustain reductions in the incident rate across the aviation industry. This not only indicates an impasse in attempts to achieve 'zero' accidents. It is also a source of long-term concern because a stable incident rate combined with rising numbers of departures will yield increases in the frequency of adverse events. In other areas, the incident rates are rising in spite of the best efforts of safety managers. For instance, the frequency and rate of serious accidents in the US Army declined steadily in the decade prior to 2000. However, since that time there has been a rise in the number of soldiers killed or seriously injured by these adverse events. The nature of military operations has changed over this interval. Not only have operations in the Middle East increased risk exposure but the changing technology used by personnel has also affected the nature of many adverse events. In May 2003, Defense Secretary Rumsfeld focused concern: "World-class organizations do not tolerate preventable accidents. Our accident rates have increased recently, and we need to turn this situation around". He set the challenge to "to reduce the number of mishaps and accident rates by at least 50% in the next two years".

The US Army has recently established a number of initiatives that are intended to reduce the frequency of adverse events. For example, the 'Safety Sends' initiative is using Internet communication techniques to

update units on potential hazards. The latest update reported on the fatal accidents from 8 March 2004 to 8 April 2004, 29 'Class A' mishaps resulted in 25 fatalities. 26 were ground incidents. 19 of these related to vehicle accidents and these accounted for 18 fatalities. 11 soldiers were killed in Privately Operated Vehicles. 4 of these soldiers were not wearing seatbelts. One soldier fell asleep at the wheel. 3 HMMWVs, an LMTV, and an M2 Bradley were involved in rollover accidents with 6 fatalities. There were 3 physical training related fatalities over this 4-week period. An important observation is that these accidents form part of a wider pattern in which most US army personnel are killed in road traffic accidents and in which 'roll over' incidents remain a continuing cause of injury. There are few surprises in this data.

It is, therefore, paradoxical to argue that many accidents and incidents stem from a 'lack of imagination'. The personnel involved failed to predict that their actions could place them and their colleagues at risk. Designers place operators at risk through this same lack of imagination; they do not anticipate the many diverse ways in which complex systems will be operated within their working environment. As we shall see, many of the contributions in this volume describe what happens when either the operator or systems designer fail to imagine the possible hazards associated with safety-critical applications. Recent accident reports provide further evidence to support this argument. For instance, the Gunner of a Bradley fighting vehicle was found unconscious from the start of carbon monoxide poisoning. The investigation later found that the soldier had been riding half in and half out of the overhead troop hatch. This was the exact location where the highest concentration of carbon monoxide was found. It was also discovered that the soldier was a smoker and hence may already have had elevated levels of carbon monoxide in their bloodstream. Subsequent investigations also revealed that the driver of the vehicle had discovered the seal on the engine panel was crimped but failed to recognise it as a potential hazard. The crew chief had also noticed problems with the seal and that a visible black 'streak' had been left by the blow back from the engine. After the incident, a motor pool mechanic found that the coupling between the engine and exhaust could also have contributed to the incident. This incident illustrates the 'difficulty of imagination'. In the aftermath of the incident, it is too easy to argue with hindsight bias that the individuals concerned should have anticipated this course of events. Given the pressures of more complex working environments, however, it seems reasonable to ask whether we would really have predicted the potential confluence of events that led a smoker to position themselves in the greatest concentration of carbon dioxide that stemmed from a loose exhaust coupling and was dispersed by a crimp in the engine panel seal. It is also important

to reiterate that it is correspondingly more difficult to anticipate potential accidents involving human interaction with complex, distributed, computer controlled systems. The risks of carbon monoxide poisoning from a Bradley are relatively well understood when compared, for example, with the risks associated from night vision devices or from interaction with Unmanned Airborne Vehicles.

The previous paragraphs have argued that many accidents occur because operators and designers do not imagine the manner in which a complex, safety-critical system can fail. It follows, therefore, that good design depends upon the ability to imagine and then respond to these potential failure modes before they occur. The word 'imagination' has strong links with terms such as 'subjectivity' and individual 'creativity' that are not normally associated with the engineering of safety-critical systems. In contrast, objectivity is often seen as a prerequisite for the certification, validation and verification of potential designs. It is for this reason that many of the papers in this collection present techniques that are both intended to help designers anticipate potential hazards and then document the reasons for their predictions. Imagination and domain expertise are essential to identify failure modes. For instance, Johansson et al describe how critiquing tools can be used to provide teams with the ability to identify potential problems before they occur. However, analytical methods must then be recruited to document the basis for any risk assessment. Karsten et al and Jambon et al go on to provide examples of the way in which formal methods can be used to represent and reason about risk mitigation techniques.

Critiquing tools support the designer's 'imagination' in ways that can be used to identify potential failure modes. Semi-formal and formal methods can be combined with risk assessment techniques to arguably provide the objective rationales that support certification and design. However, a further strand of papers in this collection point to the limitations that affect these analytical techniques. In particular, their use needs to be closely informed by operational experience. For example, a recent incident occurred when fuel contaminated the oil of another Bradley. This reduced the lubrication available to the engine and it seized. The driver tried to continue. A rod tore through the bottom of the oil pan, taking part of the engine block with it. Friction ignited fuel residues and an explosion blew part of the engine compartment panel into the driver. He was stunned but managed to exit through the driver's hatch. The rest of the crew heard the driver yell 'fire' and the troops in the back of the vehicle tried unsuccessfully to open their exit ramp. The driver heard them and returned to his cab to operate the ramp. Within 15 minutes the fire reached the live 25mm ammunition and TOW missiles. The Bradley was destroyed but

nobody was injured. This incident is instructive because the battalion commander had performed a risk assessment using techniques that are similar to those advocated within this collection. In consequence, he had successfully identified vehicle fire as a hazard. When investigators reviewed the unit's risk mitigation guidance, there was no reference to crews rehearsing vehicle fire drills, as described in applicable manuals. The occupants of the destroyed Bradley didn't understand the vehicle's fire suppression system. The command had properly identified the hazard, but failed to properly assess it and provide proper control measures to reduce the risk. Hence, as Andersen and Gunnar argue, risk assessment is insufficient unless adequate barriers are deployed.

This incident is also instructive because it also acts as a reminder of the relationship between risk assessment, design and human error. At one level, the Bradley's crew failed to follow standard operating procedures because they did not deploy the vehicle's fire suppression systems. Similarly, there should have been closer coordination in evacuating the troops in the back of the vehicle. However, as Nasrine et al and Urs point out human error analysis is seldom so straightforward. After the Bradley incident the enquiry did not emphasise the crew 'errors'. There are good reasons why investigators emphasised the need to practice using the fire suppression systems. The Bradley has two different applications. The first protects the squad compartment. The second separate system is installed in the engine compartment. Each has separate fire bottles. The ones for the squad compartment are next to the turret, while the fire bottle for the engine compartment is underneath the instrument panel. As mentioned, these systems are independent. If one system is activated then it will not automatically start he other. The squad system can be set to work in either automatic or manual mode. In automatic mode, the system will discharge a Halon suppression agent from the two rear fire bottles as soon as the sensors detect a fire. The system can be activated manually by pulling the fire extinguisher handle in the right rear of the squad compartment or by pulling a handle outside the vehicle. The need to practice using these systems stems in part from the adverse effects that Halon discharge can have upon the occupants of the vehicle if they do not exit in sufficient time. In contrast, in order to operate the engine fire suppression system the driver must first shut down the vehicle and reach under the instrument panel. They must then turn a dedicated lever to the left or they can pull on a handle outside the driver's hatch.

Several of the papers in this collection, including Turnell et al, Johnson and Prinzo, make the point that risk assessments must be informed by a close analysis of the working practices and management structures in end-user organisations. The previous incident provides a further illustration of this

point; the Battalion commander identified the hazard but failed to mitigate the potential consequences by ensuring adequate training. This incident also illustrates a number of further issues that are addressed in this collection. In particular, the papers by Nisula and by Knight et al, all address the role of incident and accident reporting in the development of safety-critical interactive systems. The fire in the Bradley occurred because the lubricating qualities of the engine oil were compromised as a result of fuel contamination. A number of precursor events might have alerted personnel to the potential dangers. Another driver had been using the same Bradley and had performed a number of preventive maintenance checks. He identified a potential fuel leak and noticed fuel in the engine oil. Dismounted infantry had noticed a strong fuel smell in the crew compartment. Company maintenance personnel were informed but couldn't find the leak. They did find evidence of the oil contamination but the pressure of mission requirements forced them to return the vehicle into service. The crew attempted to deliver the vehicle to a field service point but this had been moved from its original location. The key point here is that the techniques and methods that are described in this collection can only be assessed within the context of the organisations that will use them. In this example, the army understood the importance of risk assessment as a means of structuring and documenting the necessary steps of 'imagination' that help to predict potential failures. Unfortunately, operational demands and the complex spectrum of risks that typify many military operations prevented adequate barriers from being developed. Similarly, the maintenance and operational personnel within the unit understood the importance of incident reporting. However, a complex combination of contingencies again prevented necessary actions from being taken to address the potential hazard.

To summarise, many accidents and incidents stem from problems that are well known to designers and to operators. However, many adverse events reveal a failure to 'imagine' the many different ways in which an incident could occur. This lack of imagination stems in part from attribution bias, we believe that others are more likely to be involved in adverse events than we are. It also stems from the complex ways in which component faults and operator 'error' combine to create the preconditions for failure. The papers in this collection provide techniques to address these problems, for example by extending the scope of human error analysis and risk assessment. We have argued, however, that these techniques will not be effective unless organisations scrutinise the resources that are devoted to mitigate risks once they have been identified. Other papers describe how incident and accident analysis can extend the scope of our imagination by providing important insights into previous failures. Again, however, organisational barriers

often intervene so that these lessons can be difficult to act on in an effective manner.

The papers in this collection also offer a number of further insights. Firstly, they illustrate the generic nature of many of the issues involved in human 'error'. Different contributions describe overlapping aspects in aviation, robotics, maritime applications, the leisure industries, military operations, healthcare etc. Secondly, it might be argued that few lessons are effectively shared between these different domains. For example, the problems that were apparent in interaction with aviation navigation systems are now being observed in maritime applications. Thirdly, the papers in this collection help to identify useful national and international initiatives, for example Hart presents recent developments within the aviation industry to exchange data between countries. However, these pioneering industries are faced with considerable challenges. Rather than supporting a single national, or federal, system for reporting adverse events in healthcare, individual US states are developing separate schemes. These are often poorly integrated with existing Federal systems that are used, for example, to report device related problems. In consequence, clinical staff must choose between five or more different reporting systems when deciding to report an iatrogenic incident. Similarly, many European states perceive there to be a threat to national sovereignty when schemes are proposed to adopt common reporting practices across different air traffic management organisations.

The opening sections of this preface argued that unexpected combinations of well-known failures often surprise us. The middle sections of this preface described how the papers in this collection address this problem, by risk assessment, formal and semi-formal modelling and by incident analysis. The closing sections of the preface have illustrated some of the organisational barriers that complicate the use of these techniques. The final paragraphs have opened up this critique to identify some of the political and structural issues that can hinder work in this area. This conference cannot hope to address all of these issues. We have opened up a dialogue in previous meetings now it is time to establish a clearer research agenda, in particular to determine how well many of the proposed techniques would survive in organisations as complex as the US military.

Chris Johnson and Philippe Palanque, 18th May 2004.

Acknowledgments

The HESSD 2004 Working Conference has been sponsored by the Resaerch Training Network ADVISES (Analysis Design and Validation of Interactive Safety-critical and Error-tolerant Systems). More information about ADVISES is available at the URL: http://www.dcs.gla.ac.uk/advises/

This page intentionally left blank

THE ROLE OF NIGHT VISION EQUIPMENT IN MILITARY INCIDENTS AND ACCIDENTS

C.W. Johnson
Department of Computing Science, University of Glasgow, Glasgow, G12 9QQ, Scotland.

Abstract: Night vision devices provide enormous benefits. They enable personnel to carry out operations under conditions that would not otherwise be possible. However, these benefits carry considerable risks. For instance, individuals often become over confident about their ability to use image intensification and infrared devices. In consequence, the use of night vision equipment is an increasingly common factor in military incidents and accidents. This paper uses an analysis of incident and accident data to identify requirements for the successful deployment of night vision equipment. It is argued that these applications must be integrated more closely with existing navigational systems. The successful application of this technology also depends upon adequate risk assessment and team-based training.

Key words: Accident analysis, Risk, Decision Making, Night Vision Equipment.

1. INTRODUCTION

There are two main classes of night vision devices. Image intensification (I^2) systems enhance the lighting that is available within the existing environment. Infrared (IR) devices, in contrast, will typically use heat emissions to identify objects that cannot otherwise be detected using available light sources. These systems support a wide range of military operations that would not otherwise have been possible. However, the additional capabilities provided by night vision devices also create new risks. Night operations continue to result in significantly more accidents and incidents than their daytime counterparts (Ruffner et al, 2004). We are interested in the role that night vision equipment plays in incidents and accident because many armed forces have faced recent increases in the

number and frequency of adverse events. For instance, the number of aviation fatalities from mishaps across all US Department of Defense personnel rose from 65 in 2001 to 82 in 2002. In May 2003, Defense Secretary Rumsfeld focused concern on these and similar statistics across the US military: "World-class organizations do not tolerate preventable accidents. Our accident rates have increased recently, and we need to turn this situation around" (Gilmore, 2003). He set the challenge to "to reduce the number of mishaps and accident rates by at least 50% in the next two years".

2. A BRIEF OVERVIEW OF NIGHT VISION

Military personnel, typically, rely on their visual sense during most operations. Safe flight relies upon good depth perception for landing, good visual acuity is critical if pilots are to identify terrain features. Drivers of land-based vehicles rely on depth perception to judge whether or not they can cross ditches, visual acuity is important in many aspects of land-based navigation. However, color vision, depth perception, and visual acuity all vary depending on which of the three different types of vision soldiers must rely on in a particular operation. *Photopic vision* occurs with high levels of illumination. The cones concentrated in the center of the fovea are primarily responsible for vision in bright light. High light condition will bleach out the rod cells that support peripheral vision. However, the reliance on cones produces sharp image interpretation and color vision using photopic vision. In contrast, *mesopic vision,* typically occurs at dawn and dusk or under full moonlight. This relies on a combination of rods and cones. Visual acuity steadily decreases with declining light. Color vision degrades as the light level decreases, and the cones become less effective. Mesopic vision is often regarded as the most dangerous if personnel do not adapt to the changing light conditions. As light levels fall, there will be a gradual loss of cone sensitivity. Operators should be trained to rely more on peripheral vision. If personnel fail to recognize the need to change scanning techniques "from central viewing to off-center viewing, incidents may occur" (Department of the Army, 2000). *Scotopic vision* is used under low-light level environments such as partial moonlight and starlight. Cones become ineffective, causing poor resolution of detail. Primary color perception during scotopic vision is shades of black, gray, and white unless the light source is high enough in intensity to stimulate the cones. A central blind spot, known as the night blind spot, also occurs when cone-cell sensitivity is lost. If an object is viewed directly at night, it may not be seen. If the object is detected, it will fade away when stared at for longer than two seconds.

The human eye can adapt to low light. Biochemical reactions increase the level of rhdopsin in the rods. This controls light sensitivity. Individual differences again affect the rate and degree of adaptation. It can take between 30-45 minutes for most people to achieve their maximum acuity under low levels of light. Brief flashes, for instance from strobe lights, have little effect on night vision. However, looking at a flare or searchlight for longer than a second will have an adverse effect on most people. A number of other factors, such as smoking and individual differences, also adversely affect night vision. Night myopia arises from the way in which the visual spectrum is dominated by blue wavelengths of light. Nearsighted individuals viewing blue-green light at night typically experience blurred vision. Even personnel with perfect vision will find that image sharpness decreases as pupil diameter increases. Similarly, "dark focus" occurs because the focusing mechanism of the eye often moves toward a resting position in low light levels. Special corrective lenses can be used to address this problem for individuals who suffer from night myopia. Binocular cues stem from slight differences in the images that are presented to each of the operator's eyes. Low lighting can make it difficult for personnel to perceive any visible differences. The effect is increased when objects are viewed at a distance. Low light levels also affect a number of monocular cues for depth perception. These include geometric perspective, motion parallax, retinal image size, and aerial perspective. As we shall see, the problems of depth perception play an important role in the causes of incidents and accidents.

A number of training techniques can help maximize any remaining visual resources in low levels of light. For example, the following list summarizes the Canadian Army's (2004) guidelines for night observation:

1. **Aim-off with the eyes** - Never look directly at what is to be seen. For example, if the eye looks directly at a pin-point of light it will not see the outline of the tank from which the light is coming.
2. **Do Not Stare Fixedly** - The eyes tire rapidly at night so an object will disappear if it is looked at for a long time.
3. **Avoid Looking at Any Bright Lights** - Shield the eyes from parachute flares, spotlight or headlights. Dim flashlights and turret lights and blink when firing weapons.
4. **Look Briefly at Illuminated Objects** - The time spent glancing at lighted objects such as maps or illuminated dials must be kept to a minimum.
5. **Do Not Scan Quickly** - Move the eyes in a series of separate movements to give the eye a chance to pick up a target which will appear much slower than daylight.

6. **Limit Time Spent Scanning** - Continuous scanning will cause the eye to partially black out. The eyes should be rested for 10 seconds every 2 minutes.
7. **If Necessary Use Eyes Individually** - If a lit area has to be observed, then protect the night vision of one eye by keeping it shut. One eye should be shut as an automatic reaction if a bright light suddenly appears.

2.1 Image Intensification Systems

Personnel can compensate for the limitations imposed by low light conditions either by training to make the most of their night vision or through the provision of night vision equipment. Image intensification systems support direct observations by amplifying low levels of ambient light. They do not 'turn night into day', nor do they compensate for many of the problems that affect vision in low light environments. Most image intensification systems perform poorly in total darkness. Amplification can range up to 35,000 times the available light. Higher amplification is associated with more expensive devices and can imply increased levels of distortion. The intensified image is, typically, viewed on a phosphor screen that creates a monochrome, video-like image, on the user's eyepieces.

Most image intensification systems are attached to the users' helmet. Early models included relatively heavy battery packs that restricted the users' head movements. This problem was exacerbated by the need to move the head because many devices offer a highly restricted field of vision between 40-60 degrees. A post action review of the Canadian Army's deployment in Kosovo found that "the current issue helmet and night vision goggles are not compatible and are painful to wear". (Canadian Army Center for Lessons Learned, 2001). Soldiers had to remove the devices to reduce the fatigue and frustration that built up during prolonged use. Image intensification equipment can also create problems in depth perception. Colour cues and binocular information are lost with many commercial systems. All of these limitations are being addressed by technological innovation. In particular, it is now possible to buy light weight and extended field of vision systems. These tend to be expensive and can be difficult to maintain under field conditions (Salazar and Nakagawara, 1999).

Visual acuity from night vision devices provides a vast improvement over human night vision. However, it is far from perfect. As with direct sight, higher levels of accuity are associated with closer, slower targets. The visual accuity offered by image intensification rapidly diminshes for objects over 400 feet away. Rain, clouds, mist, dust, smoke, fog all reduce accuity. For example, 'brown out' has contributed to a number of incidents where helicopter crews rely on images that are suddenly degraded by the

dust that is brought up in the wash created by their rotors (Department of the Army, 2000). A recent incident involving a Canadian military helicopter in Bosnia providesa further illustration of these environmental problems (Canadian Air Force, 2002). Reports of adverse weather conditions initially convinced the crew to remain in Banja Luka. However, if they left immediately they calculatted that they could return to their base in Velika Kladusa within their eight hour flying limit. "We strapped on our night vision goggles after refueling and decided to go for it". They were seven miles from their destination when they noticed that the lights on the hills were no longer where they expected them to be. They also began to lose sight of the lights ahead of them using their night vision equipment. The cloud lowered until it engulfed the hills that surrounded them. They realized that they could not go back to Banja Luka and so were forced to follow the only open valley in sight. The presence of mines from previous conflicts meant that they could not simply set down in any available field (Canadian Air Force, 2002). The subsequent analysis of this incident identified the danger that crews will become unduly complacent about the support provided by night vision equipment under adverse meteorological conditions.

The performance of image intensification systems can be impaired by a number of external light sources. Looking at the moon has the same effects as looking directly at the sun under daylight lighting conditions. This creates problems when soldiers move toward a bright moon that is low on the horizon. The brightness of the 'ambient' light source degrades the intensified image. It will also cast deep shadows that can hide hazards, including excavated fighting positions. This creates considerable problems for drivers trying to locate these emplacements using night vision equipment (US Army Centre for Lessons Learned, 2001). External light sources can also support the use of image intensification equipment. For instance, city lights often provide useful illuminations especially if cloud cover reflects the available light back onto a scene. However, there is a risk that personnel will fixate on these external light sources. Many of the problems associated with image intensification systems stem from their operational environment. Vehicle instrument lights and cockpit displays can create "washout" or halo effects. In many road-based vehicles it is possible to turn-off instrument illumination. However, it is a complex and expensive task to alter cockpit lighting systems without compromising the daytime use of the aircraft. These problems are compounded because red lights are frequently used in speedometers and engine instruments. Night vision systems are often particularly sensitive to these sources. Personnel must also be trained not to use red-lens flashlights in situations where image intensification equipment

is being used. In ground operations, oncoming headlights pose a major hazard because drivers must often use their goggles at times when other road users rely on their vehicle lights. These light sources can dazzle the wearer of a night vision device to the point where they will not see barriers and obstacles, including equipment or people. These are not the only source of light polution that affect the users of image intensification systems. Many aviation systems are sensitive to the anti-collision lights required by FAA regulations. These will be intensified to a point at which they can distract or even dazzle the wearer of an intensification system. Risk assessments should consider the range of problems that can arise with image intensification systems.

2.2 Infrared and Thermal Imaging Systems

Thermal imaging systems detect infrared radiation that is emitted by heat sources. Although the human eye cannot directly observe these signals, they can be focused in the same way as conventional light. Transducers detect the thermal emissions. Their output is then processed to represent the difference in temperature amongst the objects in a scene. Thermal contrast is then translated into a visual contrast that is, typically, represented in shades of gray on a monochrome display. In contrast to image intensification devices, infrared systems can be used in total darkness because they do not rely on the light reflected by an object. A further benefit is that thermal imaging systems avoid the "blooming" that occurs when strong light sources swamp intensification systems. Infrared devices also avoid some climatic problems. For instance, they can see through some types of fog. However, problems can arise under different environmental conditions. A wet runway may be cooled to such an extent that it appears to be further away than it actually is. High-humidity reduces thermal contrast and so will adversely affect image quality. Infrared systems cannot be used to identify precise details on remote objects, such as facial features, that are not distinguishable by different heat profiles.

Thermal imaging systems can be used in conjunction with infrared landing and searchlights. These tend to be most effective at low levels of illumination. If there are external lights then pilots tend to limit their scan to within the area directly covered by the searchlight. They have to be trained to expand their search on either side of the beam. Brownout can also occur when there are reflections from an infrared searchlight caused by the dust that is raised in a rotor wash. The heat emitted by infrared searchlights can help enemy personnel who may themselves be using night vision equipment. As with image intensification systems, individuals can quickly become fatigued through prolonged use of these devices. A recent Lessons Learned

review was conducted into the initial deployment of light armored vehicles. One of four main findings was that "Long periods of using thermal optics can lead to crew fatigue…this can be overcome by having the dismounts trained on the functions of the turret" (New Zealand Army, 2003).

	FY95	FY96
Day	7.59	7.69
Night	9.72	13.87
Night unaided	6.37	9.31
Night aided	11.28	15.80
Night systems	17.15	22.54
Night goggles	11.97	14.37
Total	8.09	9.14

Table 1. **Class A-C Rotary-wing Accidents per 100,000 flying hours**

3. STATISTICAL STUDIES OF NVD MISHAPS

Table 1 presents the results of a study by the US Army Safety Centre into the accident rate for various forms of night operation involving rotary winged aircraft. As can be seen, there is a lower accident rate for flights involving direct 'unaided' visual observations than there is for flights with this equipment. Such a counter-intuitive finding can be explained in a number of ways. It might be that the use of night vision equipment impairs situation awareness, distracts from the use of other information systems and hence increases the likelihood of an adverse event. Equally, it might be argued that these devices tend to be used under adverse meteorological and environmental conditions when accidents are more likely to occur anyway. These different hypotheses illustrate the problems involved in moving from simple correlations to more detailed causal explanations. For instance, the US Army's Black Hawk helicopter fleet has suffered more than 20 fatal accidents in its 27 year service history. Approximately half of these occurred while pilots were wearing night vision devices (Hess, 2002). However, the fact that an accident occurred while the crew were using this equipment does not imply that the incident was caused by these devices. It can be very difficult to assess the role that particular technologies play in an adverse event. This is especially problematic when crewmembers may have suffered psychological or physiological trauma. They may be unable or unwilling to discuss the details of their actions in the aftermath of an

accident or near-miss incident. Further problems arise because these statistical studies do not consider those accidents under direct visual conditions that could have been avoided if the crew had been provided with night vision equipment.

Some attempts have been made to conduct a more detailed analysis of the accident statistics. For instance, Ruffner, Piccione and Woodward (1997) identified 160 US army accidents that were related to the use of night vision devices in ground vehicles between 1986-1996. Over two-thirds were attributable to three categories of terrain and roadway hazards: drop-offs greater than three feet (34%), ditches of three feet or less (23%) and rear collisions with another vehicle (11%). 34% involved the High Mobility Multipurpose Wheeled Vehicle (HMMWV), 18% involved the M1 Abrams Tank and 14% involved the M2/M3 Bradley Fighting Vehicle. The most commonly occurring environmental conditions that included dust (24%), blooming from light source (9%) and smoke (8%). Braithwaite, Douglass, Durnford and Lucas (1998) conducted a similar study of aviation accidents that focused on spatial disorientation caused by the use of night vision devices in helicopter operations. They argued that the various limitations of night vision devices, including the issues of depth perception and orientation mentioned in previous pages, predispose aircrew to 'spatial disorientation'. They found that approximately 43% of all spatial disorientation mishaps occurred during flights that used night vision equipment. Only 13% of accidents that did not involve spatial disorientation involved these devices. An examination of the spatial disorientation accident rates per 100,000 flying hours revealed a significant difference between the rate for day flying and the rate for flight using night vision devices. They concluded that the use of night vision devices increased the risk of a spatial disorientation accident by almost five times.

4. LACK OF NVD LEADING TO MISHAPS

It is often argued that the provision of night vision devices would have prevented many accidents. Such counterfactual arguments can be illustrated by the loss of a US Marine KC-130. The aircraft crashed into a Pakistan hillside near Shamsi airfield. There were no approach lights or navigational aids. The KC-130 was not equipped with any night vision equipment. Helicopter operations and noise restrictions prevented the crew from using their preferred approach. However, other KC-130s had landed at the same airfield without problems. The crew was experienced and rested. They had all flown into the airfield before. The official report concluded that the

crew had "stopped navigating with instruments" and relied on direct visual observations during their approach (Durrett, 2002). Several analysts, therefore, argued that night vision equipment would have helped to avoid the accident because direct visual observations had failed to identify the hazards (Vogel, 2002). After the crash, the Marines began to retrofit KC-130s with night-vision equipment and a GPS linked map-based navigation system. The official report insisted that while the provision of night vision equipment would have helped the crew, it would not necessarily have prevented the accident (Durrett, 2002). .

The problems of using accident information to analyze the strengths and weaknesses of night vision technology can also be illustrated by litigation following a land-based training accident (Maryland Court of Appeals, 1999). A US Army Major was run over by a truck driven by 2 Maryland Army National Guardsmen during a training exercise. The Major belonged to an active duty unit that was evaluating the exercise. The accident occurred just after midnight, when the two guards drove their truck along a dirt road to pick up a patrol. The Major had remained seated in the roadway after he had finished evaluating another exercise. He made no apparent effort to move as the truck approached. The vehicle was driving under "blackout conditions" without headlights. Although one of the drivers had a set of night vision goggles, he was not using them. Neither soldier had received any training in their use. Neither saw the Major who suffered serious injuries that were exacerbated by a series of delays in his evacuation. He was transported to the wrong hospital and was eventually declared dead on arrival at the intended destination.

The National Guard determined that the Major's death was caused by his lack of situation awareness during night vehicle maneuvers. They argued that if the Major had been alert, he would have heard the truck. The accident was also blamed on resource limitations that prevented the National Guard from training troops to use night vision equipment. In contrast, the Army rejected lack of funding and training as reasons for the drivers not using night vision goggles. The accident was caused more by the driver's excess speed than the Major's inattention. The Major's widow sued the State and the Maryland National Guard for maintaining insufficient supplies of night vision goggles and for failing to provide training to the drivers in the use of night vision goggles. Maryland's Court of Appeals unanimously upheld a Montgomery County Circuit Court decision to reject the $6 million lawsuit. This ruling illustrates the difficulty of using previous accidents to justify the introduction of night vision equipment. The judges' decision hinged on whether the court had jurisdiction over National Guard operational matters, including the provision of particular items of equipment. To establish

negligence it was argued that a jury would have to decide how many night vision goggles should have been acquired. The jury might also have to consider how such vision equipment should have been allocated, what kind of training should have been provided and when it should have been offered etc (Maryland Court of Appeals, 1999).

4.1 Night Vision Devices Contribute to Accidents

In contrast to those mishaps that might have been prevented by night vision equipment, many mishaps directly stem from the provision of these devices. For example, existing night vision currency requirements in the US Army's Aircrew Training Manual state that aviators must fly at least one hour using night vision equipment every 45 days. A recent incident demonstrated that the minimum requirement is insufficient for many missions. A UH-60L instructor pilot had over 8,000 hours of rotary-wing experience. All the crewmembers had flown together many times in the past. Both pilots were qualified and current for the night vision goggle training mission. However, they both averaged less than 3 hours of night vision flight per month over the preceding 7 months. The Army Safety Centre (2003) report argued, "If any one of the conditions — low recent experience, dust, winds, or low illumination — had not been present, perhaps the accident would not have occurred. If the aircrew had more recent experience, they would have been better able to deal with the harsh environment. If the illumination had been better, their low recent experience might not have been a factor. If the conditions had not been as dusty, perhaps the crew would not have become disoriented". This illustrates how a number of adverse factors can combine to create the conditions in which an incident occurs. In other words, the use of night vision equipment plays a necessary but insufficient role in the accident. Sufficient conditions often exist when personnel rely on these devices in extremely hazardous environmental or meteorological conditions.

The complex nature of many night vision incidents can also be illustrated by an adverse event involving an officer with a motorized rifle platoon (US Army Centre for Lessons Learned, 2001). His unit was to occupy a battle position during a training exercise using an M551A1 Sheridan light tank. The officer's platoon was to move from their hiding positions to occupy prepared fighting positions. His orders included information about the safety requirements associated with zero illumination operations. The officer also had access to a compass, a map and a GPS receiver to assist with nighttime navigation. Although the officer was relatively unfamiliar with the area, the gunner had several years of experience on this range. Even so, they spent a number of hours driving around looking for their battle position. Standard

operating procedures stated that the gunner should have dismounted to guide the driver when traveling cross-country in zero illumination. Instead, the officer used night vision goggles while his driver used a night sight. When they failed to find their fighting position, the officer was told to wait until first light before continuing the search. He carried on looking until the vehicle eventually overturned in the excavation. The officer was standing in the nametag defilade position and received fatal crush injuries. The Army Safety Centre argued that the crew relied too much on their night vision equipment as they searched for their battle positions. Soldiers must gain "an understanding and appreciation of the risk-management process and know that if the risks outweigh the benefits, then the mission should be a no-go" (US Army Centre for Lessons Learned, 2001).

4.2 Risk management

Risk management is the process of identifying and controlling hazards. The introduction of night vision technology can reduce the likelihood of some accidents whilst at the same time increasing the risks associated with other types of adverse event. Personnel are likely to conduct operations that would not have been attempted without the technology and which in retrospect ought not to have been attempted even with this additional support. Other risks stem from the limitations of the technology; these include visual illusions and the problems associated with environmental hazards. It is difficult to survey the risk 'landscape' in which night vision increases the likelihood of some hazards and diminishes the likelihood of others. For example, peacekeeping operations often present senior staff with complex decisions in which the use of night vision equipment forms part of a much wider set of concerns (Johnson, 2002). For example, the Canadian force in Somalia was involved in an incident that killed one Somali and wounded another (Canadian Department of National Defence, 1997). It was a turning point in Canadian involvement and forced significant changes in their rules of engagement. A Reconnaissance Platoon observed two Somalis walking around the wire of the Canadian Engineer's compound. The detachments had overlapping arcs of observation and fire. Infrared chemical lights were used to mark their positions in a way that was visible through night vision equipment but invisible to the naked eye. It appears that the 2 men fled after being challenged. They were then were shot at from behind. One was immediately wounded and the other was subsequently shot dead by another part of the patrol. Night vision equipment only played a small part in this incident. The soldiers' interpretation of their rules of engagement and the leadership of the Reconnaissance Platoon were identified as primary

causes. However, the subsequent inquiry did examine the decision to use night vision equipment. It was argued that if the compound had been better illuminated with conventional lighting then local civilians, especially petty thieves, would have been less inclined to approach the installation. Shortly after the incident, the Engineers constructed a light tower. This was perceived to have significantly reduced the problem of petty theft. However, the shootings may also have had a deterrent effect. The key issue here is that additional lighting was not initially installed because it would have interfered with the use of night vision goggles. The risk of nighttime friendly fire incidents was perceived to be of paramount importance. The shooting showed that this underestimated the risks of using night vision equipment in close proximity to the local civilian population (Canadian Department of National Defence, 1997).

4.3 Night-Vision Accidents and Training

US Army driver training requirements cover the use of night vision equipment in AR 600-55. This is supported by training circulars such as TC 21-305-2 *Training Program For Night Vision Goggle Driving Operations* and FM 21-*305*. Support is provided through a range of courses designed for specific vehicles as well as more general training, including *TC 1-204 Night Flight Technique and Procedures.* Much of this material has been informed by the lessons of previous adverse events. For example, a series of accidents led to a reminder being issued across the US Army that bright lights from vehicle headlights and other sources will drive the goggles' gain down to the point that everything else in the field-of-view all but disappears. In addition, if the bright light exposure continues for 70 seconds (+30 seconds), the PVS-7s will turn off. Similarly, officers were reminded that the natural illumination provided by the moon is often critical for image intensification systems and so missions should be planned to take into account the 15 degrees per hour change in the height of the moon as it waxes and wanes (US Army Safety Center, 2003a). The US Army also operates systems for learning lessons about the use of night vision equipment within particular operational contexts. In particular the insights gained from Operations Desert Shield and Desert Storm together with rotations in Kuwait helped to develop training materials that were put to use in more recent conflicts (US Army Safety Center, 2003b). Desert operations in Iraq again illustrated the importance of integrating information obtained from night vision equipment with accurate data from GPS applications. In particular, operational experience reinforced the need for personnel to be trained to keep the lenses clean and the goggles stored safely when not in use. Sand and dust accounted for a higher than expected attrition rate for most units

with access to these devices. Pilots were accustomed to dry lakebeds and scrub in their National Training Centre but were less prepared for the impact of shifting sand dunes and extreme temperatures on night vision equipment. For instance, "the authorized airspeed for nap of the earth flight is 40 knots, but an aircraft flying in zero illumination at 25 feet in sand dunes should fly just ahead of effective transitional lift…Just keep in mind that at airspeeds below ETL, you may encounter rotor induced blowing sand" (US Army Safety Center, 2003b). Operation experience also identified a number of visual illusions with night vision equipment. These devices can provide an impression of a false horizon when light-colored areas of sand surround dark areas, especially when other environmental factors, including dust and haze, also obscure the horizon. Desert conditions often also lack the visual markers and reference points that support accurate height perception. Under such circumstances, ground lights can often be mistaken for the lights of other aircraft or even stars. Lack of features and relatively slow speeds can also persuade pilots that they have stopped moving even though the aircraft is actually moving forward. These illusions can be so persuasive that individuals will still fall prey to them even though they have been trained to recognize that they can occur. Greater attention has recently been paid to team and crew coordination as a potential barrier to incidents and accidents. For instance, the Army Safety Center's Southwest Asia Leaders' Safety Guide emphasizes the need to synchronize crew observations and communications in order to combat some of the problems created by these illusions. Guidance is provided on scanning responsibilities for pilots and non-rated crewmembers in different types of flight.

The provision of training does not always match up to the standards that are claimed in many official publications. For instance, one of the lessons learned during the Canadian deployment in Bosnia was that more ground forces need to be trained in a wider range of this equipment. One of the participants in this deployment observed that "personnel were unable to train on the variety of Night Vision Devices that were eventually made available to us in theatre… not having this equipment available prior to deployment meant that we had to utilize valuable time to train personnel on equipment that they should have been familiar with before they arrived". Some of the equipment that they were expected to use only arrived six weeks after their deployment. However, the units were able to overcome these limitations. The Post Action review found that this equipment helped dismounted patrols in the towns and villages. The technology provided local inhabitants with a "dramatic" example of their fighting capability. This was claimed to have deterred crime and established credibility (Canadian Army Centre for Lessons Learned, 2001).

We have not considered the problem of fratricide. Many friendly-fire incidents directly stem from the use of night vision devices. Brevity prevents a more sustained analysis of these adverse events. Many of the issues are similar to those that lead to more general mishaps.

5. CONCLUSIONS AND FURTHER WORK

This paper has looked beyond the advertising and hype that surrounds many night vision devices. Our analysis has shown the complex role that image intensification and thermal imaging plays in military accidents and incidents. Some investigators have argued that these devices were a primary cause of military mishaps. Conversely, it has also been argued that the availability of night vision equipment would have prevented other accidents from occurring. A key conclusion is that the successful introduction of these systems depends upon a range of supporting factors. These include complementary technologies, such as GPS systems. The supporting infrastructure also depends upon appropriate training. This should help users to familiarizing themselves with individual devices but must also consider the ways in which teams of soldiers interact to overcome the limitations of existing technology. Greater emphasis should also be placed on formal risk assessment before these devices are deployed in military operations[1].

Ruffner, Piccione and Woodward (1997) have shown that existing night vision training helps drivers to identify ditches and other road conditions. It does not, however, help them to identify those depressions and other hazards that they have shown to be the cause of most night vision accidents. The accidents and incidents identified in this paper have supported many of the criticisms put forward by Ruffner et al. Several of the coalition partners in the Gulf were forced to use accelerated procurement to ensure that sufficient devices were made available to troops prior to the conflict. The UK Ministry of Defense (2003) issued an Urgent Operations Requirement action. Further work is required to determine whether this successful acquisition shortly before the conflict led to accelerate training procedures and whether this, in turn, led to the accidents and incidents predicted by Ruffner and his colleagues.

[1] This work was partly funded by EC RTN ADVISES (CT 2002-00288).

REFERENCES

M.G. Braithwaite, P.K. Douglass, S.J. Durnford and G. Lucas, The hazard of spatial disorientation during helicopter flight using night vision devices. Journal of Aviation and Space Environmental Medicine, (69)11:103844, 1998.

Canadian Air Force, A Dark and Stormy Night, Flight Comment, No 2, pp 6-7, Spring, 2002.

Canadian Army, Armour School Master Lesson Plan, Armored Reconnaissance Specialist Course: Observation, 2004.

Canadian Army Centre for Lessons Learned, Night Vision in Kosovo, The Bulletin, (8)1:6-11, April 2001.

Canadian Dept of National Defence, The Somalia Inquiry Report; Chap 5 March 5th Incident, 1997. http://www.forces.gc.ca/site/Reports/somalia/vol5/V5C38B_e.asp

W.D. Durrett, Report into the Loss of a KC-130 at Shamsi Pakestan, January 9th 2002, US Marine Corps, San Diego, 2002

G.J. Gilmore, 'We Don't Need to Lose People' to Accidents, DoD Personnel Chief Asserts, US Department of Defence, DefenseLink, June 2003.

P. Hess, Army Identifies Soldiers Killed in Crash, UPI, December 2002.
http://www.upi.com/view.cfm?StoryID=20021213-124412-7962r

C.W. Johnson, Risk and Decision Making in Military Accident Reporting Systems. In L. Johnson (ed.) Proceedings of Human Factors 2002, Melbourne, Australia, 2002.

C.W. Johnson, (2003). Handbook of Incident Reporting, Glasgow University Press, Glasgow, Scotland.

Maryland Court of Appeals, The Estate of Andrew Burris, et al. v. The State of Maryland, et al. No. 130, Sept. Term, 1999. Opinion by Wilner, J.

New Zealand Army, Lessons Learned from Initial Deployment of the Light Armored Vehicle (LAVIII), LAV Update Number 3, August 2003.

J. W. Ruffner, D. Piccione and K. Woodward, Development of a night driving simulator concept for night vision image intensification device training. In Proc of Enhanced and Synthetic Vision Conference, SPIE 11th International Symposium on Aerospace/Defense Sensing, Simulation, and Controls, Orlando, Vol 3088. PP. 190-197, 1997.

J.W. Ruffner, J. D., Antonio, D.Q. Joralmon and E. Martin, Night vision goggle training technologies and situational awareness. *Proc of Advanced Technology Electronic Defense System Conference/Tactical Situational Awareness Symposium,* San Diego, CA. 2004.

G.J. Salazar and V.B Nakagawara, Night Vision Goggles in Civilian Helicopter Operations, Federal Air Surgeon's Medical Bulletin, Fall 1999.

US Army Centre for Lessons Learned, An M551A1 in the Wrong Hands, Countermeasure, Volume 29, Number 2, February 2001,

US Army Centre for Lessons Learned, NVG Currency, A Perishable Skill — Currency is Not Proficiency, Flight Fax, Vol. 31, Number 2, February 2003.

US Army Centre for Lessons Learned. Fight at Night and Survive, Countermeasure Vol 24. Number 4, April 2003a.

US Army Centre for Lessons Learned, Night Vision Goggles Desert Operations Lessons Learned - 13 Years in the Making, Flight Fax, Vol. 31, Number 4, April 2003b.

US Army Safety Centre, U.S. Army Accident Information, Aviation Accident Statistics for the Current Fiscal Year, As of 19 January 2004.

US Department of the Army, Aeromedical Training for Flight Personnel, Washington, DC, 29 September 2000, Field Manual 2-04-301 (1-301)

S. Vogel, Marine KC-130 That Hit Mountain Had No Night Vision, Washington Post, Sunday, February 17, 2002; Page A17.

THE GLOBAL AVIATION INFORMATION NETWORK (GAIN)

Using Information to Make the Aviation System Less Error Prone and More Error Tolerant

Christopher A. Hart
U.S. Federal Aviation Administration

Abstract: The worldwide commercial aviation system is a complex system involving hardware, software, and liveware (humans). All of these components must work together efficiently and effectively in a variety of environments in order for the system to function successfully. One of the least predictable aspects of how the system operates is what the humans will do. In the aviation system, much of this lack of predictability results from inadvertent error and/or operators of the system trying to optimize the functioning of the system in unanticipated situations. When undesirable consequences result from the inadvertent error and/or well-intentioned efforts to make the system work better, the human action is usually classified as "human error." As the aviation system becomes more complex, safety professionals are concluding that responding successfully to "human error" necessitates increased focus on the system. Focusing primarily upon the individual who committed the "error" (a) assumes, sometimes incorrectly, that the most effective remedy is getting the individual to behave differently, and (b) fails to consider the role of the system in leading to the undesired behavior. An essential element for enhanced system focus is better information. Rapid advances in information technologies are creating unprecedented opportunities for safety professionals to collect better information about how the operators of the system make it work. That information helps safety professionals improve the system by making it (a) less likely to result in human error, i.e., less error prone; and (b) more capable of withstanding human error without catastrophic result, i.e., more error tolerant. The Global Aviation Information Network (GAIN) is promoting and facilitating the voluntary collection, analysis, and sharing of information in the international aviation community to improve safety. GAIN was proposed by the U.S. Federal Aviation Administration (FAA), but it has evolved into an international coalition of aviation community members – airlines, manufacturers, unions, and governments. GAIN is helping to create legal and cultural environments that encourage and facilitate the collection of

large quantities of data. GAIN is also creating tools and processes to help aviation safety professionals convert that data into useful information to (a) identify potential safety issues, (b) prioritize them, (c) develop solutions, and (d) evaluate whether the solutions are working. Two aspects of GAIN that have been discovered from experience are significantly enhancing its development. First, the tools and processes can be used not only in other transportation modes, but also in other industries, including chemical manufacturing, nuclear power, public utilities, health care and national security. Second, experience is demonstrating that the systematic collection and sharing of safety information can not only facilitate the correction of troublesome trends, but can also result in significant immediate cost savings in operations and maintenance. In theory, other industries applying these tools and processes should also be able to reap significant immediate economic benefits. Extensive information about GAIN is on the Internet at www.gainweb.org

Key words: Aviation safety, mishap prevention, proactive information programs, human error, error prone, error tolerant, data collection, data analysis, data sharing

1. ROOT CAUSES OF HUMAN ERROR IN COMPLEX SYSTEMS

Most of the worldwide commercial aviation community workforce is highly trained, competent, experienced, and proud of making the system work well. Despite these efforts to make the system work, however (along with numerous other activities to improve safety), mishaps occur – albeit at a commendably low rate. Inasmuch as the workforce is proud, competent, and trying to make the system work, why do they nonetheless make errors that can be harmful, even (in the case of pilots) to themselves?

The commercial aviation system consists of a complex array of ever-changing, interdependent, tightly coupled components, all of which must work together efficiently and effectively in order for the system to function successfully. The complexity of the system has been increasing over the years, and most experts expect even more complexity in the future. The increasing complexity of the system engenders human error in three ways.[2]

First, more complexity increases the difficulty of designing human error out of a system, even when it is operated by a competent, highly trained, experienced workforce. Designing a component of a system to be "error

[2] These three are in addition to other factors that can exacerbate human error, irrespective of whether complexity is increasing, e.g., pressures to accomplish more with less.

proof" is challenging enough, but making it error proof in a dynamic, tightly coupled, interdependent environment is considerably more challenging.

Second, more complexity increases the likelihood that the operator will face situations that the operator, and possibly even the designer, did not anticipate. In a complex, tightly coupled, dynamic system, it is very difficult for component designers to foresee all of the circumstances or environments in which the components will be operated.

Third, more complexity increases the likelihood that the operator will encounter situations in which responding "according to the book" would not, in the perception of the operator, make the system work best. Consequently, competent, highly trained, experienced operators who are trying to make the system work better may not respond "according to the book."

Sometimes the actions of the operators in these three categories – inadvertent error, unanticipated situations, and non-optimal operating instructions – lead to desirable results, and sometimes they do not. If the results are undesirable, the actions are generally classified as "human error."

The "human error" categorization is literally accurate because the "error" was performed by a human. In the case of unanticipated situations and non-optimal operating instructions, however, and sometimes even in the case of inadvertent error, the description is unduly pejorative in suggesting that the person did something "wrong." If other people similarly situated would have taken the same action under the circumstances, as is often the case, query how accurate or helpful it is to label the action as "error."

For example, if people trip over a step "x" times out of a thousand, how big must "x" be before we stop blaming the person and start focusing more attention on the step, e.g., should it be painted, lighted, or ramped, or should a warning sign be posted? Blaming the problem on "human error," even if literally accurate, (a) fails to prevent recurrences of the problem, and (b) exacerbates the problem because the negative implication of "error" discourages people from reporting the problem to those who can fix it.

2. THE NEED FOR INCREASED SYSTEM FOCUS

When people are trying to make an increasingly complex system work well but still make errors, including errors that can hurt themselves, our historic primary focus on the individual is no longer sufficient. Instead of focusing primarily upon the operator, e.g., with regulation, punishment, or training, we must probe further to find out why the operator did or did not take a certain action. Determining "why" requires focusing more attention on the system in which the operators are operating. Because human error cannot be eliminated, the challenge of this increased system focus is how to

make the system (a) less likely to create conditions that could result in human error, i.e., less error prone; and (b) more capable of withstanding such errors without catastrophic result, i.e., more error tolerant.[3]

Responding to human error by making the system less error prone and more error tolerant does not mean *reducing* the safety accountability of the system's operators. To the contrary, it means *increasing* the safety accountability of the people who design, build, and maintain the system.

An example of the need to expand to more of a system focus is a 1974 accident on an approach to Dulles International Airport (Washington, D.C.) (Aircraft Accident Report 1975). The pilots were following the published instructions for navigating to the runway (known as the "approach chart"), but they were confused by the chart and the air traffic controller instructions, and they descended too soon and hit a ridge. The accident hearing revealed that other pilots had experienced the same confusion but did not crash because, unlike on the day of the crash, the ridge was not obscured by the clouds.

In this accident, the pilots made the final error, but effectively preventing recurrences necessitates going far beyond merely warning pilots to be more careful. Among other things, the remedies include correcting the confusing approach chart, revising pilot/controller communications protocols, installing more sophisticated navigation equipment at airports, installing terrain alerting software in air traffic control radar systems, and installing terrain alerting equipment in airplanes.

One of the most tragic aspects of this accident is that pilots from one airline reported the approach chart confusion to their management – which was unusual in those days – and management distributed warnings to their pilots; but the crash involved a different airline. Thus, this accident is cited here, despite its age, because is an example of a problem that exists to this day – inadequate collection and sharing of information in the worldwide aviation community about potential safety problems in the system.

In health care, the U.S. Institute of Medicine (IOM) issued a report about the need to expand beyond operator-focused remedies to system-focused remedies (U.S. Institute of Medicine 2001). Noting a concern that 44,000 -

[3] Improvements that increase error tolerance may facilitate additional "corner cutting" of safety margins. Query, for example, whether new in-cockpit systems that show the pilot the location of higher terrain and other airplanes may encourage illegal flight in clouds. Conversely, some improvements may reduce error tolerance. For example, improvements that allow aircraft to reach higher altitudes reduce the tolerance for cabin pressurization system error because of the longer time needed for descent to an oxygen-safe altitude in the event of failure. If the system risks are affected by new technologies, system safety principles call for a review of the hazards and the acceptability of the associated risks.

98,000 people die each year from medical mistakes,[4] the IOM proposed the systematic collection and analysis of information about "near-miss" mistakes – mishap precursors – in order to learn more about how to identify them and develop remedies. They recommended a proactive information approach because:

> Preventing errors means designing the health care system at all levels to make it safer. Building safety into processes of care is a much more effective way to reduce errors than blaming individuals . . . . The focus must shift from blaming individuals for past errors to a focus on preventing future errors by designing safety into the system. . . . [W]hen an error occurs, blaming an individual does little to make the system safer and prevent someone else from committing the same error.[5]

Improving the system is not trivial because, to its credit, most commercial aviation systems enjoy robust backups, redundancies, and safeguards, and mishaps rarely result from a single problem. Usually several things must go wrong, as "links in the accident chain," for a mishap to occur. However, the absence of any single weak point means that there is no single easily identifiable point to intervene with a remedy. A Boeing study reveals accident chains with as many as twenty links, each of which is an event that, with a different outcome, could have interrupted the accident chain (Boeing Commercial Airplane Group 1995),

This scenario can be represented by a box containing several spinning disks with holes, with a light shining into the box (Figure 1). The disks are defenses against mishaps, and the holes are breaches in the defenses. Each breach may occur without harmful result; but when the links combine in the wrong way – when the holes in the disks line up – the light emerges from the box and a mishap occurs. This borrows from the Swiss cheese analogy created by Prof. James Reason of Manchester University in the United Kingdom – a mishap occurs when the holes in a stack of cheese slices line up (Reason 1990 p. 208). The spinning disks portray the dynamic and interactive nature of the aviation system that is not as apparent with cheese.

[4] As suggested by this large range of estimates (more than a factor of two), the actual number of fatalities is not known and is a matter of considerable controversy.

[5] Id., at pp. 4-5.

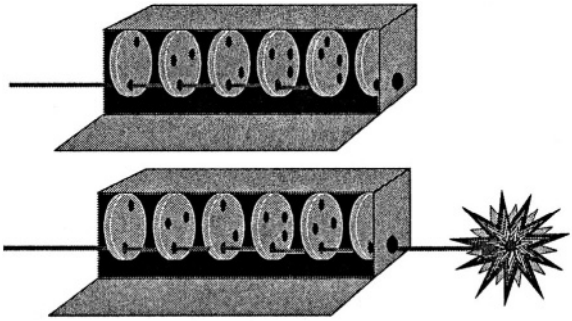

Figure 1. The Spinning Disks

Following the analogy created by Prof. Reason, the disks toward the right end of the box relate to "active" problems, e.g., the pilot's confusion about the approach chart. The disks toward the left end relate to more "latent" problems that may infect the system for years before they manifest themselves in a mishap, e.g., inadequate management focus on safety.

Many of the disks involve various parts of the system interacting with each other. Nonetheless, accident investigations have frequently placed the cause upon the person who made the "final" mistake – most often the pilots. This placement of causation ignores the fact that the person who made the final mistake probably had little or no control over most of the spinning disks to the left of the last disk, those that interacted to help create a scenario for a mishap. Fixing only the last disk unduly focuses on the individual who happened to be in the wrong place at the wrong time. In order to be more proactive, we will have to focus more on the entire system, which involves addressing all of the disks.

3. OBTAINING BETTER INFORMATION

Because of the robustness of the defenses against mishaps, the aviation community mishap scenario can be depicted by the Heinrich Pyramid (Figure 2).[6] The Heinrich Pyramid shows that for every fatal accident, there will be 3-5 non-fatal accidents, 10-15 incidents, and *hundreds* of unreported occurrences (the exact ratios vary with the nature of the endeavor).

[6] Heinrich, H.W., Industrial Accident Prevention (First Edition, McGraw Hill, 1931)

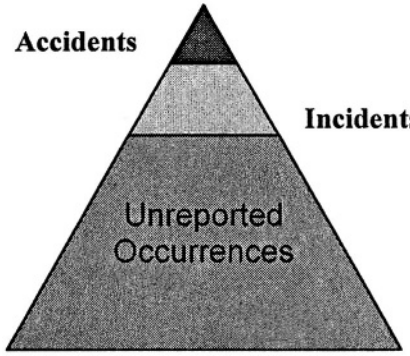

Figure 2. The Heinrich Pyramid

Usually these occurrences were not reported because each one, alone, was innocuous and did not result in a mishap. Today's unreported occurrences, however, are the "building blocks" of tomorrow's mishaps. When they happen to combine with other unreported occurrence "building blocks," they may someday result in a mishap.

In response to this situation, many industries are developing processes to collect and analyze information about precursors before they result in mishaps. All too often, the "hands-on" people on the "front lines" reveal, *after* a mishap, that, "We all knew about that problem." The challenge is to get the information that "we all knew about" and do something about it *before* it results in mishaps.

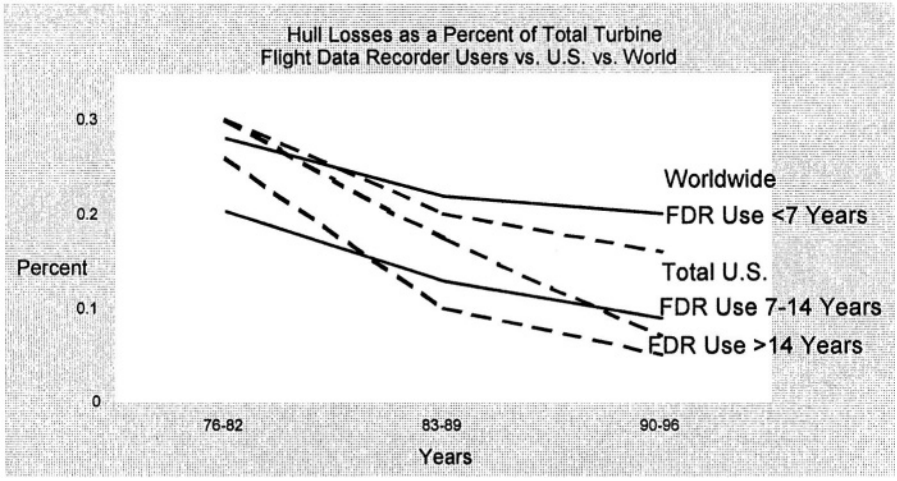

Figure 3. Effectiveness of FDR Use (Source: Total U.S. – FAA NASDAC; Other – Skandia Insurance Co., Ltd.)

In the aviation community, reporting about events near the top of the pyramid is usually mandatory, and reporting about events in the larger part of the pyramid is usually voluntary. Although mandating reporting may increase the amount of information collected, there is no reasonable way to mandate the reporting of occurrences – such as a misunderstood approach chart, as discussed above – that do not rise to the level of mishaps or potential regulatory violations. Instead, short of a mishap, the system will generally have to rely upon *voluntary* reporting, mostly from front-line workers, to learn about these types of problems. Voluntary reporting will not occur, however, unless legal and cultural barriers that deter such reporting are addressed.

1. Legal Concerns. In most countries, most or all of the following four legal concerns discourage the development of systems that would enable and encourage front-line workers – whose voluntary reporting is most important – to come forth with information.

First, potential information providers may be concerned that company management and/or regulatory authorities might use the information for punitive and/or enforcement purposes. Thus, a worker might be reluctant to report about a problem that resulted from a confusing process, fearing that management and/or the government might disagree that the process is confusing (especially if management and/or the government created the process), and punish the worker instead.

A second concern in some countries is that reporting potential problems to government regulatory agencies may result in public access to the information (including media access), and such access could be embarrassing, bad for business, or worse.

A third concern is potential criminal prosecution, and a fourth concern is that collected information may be used against the source in civil litigation.

With help from GAIN, excellent progress has been made in the U.S. on these issues, following examples set years ago by the U.K. Civil Aviation Authority. In addition, GAIN is working through the International Civil Aviation Organization (ICAO), the aviation arm of the United Nations, to get all of its 188 member countries to review their legal and regulatory structures and make modifications as needed. As a result, ICAO has taken several actions that are helping to address these legal issues worldwide.

2. Cultural Issues. Although aviation community leaders often pronounce that safety is their most important goal, the most important goal for most hands-on workers – for the advancement of their careers – is to satisfy their immediate supervisor. More often than not, however, the supervisor's career future depends upon satisfying *production, capacity, and/or efficiency* goals. If a safety concern from the hands-on workers may

undercut any of these supervisor goals, the supervisor may implicitly or explicitly discourage the reporting of potential safety concerns. Thus, one of the most significant cultural barriers is the tension that sometimes occurs between safety goals and the production, capacity, and/or efficiency goals. This is a potential problem in *all* types of aviation community entities, including airlines, manufacturers, air traffic control organizations, and government regulators.

As a result, even if the head of the organization and the hands-on workers agree that safety is important, the organization's culture will not encourage the reporting of potential safety issues by the hands-on workers unless safety is one of the supervisor's job requirements.

3. Improved Analytical Tools. Once the legal and cultural issues are addressed enough to facilitate more systematic collection of potential precursor information, the aviation community will face another major obstacle – the need for more sophisticated analytical tools to convert large quantities of data into useful information, i.e., to "separate sparse quantities of gold from large quantities of gravel." These tools will not solve problems automatically, but they must generally be able to help experienced safety professionals (a) identify potential safety issues, (b) prioritize them, (c) develop solutions, and (d) determine whether the solutions are having the desired outcome without creating any undesired effects. Tools will be needed for both digital data and textual data.

In the course of identifying and resolving concerns, with the help of more data and better analytical tools, the aviation community will have to respond in a way that is significantly different than how it has responded in the past. First and foremost, as discussed above, will be the need to expand beyond operator-focused remedies – such as blaming, punishing, and re-training – to system-focused remedies.

As safety professionals focus more on improving the system, they will need to incorporate the following two concepts into the analytical mix.

- System-Wide Interventions. First, improvements to the system have frequently related to individual components of the system. However, because the components of the system are tightly coupled and interdependent, as noted above, safety professionals will have to become better at addressing problems on a system-wide basis, not only on a component-centric level. Existing safety risk management methods are flexible enough to be applied at every level – from sub-

component to system-wide – but the aviation community does not yet have much experience applying them at system-wide levels.[7]

- Human Factors. Second, designers must learn more about creating systems and processes that account appropriately for the human factors involved. Many industries, including aviation, are studying human factors issues to varying degrees, but most are still early on the learning curve.

4. The Importance of Sharing. The collection and analysis of information can result in benefits even if the information is not shared, but the benefits increase significantly if the information is shared – not only laterally, among competing members in the aviation community, but also between various components of the community. Sharing makes the whole much greater than the sum of the parts because it allows the entire community to benefit from the experiences of every member.

Thus, if any member of the community experiences a problem and fixes it, other members can address the problem proactively, before encountering it themselves. Moreover, problems that appear to be isolated instances can much more quickly be identified as system trends of importance when the information is shared among members of the community.

The benefits of sharing, in turn, increase the importance of more sophisticated analytical tools because there is little need, desire, or capability to share raw data, except "virtual" sharing, as discussed below. What will usually be shared is *analyzed* data, or information.[8] Thus, meaningful sharing will probably not occur until data are converted by analytical tools into useful information.

"Virtual" sharing is the electronic sharing of data without the data leaving the owner. Thus, if an airline wanted to know if another airline had encountered a certain problem, it could seek permission of other airlines to apply its search tools to their databases. Database owners would always control who could search for what in their databases, and they could give different levels of permission to different users.

Both types of sharing are facilitated by the network infrastructure that GAIN has proposed, as discussed below. In order for this concept to work, however, industry, labor, and governments must work together to encourage (a) the establishment of more programs to collect and analyze information,

[7] Experience has shown that analysis of individual entity data is best conducted, in the first instance, by the entity itself because it understands the context in which the data were created. Yet to be determined is how system-wide data will be collected and analyzed.

[8] The shared information will also be de-identified because the benefit of sharing information about precursors usually outweighs any need to identify the source.

and (b) more systematic sharing of information. Governments must help facilitate collection and sharing by assuring that their laws, regulations, and policies do not discourage such activities, and by funding research to develop better analytical tools for using large quantities of data effectively.

Last, but not least, meaningful sharing requires *trust.* Because industry, labor, and governments must work effectively together in order for the aviation system to work, they must realize that blaming each other when something goes wrong is tantamount to saying that, "Your end of the ship is sinking." In order to make a safe system even safer, industry, labor, and governments must learn to trust each other and work better together to develop system solutions for system problems.

4. THE GAIN CONCEPT

In order to accomplish this information collection, analysis, and sharing to learn about the potential individual links in an accident chain, the FAA proposed the Global Aviation Information Network (GAIN). GAIN was proposed by the FAA to be a privately owned and operated worldwide information infrastructure,[9] and as hoped, it has evolved into an international coalition of aviation community members – airlines, manufacturers, unions, and governments.

GAIN is helping to promote and facilitate the voluntary collection, analysis, and sharing of safety information in the international aviation community in two ways. First, GAIN is helping to create legal and cultural environments that encourage and facilitate the collection of large quantities of data. Second, GAIN is creating tools and processes to help aviation safety professionals convert that data into useful information. With a voluntary, privately owned and operated global network of data collection and exchange systems – thus the word "Network" in the name – government, industry, and labor can cooperate with each other, to their mutual benefit, to make a safe system safer (Figure 4).

[9] If this proactive information concept reduces costs and helps to improve safety, as expected, then the aviation community will *want* to own it, and the savings will create a strong incentive to improve safety. Thus, private ownership would operate GAIN far more efficiently and effectively than a government agency because – without criticizing any government agency – private industry has both (a) greater ability to respond quickly and precisely to issues that arise, and (b) more direct economic incentive to do so. As ancillary benefits, private ownership of GAIN would help alleviate concerns that GAIN is a guise for gathering information to be used by regulatory agencies for enforcement, as well as concerns about public access to sensitive data.

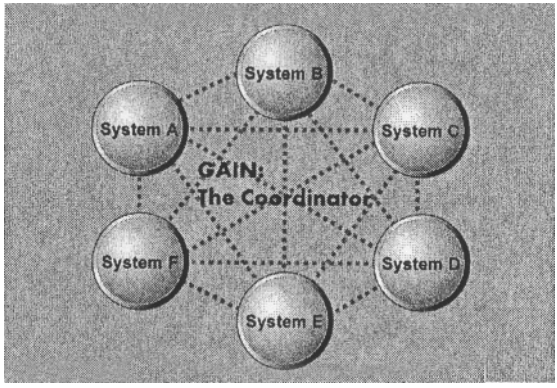

Figure 4. GAIN as a Network of Systems

Although proactive aviation information activities have been underway in some countries for years, the FAA proposed GAIN in an effort to bring many of these programs together into a more unified and systematic international network. Among the world leaders in this endeavor are the U.K. and some of its airlines, where flight data recorders have been routinely accessed as a source of valuable information for several decades.[10]

In addition, in 1996, the French Academie Nationale de L'Air et de L'Espace published a document entitled "Feedback From Experience in Civil Transport Aviation" that recommended a proposal to collect, analyze, and disseminate aviation safety information, which GAIN closely resembles. Some Scandinavian countries have been reading flight data recorders routinely for many years; Japan Airlines has had a proactive flight monitoring information program for several years; and the former Soviet Union had commenced various proactive safety information activities.

More systematic collection, analysis, and sharing of information can be a win-win for everyone involved. Private industry wins because of fewer mishaps. Labor wins because, instead of being the brunt of blame and punishment, front-line employees become a valuable source of information about potential problems and proposed solutions to accomplish what everyone wants – fewer mishaps. This presents a significant opportunity to change the relationship between labor and management from adversarial – blaming each other when things go wrong – to partners who are working together to improve safety. Government regulators win because the more they understand the problems, the more precise they can be about proposing

[10] One of the most widely used flight data analysis software packages in the worldwide aviation community is BASIS, the British Airways Safety Information System.

remedies, which makes the remedies both more effective and more credible. This further benefits industry because improved effectiveness of remedies means greater "bang for the buck" on implementing the remedies. Last but not least, the public wins because of fewer mishaps.

5. POTENTIAL BENEFITS IN ADDITION TO SAFETY

As more aviation community members implement GAIN concepts, experience is demonstrating unforeseen potential for applicability to many other industries, and for generating significant immediate economic benefits.

1. Breadth of Applicability. As in commercial aviation, many industries and endeavors, including most transportation modes, chemical manufacturing, public utilities, nuclear power, and most notably, health care, have enjoyed a declining mishap rate for several years. Many of those industries, however, are recently finding that their mishap rate decline is becoming flatter. As they explore proactive information programs to identify mishap precursors and remedy them in an effort to resume the rate reduction, it is becoming apparent that many of the reasons for the flatter decline, as well as many of the solutions, are common to most or all of these industries. Accordingly, although one size does not fit all, the opportunity exists as never before for these industries to work together and exchange notes about problems and solutions, to the benefit of all involved.

Also potentially benefiting are national security and information infrastructure protection.

GAIN is actively exploring the opportunities with these and other industries in order to avoid "reinventing the same wheel."

2. Immediate Economic Benefits? Not yet clear is whether all of these industries will also enjoy a benefit that is becoming apparent in the aviation community. Airline safety professionals have sometimes encountered difficulty "selling" proactive information programs to their management because of the commendably low fatal accident rate in commercial aviation, combined with the impossibility of proving that an accident was prevented. Fortunately, the first few airlines that implemented proactive information programs quickly started reporting *immediate* and sometimes *major* savings in operations and maintenance costs as a result of information from their safety programs. It is not yet clear whether other industries, most notably health care, will enjoy such immediate savings from their information programs, but conceptually the likelihood seems high.

Immediacy of economic benefits, if demonstrated, could be a very significant development for mishap prevention programs, by converting them to immediate and sometimes major profit centers, rather than mere "motherhood and apple pie" good ideas with potential statistically likely future economic benefits.

6. CONCLUSION

As the worldwide aviation community becomes more complex and endeavors to improve an already commendable safety record, its most difficult challenge is addressing human error, i.e., making the system less error prone and more error tolerant. Rapid advances in information technologies are providing opportunities as never before to collect, analyze, and share information to further improve the safety of the aviation system. GAIN is assisting these efforts by (a) helping to create legal and cultural environments around the world in which proactive information collection and sharing activities can flourish, and (b) developing tools and processes to help the worldwide aviation community take advantage of the major technological advances in its ability to collect, analyze, and share safety information.

REFERENCES

Aircraft Accident Report (1975), Trans World Airlines, Inc. Boeing 727-231, N54328, Berryville, Virginia, December 1, 1974. (Report Number AAR-75-16). Washington, DC: U.S. National Transportation Safety Board.

Boeing Commercial Airplane Group (1995), *Accident Prevention Strategies,* Document D6-56978-98.

Reason J. (1990) *Human Error,* Cambridge University Press.

U.S. Institute of Medicine, (2001). *To Err is Human: Building a Safer Health System.*

DEVELOPMENT OF CRITIQUING SYSTEMS IN NETWORK ORGANIZATIONS

Ola Leifler[1], Björn Johansson[1], Mats Persson[2] & Georgios Rigas[2]
[1]*Linköping University, Sweden,* [2]*National Defence College, Sweden*

Abstract: Recently, network organizations have been suggested as a solution for future crisis management and warfare. This will, however, have consequences for the development of decision support and critiquing systems. This paper suggests that there are special conditions that need to be taken into account when providing the means for decision-making in networked organizations. Hence, three research problems are suggested that need to be investigated in order to develop useful critiquing systems for future command and control systems.

Key words: decision support, critiquing systems, crisis management

1. INTRODUCTION

The nature of decision support is bound to change with the new types of network-centric organizations for military command and control that are emerging. When responsibilities are delegated to a larger extent and organizations need to adapt more and more rapidly to changes in the environment, the need for rapid and well-informed decisions increases. There has also been a great increase in precision and strike force in the military sector (van Creveld, 2000), meaning that each decision is likely to have much greater consequences than earlier. Thus, at the same time that consequences and risks are increasing, more people are supposed to make decisions and take action in parallel. This is to be done under time pressure and without errors. This, in turn, calls for a new view of decision support as a coordination tool and as a tool for collaborative work, supporting decisions

at a number of organizational levels and in specialist functions, rather than the older expert-system, single decision-maker, view.

In 1988, Kraemer and King presented a survey of the development of group decision support systems in the United States (Kraemer and King, 1988). Although many items they described as decision support (shared databases, intranets, shared displays) are everyday technology now, the problems of building decision support systems for teams or groups seem to be as valid as they were 15 years ago (Kraemer and King, 1988, p. 369):

> The technical systems necessary to create effective GDSS tools for real-time decision-making are difficult to build, and the potential uses of such systems are not well specified. The most coherent specifications of the decision process are built on the rational model of decision making, which at best accounts for only a part of the true decision-making behavior that takes place in group decision making.

Although the field of decision-making clearly has made progress with fields like Dynamic Decision Making and Naturalistic Decision Making, their observations still hold, something that we will elaborate on later in this paper. Even if research has given a deepened understanding of the nature of decision making, there is an evident need for a collaborative or distributed view on decision making in the discussion of future decision support systems, otherwise we might end up with products based on assumptions that are no longer valid in contemporary work environments.

The authors of this paper are all researchers within a Swedish research project aiming at developing a mobile command and control concept for the future network-based defense (see Section 2). One important part of this project is to evaluate and develop a critiquing system (Sundin and Friman, 1998) for the staff. As a point of departure for this work, we will analyze the problems of current decision support technology in connection with team decision-making and networked organization structures.

2. ROLF 2010

Organizations such as the rescue services or military defense have traditionally been hierarchically structured. As such, the command and control function has been located in the uppermost hierarchical level. Today it is argued that current hierarchical organizations are too rigid to be able to act and react on situations in future and highly dynamic environments. As an option for handling the dynamics, so called “network organizations” have been proposed as a solution. The structure of network organizations is

considered a possible answer to several problems of exerting command and control among military communities (Alberts et al., 2000; Cebrowski and Garstka, 1998; Sundin and Friman, 2000).

Implementations of such an organizational structure imply that traditional hierarchical levels of command could be flattened out, reduced or even completely removed. Reducing levels of command is considered advantageous and necessary to shorten the reaction time to changes in the environment since data processing within every level of command is considered time consuming and thereby would seriously hamper any necessary action. Furthermore, it is assumed that if a decision-maker is provided with enough data presented in an understandable way he or she will be able to make "optimal" decisions. Consequently, larger amounts of data have to be handled by the commanders within network organizations in comparison with traditional hierarchical organizations.

In Sweden, a new command and control concept is currently under development. The project, known as ROLF 2010 (Joint Mobile Command and Control Function), is aimed at creating a staff environment where a team of decision-makers can work jointly (see Figure 1). With the aid of information technology in the form of shared workspaces and direct access to sensor information, it is envisioned that commanders will be able to make swift and correct decisions. However, greater demands will also be made on mission control centers such as the ROLF 2010 staff unit, since "[…] it will require that the commander and his staff will be able to handle greater amounts of information and greater complexity than before" (Sundin and Friman, 2000).

Naturally, a very important part of such a system is the tools used by the commanders in the decision process. In the ROLF-vision, a critiquing system, rather than a traditional decision support system, is a central part.

In a network-centric organization, analyzing large amounts of data for the decision maker is of course an important issue, but what is really new is the amount of coordination that the decision support systems have to be both aware of, and actively supporting.

By "being aware of" coordination, we refer to the fact that the internal model of the decision-making process, used by the decision-support system, has to take into account the concurrent actions taken by all human members that the system is supposed to interact with. That is, we cannot be satisfied with having a decision-support system that is only aware of the local actions performed by the human operator it is currently interacting with.

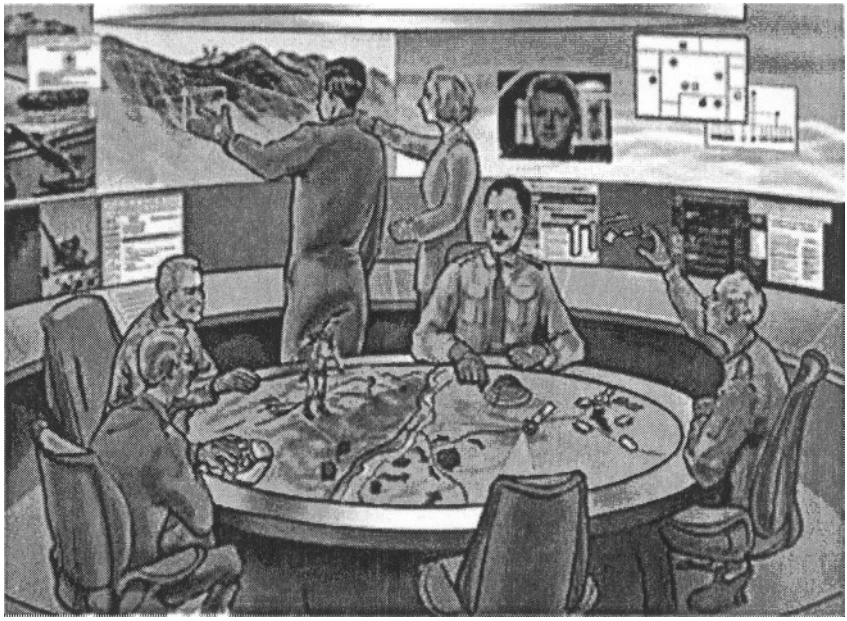

Figure 1. An artists view of the future ROLF 2010 work environment.

By "actively supporting" coordination, we refer to the fact that the system would have to provide more than only a means of communication, although such functionalities were traditionally regarded as decision support systems (see Section 1). We believe that analyzing user input and providing comments and feedback on such input would, in a network-centric organization, have to be done with *groups* of people providing the input and also receiving responses. This can be seen in contrast to traditional systems for intelligent decision support that have relied on a simple use case where a single user is provided with expert advice from a computer systems that analyzes a user-defined solution for a given problem, with respect to criteria defined by human domain experts.

3. INTELLIGENT DECISION SUPPORT

The idea of decision support is to aid a single or several decision-makers in situations where there is either too much, too incomprehensive information or the consequence of the decision is of such great importance that even trivial problems need some guidance to ensure a correct decision. Since humans are very good in their comprehension of situations but have a limited capability for analyzing, computers seem to be a suitable aid for that task.

Decision support can be manifested in a number of ways. First, a user could explicitly ask an expert system for a solution to a problem. This kind of decision support has been successful to some extent within areas where systems are "closed", meaning that the influences from external, unforeseen sources are minimal. Another kind of approach is the "critiquing" where the user presents a proposal for action, and is given feedback on that proposal. Such a system aims at three things (Silverman, 1992):

- Recognizing and analyzing human erroneous action;
- Giving persuasive feedback, and
- Adapting to the situation and previous experience.

Given that several individuals making decisions jointly will operate future command and control situations, critiquing systems emerge as being of great interest. As opposed to suggesting "solutions" to whatever problem is at hand for a group of decision-makers, critiquing could mean that the group members are notified of specific actions taken by others, actions that might in turn have an effect on their own work. This kind of feedback mechanism could, we believe, serve as a useful platform for constructing new kinds of intelligent decision support. However, what does it mean to provide "intelligent decision support" for a group of people collaborating on some task?

4. TEAMWORK

Irrespective of the general task at hand for the personnel exerting command and control in a ROLF 2010 like environment, the members typically operate as a team. To distinguish what constitutes and distinguishes a team from a group we have agreed upon Cannon-Bowers', Salas' and Converse's definition of a team as (Cannon-Bowers et al., 1993):

> [T]wo or more individuals who must interact cooperatively and adoptively in pursuit of shared valued objectives. Further, team members have clearly defined differentiated roles and responsibilities, hold task-relevant knowledge, and are interdependent.

The necessity for interdependency could be exemplified by Brannick and Prince's (Brannick and Prince, 1997) statement where they stress that one of the central features of teamwork is coordination where there is some kind of adjustment that the team members make in order to reach the goal.

Given that the team will carry out its task in an uncertain environment, the primary components of a decision-support system to develop are those which help to reduce that uncertainty. However, as Galegher points out, decision making does not require only gathering data about the surroundings

(Galegher, 1990). It also comprises the interpretation of the problems at hand, the definition of (sub)goals and strategies and effectively being able to make representations of the decisions to internal and external constituencies.

Being able to design appropriate and efficient solutions implies an understanding of how team decision-making is carried out. This is however not easy and to illustrate what difficulties designers may encounter Galegher writes (Galegher, 1990, p. 8):

> Real-world decision making, however, is typically not well specified, stable, or orderly enough to permit decision makers to understand their situation and consciously adopt a suitable problem solving approach. Rather, problems may contain elements of both uncertainty and equivocality, and are likely to present themselves at unpredictable times. Moreover, groups may lack awareness of the type of decision in which they are embroiled.

The uses of technology to support decision teams are problematic from other view-points as well. The implementation of command posts such as ROLF 2010 could easily encourage centralizing command and control. However, as van Creveld puts it (van Creveld, 2000):

> The more centralized the system, the greater the danger that it will be paralyzed if enemy action causes the directing brain to be eliminated or communications with it to be impaired.

This important statement gives cause for reconsidering novel organization forms and command structures on the one hand, as well as directives on how decision teams should operate on the other.

Focusing on the latter will most likely imply new work methods among the team members than those traditionally used. Furthermore, it is possible to commit the decision team to support the organization as a customer service by providing necessary information to different organizational parts when needed. However, when it comes to organizational changes where traditional hierarchies are shifted to post-bureaucratic, or network, organizations a problem of responsibility for made decisions appears. There have been successful attempts to reduce some of the side effects of hierarchies, but decision processes are still hierarchical in nature (Galbraith, 1993). An important issue is therefore how to support decision teams in determining when a delegation of mandate and responsibilities to other levels of command can be appropriate or not.

Computer support at the group level has traditionally only focused on providing means of communication and making certain aspects of group coordination explicit (Schmidt and Bannon, 1992). On the other hand, much more has been done to support individual decision makers, where there have

been several military projects involved. Since we are more interested in systems that try to "understand" the actions taken by the user rather than just providing a simple means of book-keeping and communication, we will restrict ourselves to Critiquing Systems.

5. CURRENT CRITIQUING SYSTEMS

5.1 Traditional critiquing

When constructing a critiquing system, it differs slightly from the design of a more traditional expert system in that the system expects a user to provide a solution to a given type of problem. This solution is then either compared to what the system itself would have suggested with its built-in background knowledge or analyzed with respect to predefined rules for constructing a solution and the metrics for evaluating them, also through domain knowledge provided by a domain expert.

Typically, the user scenario for a critiquing system is that a few experts on a certain topic are consulted to provide information about the standard operating guidelines for solving problems in the domain, common pitfalls, measures of success and other kinds of information of use to a system that should monitor and evaluate the performance of users. Users are then assumed not to be experts in the domain, yet sufficiently apt to propose somewhat correct solutions to problems that may arise in this domain and also proficient enough to understand criticism regarding their performance, if appropriately motivated. The user scenario, when knowledge has been entered and verified by the domain experts, is that a single user enters information (most often in written form) describing a suggestion to solve a specific problem. The kind of problems studied in military contexts have often been in the form of a classical planning problem where a clearly defined goal is to be reached through the use of some available resources under certain constraints (time-constraints or other).

Such well-defined problems are less common in real life - there are not always metrics that can be used to evaluate performance in a way that humans are not capable of doing better themselves. For instance, the reliability of information can best be evaluated by asking intelligence officers responsible, so no easy automatic verification mechanism would be readily available. Thus, the commonly accepted preconditions for critiquing systems are not likely to be met in real operating conditions.

5.2 Formulating plans as a user

In the case of helping a user to formulate a military plan according to a formal description of a plan (most often in the "Course of Action" formalism for military plans (US Army, 1997)) Kott et al. used a top-down approach in a project centered around the CADET (Kott et al., 2002; Group, 2003) tool. In that project, plans were given a sketch-like description at first and later made more concrete as they were described in terms of how they should be achieved. The narrowing of goals was supposed to guide the user to provide all the necessary information so as to produce at least a complete description of the plan. When providing this guidance for the user, CADET informed the user of available resources but was not overly strict about constraint satisfaction in every step, much like a word-processor that actually allows you to mistype since it is much faster to correct errors afterwards. The time constraints for the whole operation were shown in a matrix describing the interdependencies between different parts.

Previously in similar projects — that is, in projects that had the user specify a plan according to a top-down approach — the guidance for the user was often conducted so that only syntactically correct plans could be constructed as a result of the process. The syntactical control consisted of, for instance, making sure that descriptions of "how to achieve goals" were made increasingly more specific. The user was supposed to progress along the branches of the "tree" formed by having the "root" node as being your main intention with the plan and the children to a node being the divisions of a task into more manageable units. For example the INSPECT/EXPECT project (ISI, 2003; Valente et al., 1996) consisted of producing a verification tool for plans constructed with the Air Campaign Planning Tool used by the US Air Force. The verification tool could insure that basic constraints in such a hierarchical structure were fulfilled, such as the fact that nodes should either be a root node or have a parent, and either be leaves or have children. That is, a task would have to be decomposed into parts until some pre-defined level was reached.

6. DISCUSSION

There are several issues with previous approaches to decision support when compared to the ROLF 2010-like environments that signal a need to renew our perceptions of how critiquing should be performed.

6.1 Problems with formal verification

One of the problems with the formalization of military plans is that, with the development of the ROLF 2010 C^2 concept, we move in a direction where detailed control over subordinate units is sacrificed in favor of formulating a Commanders Intent as the primary means of exerting command and control. This has the effect of making plans that express this intent in fact *less* suited for formalization and formal verification and more like guidelines for human interpretation only. The act of verifying if an end-state has been achieved is not trivial if the end-state is in the form of "having control of a region".

In ROLF 2010 there have been studies around the concept and use of strategic optimizing simulations (Woodcock et al., 2003). These simulations have been intended to be used as a part of a decision support system, if provided with a critiquing system that could analyze the output of the simulations. The simulation environment is built on the use of genetic algorithms that evolve over many generations and hopefully converge on an approximate solution to the given problem. Such methods for solving problems have, however, inherent difficulties since it is difficult to understand what the evolutionary process creates. Even though the mechanisms that are used to create new solutions from old ones are fairly simple, there is little hope of understanding *why* a certain solution has been created instead of another and thus it may not be possible to give constructive criticism that reflects on the reasons for choosing it.

Though the need for verification may persist, it is not at all clear how such verification could be performed when there is so much freedom for the subordinate commanders to implement the given directives as they see fit. However, there may be other ways than merely using formal logics to deduce inconsistencies that can give valuable feedback on what is being planned or, in the case of a subordinate commander, following orders.

6.2 Decision process in ROLF 2010

Given the descriptions above, both the one of the decision making process of a team in Section 4 and of the ROLF 2010 environment in Section 2, we can see that there are some special characteristics of the ROLF 2010 environment, compared to other environments that have been fairly well-studied. We need not worry too much about the execution of our plans, neither need we care about the activation of decisions since we rely on the fact that data are collected and made available to us either by means of technical equipment or by intelligence units. Also, the decision process is one where decisions are constantly being refined and amended and even

when delivered as orders, they are not very precise regarding what should be done by the subordinate units at hand. So what is actually meant by a certain decision process? Is it a process that applies to each member of a staff? Or should it apply to the whole staff, so that the staff at any given time only is concerned with one of the activities involved in planning? If neither of the above applies, then how does the model actually describe the work process?

The quality of the feedback we can receive from a critiquing system depends heavily on what information is available from the "answer" (a COA plan, output from some kind of simulation software etc.).

If given an answer with little supplementary information as to the motivation for that answer, the corresponding range of possible items to criticize would be enormous for a critiquing system.

The opposite problem also applies for systems that give answers as output. Many simulation-based decision support systems work in this way. The user (or a sensor system or other technical systems) provides input, which is treated by the decision support system. The output given is the decision support, meant to be used as a possible solution to a problem. A decision support system capable of analyzing complex situations perhaps can provide useful output. However, such systems are bound to be so complex and opaque that it is virtually impossible for the user to look "under the hood" of the system and objectively criticize the basis for the advice. In both situations, the part given the answer needs to be informed of how the answer was reached, otherwise it is very difficult to relate to it in an objective way.

6.3 Criticizing Decisions

New organizations and new technical systems do not only function as tools, they also fundamentally change the work process, and thus the context of the decision making. Actions that earlier were decided by single individuals may now be the result of several decisions made at different organizational levels. In the process of refining and constantly evolving orders, it is not trivial to identify the actual *decision* that is to be input to a decision support system. Even if decisions may be recognized as physical documents, their contents may not be defined well enough to allow for automated analysis of correctness.

The discussion above and the analysis of earlier decision support systems leads towards a new way to look at decision support. Earlier, the main function of a decision support system was either to provide a suggestion for a solution to a problem, or to provide criticism on a user suggestion. In modern command and control structures, working in an uncertain context, where it is getting increasingly difficult to identify when and where

decisions are made and should be made, systems that *support coordination* are probably more useful than traditional decision support systems. Future decision support has to consider the team perspective.

The most central part of optimizing team performance is coordination. A system that supports all team members and their specific roles, so they both can increase the individual effort and coordinate better, is likely to have a huge impact on the total team performance.

6.4 Suggested research

There are three research aspects of intelligent group support that we believe will be crucial to address in future research:

1. It is necessary to investigate if it is possible to create a representation of the activities in the network organization. Such a representation is needed, in real time, if the critiquing system is going to be able to provide feedback on the decisions and actions of different actors. How to collect information, and what information, are central parts of this problem.
2. If a critiquing system is to be used by several actors in an organization, it probably has to be transparent enough for the users to create a common understanding of the system's abilities and limitations. If not, the critiquing system itself is likely to become an impediment for the users. Therefore, we suggest that tests must be performed with low-fidelity simulations, but with real operators, in order to investigate the impact of such a system on organizational performance.
3. In contrast to other domains where critiquing systems have been used, such as medical applications, it is difficult to define what expert knowledge really is. Apart from obvious constraints on when decisions need to be delivered, and which protocols that should be followed, it is not clear how to specify what the quality of actions that a group performs is. Thus, allowing for the group members themselves to produce simple constraints they believe to be useful could give researchers insight into what kind of issues the critiquing system should address. To assess such knowledge is a crucial part in the work of developing a critiquing system. It should, however, be noted that there are some fundamental problems in doing this, especially since there are no such organizations as the ones intended available today, when designing such a critiquing system. Once again, this calls for studies using both simulations of the intended context and professional users.

6.5 Final remarks

In this article, we have analyzed the problem of traditional decision support in relation to the demands made by future organizational structures. We have also suggested three areas that need to be investigated: information gathering, user acceptance and knowledge gathering.

Military decision-making is a field where the consequences of decisions are so great that the subject of decision support hardly can be neglected. This paper has pointed to both the problem of formal verification in decision support systems and the specific problems that arise when trying to support network organizations. Only research efforts taking the actual difficulties and circumstances of collaborative work environments seriously are likely to give useful insights into the design of computer systems for future critiquing systems.

REFERENCES

Alberts, D. S., Gartska, J. J., and Stein, F. P. (2000). *Network Centric Warefare: Developing and Leveraging Information Superiority.* National Defense University Press, Washington, DC.

Brannick, M. T. and Prince, C. (1997). An overview of team performance measurement. In Brannick, M. T., Salas, E., and Prince, C., editors, *Team performance assessment and measurement: theory, methods, and applications,* Series in applied psychology, pages 3Ð16. Lawrence Erlbaum Associates, Mahwah, N.J.

Cannon-Bowers, J. A., Salas, E., and Converse, S. (1993). Shared mental models in expert team decision making. In Castellan, N. J., editor, *Individual and group decision making: current issues,* pages 221-246. Lawrence Erlbaum Associates, Hillsdale, N.J.

Cebrowski, A. K. and Garstka, J. J. (1998). Network-centric warfare: Its origin and future. *U.S.Naval Institute Proceedings,* 124(1):28-35.

Galbraith, J. R. (1993). Challenges to the established order. In Galbraith, J. R., Lawler III, E. E., and Associates, editors, *Organizing for the Future: The New Logic for Managing Complex Organizations,* pages 1-12. Jossey-Bass, San Francisco, Calif.

Galegher, J. (1990), Technology for intellectual teamwork: Perspectives on research and design. In Galegher, J., Kraut, R., and Egido, C., editors, *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work,* pages 1-20. Lawrence Erlbaum Associates, Hillsdale, New Jersey. http://www.erlbaum.com/60.htm.

Group, L. C. (2003). CADET. http://www.arl.army.mil/aro/arowash/rt/sbir/99brochure/carnegie.htm.

ISI (2003). EXPECT. http://www.isi.edu/ikcap/expect/.

Kott, A., Ground, L., Budd, R., Rebbapragada, L., and Langston, J. (2002). Toward practical knowledge-based tools for battle planning and scheduling. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence,* pages 894-899, Edmonton, Alberta, Canada.

Kraemer, K. L. and King, J. L. (1988). Computer-based systems for cooperative work and group decision making. *ACM Computing Surveys,* 20(2):115Ð146.

Schmidt, K. and Bannon, L. (1992). Taking CSCW seriously. *Supporting Articulation Work, Computer Supported Cooperative Work -An International Journal,* 1(1- 2):7-41.

Silverman, B. G. (1992). *Critiquing Human Error - A Knowledge Based Human-Computer Collaboration Approach.* Academic Press, London.

Sundin, C. and Friman, H., editors (1998). *ROLF 2010 - A Mobile Joint Command and Control Concept.* Gotab Erlanders.

Sundin, C. and Friman, H., editors (2000). *ROLF 2010 - The Way Ahead and The First Step.* Gotab Erlanders, Stockholm.

US Army (1997). Field Manual 101-5: Staff Organisation and Operations. Department of the Army, Washington, D.C.

Valente, A., Gil, Y., and Swartout, W. (1996). INSPECT: An intelligent system for air campaign plan evaluation based on EXPECT. Technical report, USC Đ Information Sciences Institute.

Van Creveld, M. L. (2000). The Art of War: War and Military Thought. Cassell, London.

Woodcock, A. E. R., Hitchins, D. K., and Cobb, L. (September, 2003). The strategic management system (STRATMAS) and the deployment of adaptable battle staffs. http://www.dodccrp.org/Proceedings/DOCS/wcd00000/wcd0005c.htm.

This page intentionally left blank

ANALYSING DYNAMIC FUNCTION SCHEDULING DECISIONS

Karsten Loer, Michael Hildebrandt and Michael Harrison
Interdisciplinary Research Collaboration in Dependability (DIRC)[11],
Department of Computer Science, University of York, York YO10 5DD, UK

Abstract: Function allocation, as a process used in the construction of dependable complex systems, is a significant aspect of the design and implementation of interactive systems. It involves a documented and rational process for deciding what aspects of the system should be controlled by which human roles in the system and how the system should be automated to support these roles effectively. As computer systems have become more advanced, and the control of systems more complex, the notion of dynamic function allocation becomes increasingly desirable where in certain situations the automation may take over or give back function to the human user. In this paper we explore a further variant of dynamic function allocation that reflects typical work activity where the dynamic scheduling of activities takes place on the time dimension. The paper discusses this approach to dynamic function allocation called dynamic function scheduling and discusses the role that timed model checking may play in helping identify dependable dynamic function scheduling solutions.

Key words: Dynamic function scheduling; timed model checking.

1. INTRODUCTION

Complex work systems typically involve teams of people co-operating and using technology to achieve work goals. These goals are usually achieved under time constraint. In order to achieve them in a timely and reliable manner, the implementation of the *functions* that achieve the goals

[11] The DIRC project (see http://www.dirc.org.uk) is funded by the UK EPSRC, Grant N13999.

may vary according to situation. How functions are most reliably implemented in different situations is a vital and somewhat under-represented aspect of building a dependable system. This topic is dealt with in research into dynamic function allocation – see (Hancock and Scallen, 1998) and (Scerbo, 1996) for an overview. The overall focus of this work is about how automation can be used adaptively, according to the current demands on the system, and the capabilities and workload levels of the agents involved, in order to offer optimal support to the human operator.

The problem of function allocation is to take a set of functions that describe the work that the system is to do, in the *contexts* in which the work is to be carried out, and to decide how these functions should be implemented by the roles that are defined within the system. Methods are required that will enable system engineers both to take task descriptions and consider how the actions within the tasks should be implemented, and to take specific dynamic function allocation designs and analyse their implications. Typically the methods that exist are concerned with static allocations, that is, the decision about how roles are allocated to function occur at design time, see for example (IJHCS, 2000). In practice, it makes sense to consider the appropriateness of different configurations in different situations under different conditions of workload and different requirements of criteria such as situation awareness. Hence an in-car navigation system may have a different level of automation in which certain default inputs are presumed when the car is moving or in gear than when the car is stationary and in neutral.

In addition to sharing and trading functions among humans and automation, it may be possible to change the way functions are allocated *in time* in order to meet the required deadlines. Given that many modern work situations are rapidly evolving or highly scheduled, it is surprising how few human factors studies have attempted to make a conceptual or empirical contribution to understanding the temporal organisation of work – however, see for instance (DeKeyser, 1995), (Svenson and Maule, 1993) or (Hollnagel, 2000) for exceptions. Of particular relevance for designing function scheduling processes is a better understanding of temporal awareness (Grosjean and Terrier, 1999) and of the use of time as information (Michon, 1990), (Block, 1990). The authors are aware of little work that has been published on analytic approaches to function allocation, such as the analysis of a hydraulics system by (Doherty *et al.,* 2001) using the HyTech hybrid checker (Henzinger *et al.,* 1997).

There are a number of properties of temporal decision processes that are important to be understood if dynamic function scheduling is to enhance the dependability of systems. These include: (i) task arrival rates, (ii) predictability of task arrival, (iii) the agents' awareness of task arrivals

and event durations (and situation awareness in general), (iv) the agents' control mode (event-driven or anticipative; scrambled, opportunistic, tactical or strategic), (v) the uncertainty about future system states, monitoring and control lags, (vi) the pre-emptability of tasks, (vii) the deadlines of tasks relative to each other, (viii) a task's contribution to the system's objectives *(value),* (ix) the current priorities among system objectives, (x) the available resources and their service rates, (xi) the compatibility of concurrent tasks, (xii) the feasibility of combining, interleaving, postponing or dropping tasks, and (xiii) the discretion for satisficing and trading-off among system objectives.

This paper shall focus on a subset of these issues in the context of a particular system. The aim is to assess the role that timed model checking can play in helping to understand the trade-offs associated with decisions and thereby illustrate how the design of dynamic function allocation in general, and dynamic function scheduling in particular, can be aided by such checking. The paper is concerned with analysis techniques to support further exploration of dynamic function scheduling.

In Section 2 a case study based on a paintshop (Hildebrandt and Harrison, 2003) is introduced that illustrates a simple situation in which decision to delay or interrupt a function can be of value. Although it is relatively uncomplicated, this system raises important issues about the appropriate use of analysis techniques and problems associated with scaling these techniques. In Section 3 the `uppaal` (Larsen *et al.,* 1997) model of the paintshop system is described, and this is used as the basis of the analysis in Section 4. The `uppaal` hybrid model checker is capable of finding traces or counter-examples where constraints are broken. In a work design process, these traces can be used to generate scenarios where the timing constraints are violated. These scenarios form the basis for developing more appropriate scheduling and resource allocation mechanisms. The paper describes the model, the constraints that were used, and discusses the results of checking a variety of safety properties. The paper concludes with a discussion. Conclusions are drawn about how these techniques might be used more systematically, and objectives for future work are discussed.

2. CASE STUDY

The purpose of the example is that the following features of the design may be considered.

1. How resources can be allocated flexibly among multiple functions.
2. How functions can be allocated to agents along the system's time-line.

3. The action sequence of operators and what overall strategies for the implementation of a given function may be available. For instance, decisions may have to be made regarding the postponement, interleaving, synchronisation, speeding up or slowing down of function servicing, or regarding manual or automatic control. It may be appropriate to attach a notion of "value" to functions to describe the relative importance of a function and to allow the creation of priority structures among concurrent functions. Temporal properties of functions and agents are parameters in the decision process as well as variables that can be manipulated, i.e., temporal decisions are both based on and about time.

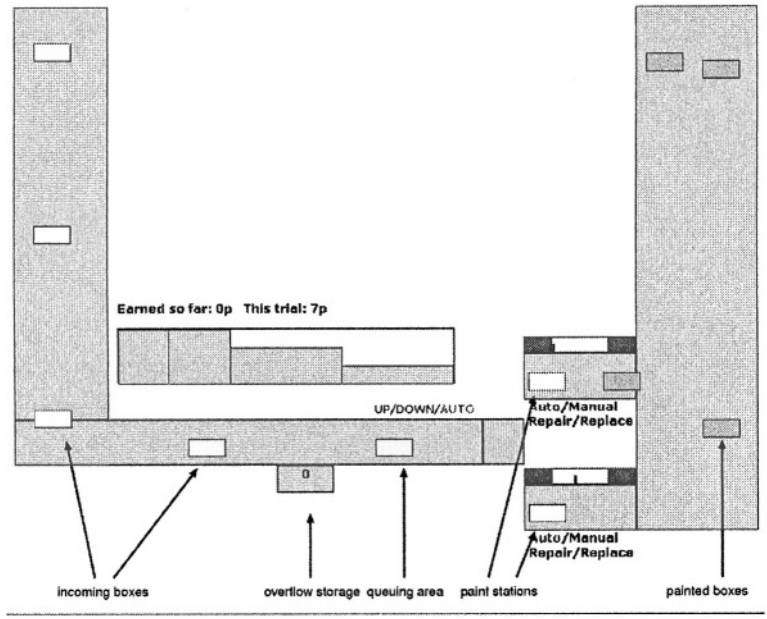

Figure 1. Sketch of the paintshop (Hildebrandt and Harrison, 2003).

PaintShop involves a conveyer belt that transports boxes to two parallel paint stations (Figure 1). Items to be painted enter the system at varying frequencies. A monetary reward is earned depending on the number of boxes painted, the number of boxes spoiled and any repair costs incurred. This system is also designed as a micro-world experiment and actual user strategies have been explored using experiment rather than model checking (Hildebrandt and Harrison, 2003). Boxes arrive at a distribution lift that allocates items to one of the stations. This process can be done automatically whereby the system allocates the box to an empty station. It can also be achieved by the operator overriding the decision of the distribution algorithm by using the 'up' and 'down' buttons forcing the lift in the specified

direction. Once the designated production line becomes available, the box is moved onto the paint station and the lift returns to the default position. The paint station can be set to automatic mode (which is the default) or manual mode. In automatic mode, the paint station will automatically specify the number of coats to be painted. The paint cycle for each coat of paint consists of a spraying period and a drying period. With each paint coat, the box whose initial colour is white will become darker. The rate of paint flowing through the nozzles is displayed just above each production line. The flow rate may decrease if nozzles become blocked or increase if the nozzle leaks. The paint process can also be performed manually. To paint an item, the operator has to click on a box and keep the mouse button pressed for a specified period of time. After this period the item will assume the new shade. If the mouse button is released before the minimum paint time the box is not painted and a spoiled box is released. In the model described in the next section, painting takes five time units in the automatic case and two time units in the manual case. When a nozzle ceases to function properly it can be repaired or replaced. Replacing a nozzle incurs no time cost but does incur a certain monetary cost. Repairing the nozzle incurs no monetary cost but causes a delay before the nozzle can be used again. In both the micro-world experiments and the model the cost and time variables were manipulated. Depending on the rate at which boxes arrive at the station, the state of the nozzles and the strategy used to employ the paint stations a certain proportion of the possible boxes will be painted. Boxes can fail to be painted and therefore rejected either because the appropriate procedure has not been carried out inside the paint station or because the queue of boxes waiting to be painted exceeds a certain number.

3. THE MODEL

The `uppaal` tool (Larsen *et al.*, 1997) was chosen to perform the modelling and analysis, as it permits the analysis of networks of linear hybrid automata with clocks whose rates may vary within a certain interval, is readily available and easy to use. The makes it possible to take different temporal reference systems into account, for example, the real-world frequency of items on the belt and the operator's perception of the frequency under varying workload. Automata may communicate either by means of integer variables (which are global) or by using binary communication channels. Communication occurs as a result of two process synchronisations using receiving actions `a?` and sending actions `a!`. Guards are used to describe the circumstances in which communications can take place. Automata may be guarded by conditions involving clocks that can be used to

represent delays or time invariants. It is not within the scope of this paper to describe the syntax and semantics of `uppaal` in detail, however the examples given below should be sufficiently clear to give the spirit of the approach. `Uppaal` provides tools for the simulation of systems – the state transition diagrams are animated, and the inter-process communication is displayed as an animated message sequence chart. The tool also supports analysis by state exploration. Thus it is possible to express and check for reachability properties such as:

1. "Is it possible to reach a state where the clock *x* is greater than 20?"
2. "Is it possible to reach a state where all boxes have been painted?"

It is beyond the scope of the paper to describe the details of the verifier – it suffices to describe both the properties that have been checked and those that could be checked.

The model consists of seven concurrent processes. The physical characteristics of the system are modelled as follows:

1. A *dispatcher* automaton dispatches objects to the incoming queue with a frequency that is determined by the workload – frequency is manipulated in the micro-world experiments. In the model that is illustrated in Figure 2a constantly high workload is assumed. This is encoded in terms of frequency, i.e. a new box arrives on the belt every two units (i.e. `workload=2`, values representing a medium and low workload are `3` and `4`, see Section 4.5). In order to reduce the complexity of the analysis, the number of boxes in the model is limited to 10. While it is acknowledged that this is a great simplification in comparison to the real-world continuous flows, this small model is sufficient for the purposes of this paper.

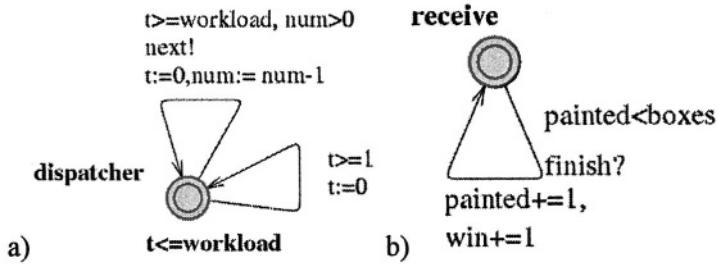

Figure 2. The (a) incoming and (b) receiving conveyor belts. (Key: `t`: clock, `num`: number of boxes yet to be dispatched, `workload`: encoding of workload as dispatch frequency, `painted`: number of finished boxes, `win`: win).

2. The *paint station* automaton (see Figure 3) – of which there are two instances (`station1` and `station2`) – models automatic and manual operation (top and bottom part of the automaton), fault occurrence and repair and replace costs. The severity of faults increases over time. A nozzle may break as soon as two items are painted but it will break for sure once four items are painted. Repairs cost 24 time units (see locations `repairingA` and `repairingM`). For a particular user, replacing a nozzle costs four tokens (see `user` automaton discussed below – note that such costs can vary, for instance, depending on a user's skills).

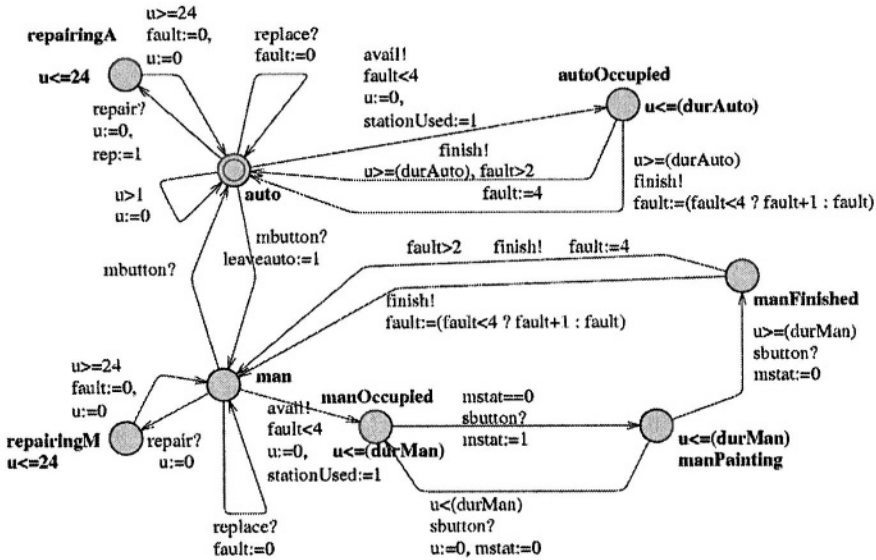

Figure 3. The paint station (Key: u: clock, fault: fault severity, mbutton: toggle manual/automatic painting, sbutton: press/release manual paint button, mstat: global flag denoting that manual painting is in progress, leaveauto: decoration that flags a mode change to manual mode).

3. The *waiter* automaton models the part of the system containing the queue of boxes `waiting` to be serviced by the paint stations as well as the lift that causes the boxes to be moved to one paint station or the other. It also models a repository for `unpainted` boxes that have fallen of the queue because the queue is too long, see Figure 4.
4. The final physical element, the *receiver,* models the belt of finished items, see Figure 2b.

3.1 The human interface and scheduling mechanism

Two processes are designed to reflect what the user does. *User* dispatches conditional user inputs and models simple repair/replace decisions: "if the fault (variables `p1fault` and `p2fault`) is bigger than 3 and sufficient funds (variable `win`) are available, replace a nozzle, otherwise perform a repair", see Figure 5a.

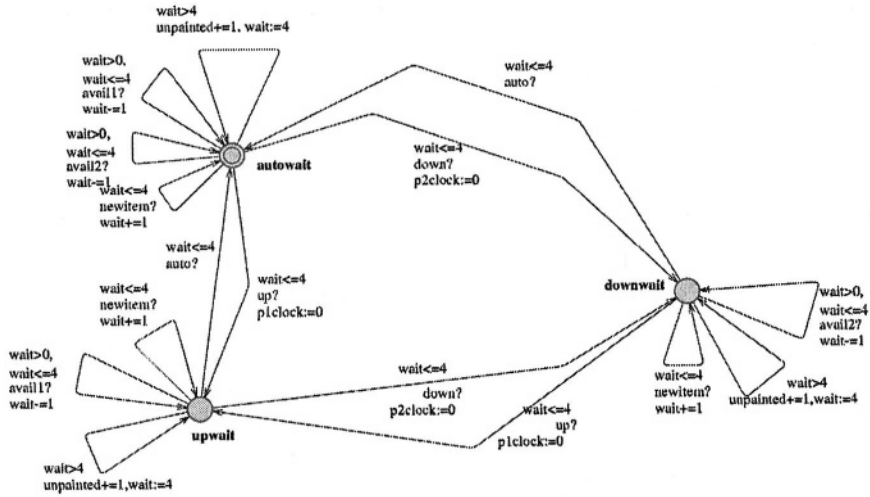

Figure 4. Boxes waiting to be channelled to the appropriate station (Key: `wait`: dispatched items waiting in queuing area, `unpainted`: overflow queue ot items failing to reach paint station, `p1clock`, `p2clock`: local clocks of paint stations).

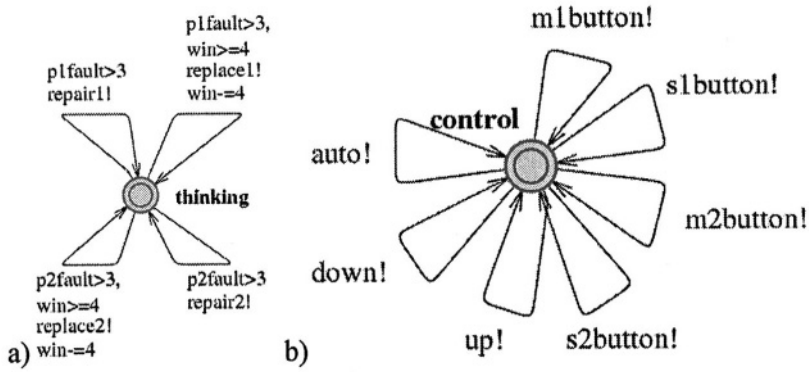

Figure 5. Simple models of (a) a user who implements a simple strategy and (b) a random user (Key: `win`: current earnings; `p1fault`, `p2fault`: fault severity of stations 1 and 2; `repair1`, `repair2`, `replace1`, `replace2`: repair/replace decision; `auto`: toggle

automatic station selection; `up/down`: select paint station manually; `m1button, m2button`: toggle manual/automatic painting; `s1button, s2button`: press/release manual paint button).

The *randomizer* (Figure 5b) provides an alternative process to the user which dispatches unconditional user inputs that are consumed by other processes ("monkey at the keyboard" style) but only generated when no internal synchronisations can be performed.

4. THE ANALYSIS

Analysis was performed on the system in a number of steps. Starting with some sanity checks to gain confidence that the model performs as intended, properties are then formulated in order to investigate possible scheduling decisions.

4.1 Sanity Checks

At this level properties are intended to assess whether the model provides the base functionality of the system effectively. Properties in this category include deadlock freedom and the reachability of system states that represent crucial system features, such as (i) different lengths of the drop-out queue, (ii) switching between automatic and manual paint mode, (iii) switching between paint stations and (iv) the concurrent operation of both paint stations.

4.2 Reachability of system goals

Once the results of the analysis in Section 4.1 give confidence that the model behaves as intended, the next stage is to assess whether system goals can be reached. For instance:

P1: *Can all n items be painted?*
The property ("`E<> painted==`*n*") is true for $0 \leq n \leq 10$.

When the negated property (here, the *never-claim* "`A[] painted!=`*n*" – "*n* items can never be painted") is checked, the model checker produces a trace that can be simulated. Stepping through that trace, the analyst is guided through a scenario where both manual and automatic mode of painting are applied. The simulation and the sequence chart provided by `uppaal` can point to simple flaws or instances of unexpected behaviour of the model. In order to obtain a broader understanding of the reasons behind flaws,

additional traces of similar instances are required. However, the tool only produces a single trace for each property. Additional traces, focussing on different aspects that may be considered contributing factors to a discovered problem, require a refinement of the property. For instance:

P2: *Can all n items be painted, using only a single paint station?*
For the analysis of this property the verifier shall explore only paths that involve a single instance of the paint station process. This is achieved by temporarily decorating the paint station by a write-once flag `stationUsed` (see Figure 3) that cannot be reset and that would be set to 1 if the second paint station was used. Property P1 then needs to be extended by a condition "`stationUsed==0`".

4.3 Finding out minimal durations under different conditions

Having considered properties associated with the verification of the model and with the reachability and mechanisms for achieving specific goals, the next stage is to consider temporal issues associated with the paint shop model.

P3: *Can all n items be painted in m time units, using only a single paint station?*
"`E<>(painted==`*n* `and stationUsed==0 and gtime==`*m*`)`"
This property was checked for different values *m* of a global clock gtime, in order to establish the minimal duration[12] (in this case 22 units for ten items, but the nozzle needs to be replaced at least twice, so the win is only two units – see first row of Table 1). Similarly, one can ask:

P4: *Can all items be painted in m time units, using both paint stations?*
Again, a minimal duration of 22 time units was found. However, while the execution time remains the same this time, only one of the nozzles needs to be replaced, so the monetary win is six units.

All traces above confirm that the fastest way to perform the work is to opt to paint it manually (compare top and bottom of Table 1). The effect the automatic strategy had on the duration was then analysed.

[12] From version 3.4 of `uppaal` it is possible to access execution duration for the trace that is generated. This is achieved using the "fastest" option within the "diagnostic trace" menu. This feature of the tool consumes a lot of resources and it turned out to be easier to use the cruder approach of iterating over *m*.

P5: *What is the minimal time required to paint all items automatically?* Here, user intervention is recorded by decorating the paint-station automaton with a temporary global write-once flag `leaveauto` (following the procedure described for property P2 above). The minimal time required to paint all items without manual intervention and by using both stations is 29 units.

Table 1. Minimal paint durations for manual and automatic mode allowing replacement costs.

Minimum required time for `manual` painting alone; allow replace										
duration (win)	number of items									
	1	2	3	4	5	6	7	8	9	10
1 station duration	4	6	8	10	12	14	16	18	20	22
win	(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)	(1)	(2)
2 stations duration	4	6	8	10	12	14	16	18	20	22
win	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(5)	(6)

Minimum required time for `automatic` painting alone; allow replace										
duration (win)	number of items									
	1	2	3	4	5	6	7	8	9	10
1 station duration	7	12	17	22	27	32	37	42	47	52
win	(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)	(1)	(2)
2 stations duration	8	10	12	14	17	19	22	24	27	29
win	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(5)	(6)

Table 2. Minimal paint durations for manual and automatic mode for maximising earnings.

Minimum required time for `manual` painting alone; maximise `win`										
duration (win)	number of items									
	1	2	3	4	5	6	7	8	9	10
1 station duration	4	6	8	10	34	36	38	40	64	66
win	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
2 stations duration	4	6	8	10	12	14	16	18	42	44
win	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)

Minimum required time for `automatic` painting alone; max. `win`										
duration (win)	number of items									
	1	2	3	4	5	6	7	8	9	10
1 station duration	7	12	17	22	51	56	61	66	95	100
win	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
2 stations duration	7	9	12	14	17	19	22	24	48	50
win	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)

The remaining row in Table 1 was obtained by analysing property P3 extended by condition "`leaveauto==0`". The analysis so far yields the following findings that might be used to devise operation strategies:

1. Painting items manually is faster than automatic painting.
2. Using both stations does not necessarily gain a time advantage over using a single station only.
3. However, using both stations can save repair costs if the operator is prepared to take the risk and leave one station broken.

It should be noted that the temporal properties of this stage could have been calculated in an alternative way by using a simple numeric model of the processes. However, the additional effort of creating the `uppaal` model pays off when multi-valued decisions are considered, as the following section demonstrates.

4.4 Focussing on monetary costs

So far the analysis has only been concerned with temporal costs and effects. The following properties have been used to check temporal *and* monetary costs associated with replacing faulty nozzles.

P6: *Can all boxes be painted without losing money?*
This property forces a search strategy where nozzle replacements are avoided. The resulting trace demonstrates that the task can be completed in 50 time units. The simulation demonstrates that both stations are used to paint in automatic mode until they break; then one station is repaired.

P7: *What is the shortest time for painting everything without losing money?*
The analysis yields that best performance (finishing the task in 44 time units) can be achieved, and the new trace suggests that this performance can only be achieved if manual control is selected. Again, both stations break, but the trace indicates that only one station needs to be repaired.

P8: *Can all items be painted without losing money, using only one paint station?*
This analysis is dual to P6, but focussing on a single paint station only (using the boolean flag procedure described in P3). This property is concerned with the robustness of the system and the additional temporal costs. The strategy exhibited by the model-checking trace could be used by an operator who does not have time pressure and therefore aims at maximising the win.

Analysing the durations under the assumption that temporal costs are secondary to monetary costs (see summary in Table 2) reveals again that the

best possible performance can be achieved by using both stations in manual mode, but the required duration increases to 44 units.

The results produced so far give some indication of what a good operation strategy might be under temporal and monetary cost extremes. However, it remains the task of the system designer to resolve if any of these strategies are suitable, and if they should be implemented as part of the system or as part of the operator training. For an informed decision it also remains necessary to draw on human-factors experience. A crucial additional factor that will influence this decision is the operator workload.

4.5 Variable workload

The analysis so far was performed assuming a constantly high workload, given by the *dispatcher* model in Figure 2a. The analysis can be repeated – using increasing, decreasing or alternating workloads – in order to collect insights about further strategies. Possible modifications of the dispatcher automaton are shown in Figure 6. However, for the purpose of this paper this analysis is omitted here.

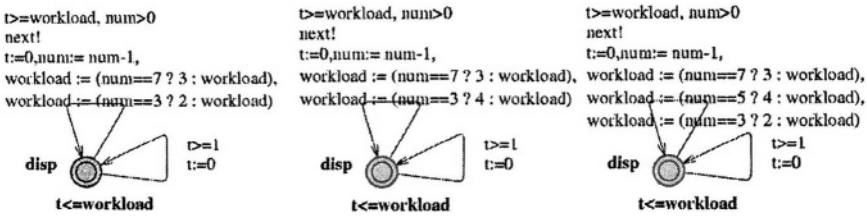

Figure 6. Modelling (a) increasing, (b) decreasing and (c) alternating workloads.

5. CONCLUSIONS

This paper discussed the feasibility of using model checking techniques to explore scheduling constraints in dynamic production systems under worst-case fault conditions. How the process might help in articulating the problems that must be resolved by human factors experts has also been briefly considered.

A number of problems emerged during the modelling and the analysis which could limit the utility of a model checking approach. First, model checking is not yet a light-weight method. Generating a state model is an effortful and time-consuming exercise, unless a model of the physical characteristics of the system has been produced in earlier stages of the design process. This is a problem which occurs equally with most other

formal modelling approaches, such as for instance micro-world simulation and is reduced as the modeller's skill increases.

Simplification of some physical characteristics (see Section 3) makes the model less representative of the physical system, and failures related to the interactions of these non-linear processes might be missed.

Another problem is introduced when the human operator is to be modelled. It is important to make the right assumptions about the operator's control mode (scrambled, opportunistic, tactical or strategic), the accuracy of the operator's temporal awareness (knowledge of available and required time, dynamics of change, probability of events, and so forth) and the operator's general situation awareness when modelling temporal reasoning performance. Formal modelling may improve the design process as it makes explicit the designer's assumptions about agents' capabilities, performance and availability, about the value and priority structure of functions in the system, and about the costs and benefits of a particular control strategy. Although the richness of naturalistic planning and control processes, and the complexity of scheduling decisions, may not be captured by these models, they help to assess how robustly a set of prototypical control strategies perform across a range of operational circumstances. This preliminary investigation explored only very simple strategies, and only analysed the effects on safety properties. These strategies tend to be focussed on extreme situations, such as gaining maximal earnings in a minimal duration. Consequently, the stated goal of assessing under what circumstances certain action can and should be delayed is limited to extreme behaviour. This is useful, since it is often extreme situations where failure has particularly dangerous effects. Although solutions to resolve extreme situations are relevant, it is essential to also consider the "normal" operating conditions. It is argued that these techniques are also useful in posing the problems clearly that must be solved by human factors experts for the particular system.

For the purpose of informing design decisions the value provided by traces that are obtained from the model checker is limited. The traces that are obtained represent single instances of behaviour that may indicate problems in the design. The `uppaal` tool supports the understanding of the component behaviour in a trace by providing animations of the automata. Additionally, the message sequence chart visualisation provides insights about the inter-process communication in the trace. However, single instances of behaviour rarely provide sufficient insights to discover problem tendencies. For a broader understanding of a problem, a *set* of traces that describe the same problem would be required. To our knowledge, no tool currently provides such information. The analysis of scheduling trade-offs will most likely require a combination of several different approaches. These

will include queuing models, production scheduling models, simulation approaches, work and task analysis techniques, and experimentation.

Future work will concentrate on assessing the contributions that each of these approaches can make towards improving our understanding of temporal planning and control, and their limitations in representing temporal properties. The appropriate method or methods for analysing flexible scheduling might be domain specific, as work domains themselves differ dramatically in their temporal properties (e.g. slow versus fast, synchronised versus independent, continuous versus discrete, periodic versus aperiodic, concurrent versus sequential, event-driven versus self-paced).

Work on elaborating the `uppaal` model of the paintshop continues. Parallel to this activity, a javascript micro-world simulation of the system has been developed in order to perform experimental studies (Hildebrandt and Harrison, 2003). In these studies, a human operator had the task of controlling paintshop. The study is currently being evaluated, and the results will be used to refine the `uppaal` model.

REFERENCES

Block, R., editor, 1990, *Cognitive Models of Psychological Time.* Lawrence Erlbaum Associates.

De Keyser, V., 1995, Time in ergonomics research. Ergonomics, **38**:1639–1660.

Doherty, G., Massink, M., and Faconti, G., 2001, Using hybrid automata to support human factors analysis in a critical system. *Journal of Formal Methods in System Design,* **19**(2):143–164.

Grosjean, V. and Terrier, P., 1999, Temporal awareness: Pivotal in performance? *Ergonomics,* **42**:1443–1456.

Hancock, P. and Scallen, S., 1998, Allocating functions in humanmachine systems. In R. Hoffman, M. S. and Warm, J., editors, Viewing psychology as a whole: the integrative science of William M. Dember, pp. 509–537.

Henzinger, T. A., Ho, P.-H., and Wong-Toi, H., 1997, HyTech: A Model Checker for Hybrid Systems. Software Tools for Technology Transfer, pp. 110–122.

Hildebrandt, M. and Harrison, M., 2002, Time-related trade-offs in dynamic function scheduling. In Johnson, C., editor, 21st European Annual Conference on Human Decision Making and Control, pages 89–95. GIST Technical Report G2002-1, Department of Computing Science, University of Glasgow, Scotland.

Hildebrandt, M. and Harrison, M. D., 2003, PaintShop – A microworld experiment investigating temporal control behaviour. Technical report, DIRC.

Hollnagel, E., 2000, Modeling the orderliness of human action. In Sarter, N. and Amalberti, R., editors, Cognitive engineering in the aviation domain. Lawrence Erlbaum Associates.

IJHCS, 2000, *International Journal of Human-Computer Studies.* **52**(2), Special Issue - Dialogues on Function Allocation.

Larsen, K. G., Pettersson, P., and Yi, W., 1997, Uppaal in a Nutshell. *Int. Journal on Software Tools for Technology Transfer,* **1**(1–2):134–152.

Michon, J., 1990, Implicit and explicit representations of time. In Block, R., editor, *Cognitive Models of Psychological Time,* pp. 37–58. Lawrence Erlbaum Associates.

Scerbo, M., 1996, Theoretical perspectives on adaptive automation. In Parasuraman, R. and Mouloua, M., editors, Automation and Human Performance: Theory and Applications, pp. 38–63. Lawrence Erlbaum Associates.

Svenson, O. and Maule, A., editors, 1993, Time Pressure and Stress in Human Judgement and Decision Making. Plenum Press.

FORMAL VERIFICATION AND VALIDATION OF INTERACTIVE SYSTEMS SPECIFICATIONS

From Informal Specifications to Formal Validation

Yamine Aït-Ameur[1], Benoit Breholée[2], Patrick Girard[1],
Laurent Guittet[1] And Francis Jambon[3]
[1] *LISI/ENSMA, BP 40109, Téléport 2, 86961 Futuroscope cedex, France*
[2] *ONERA-CERT-DTIM, 2 Avenue Edouard Belin, BP 4025, 31055 Toulouse cedex, France*
[3] *CLIPS-IMAG, BP 53, 291 avenue de la bibltothèque, 38041 Grenoble cedex 9, France*

E-mail: {yamine, girard, guittet}@ensma.fr, breholee@cert.fr, francis.jambon@imag.fr

Abstract: This paper proposes a development process for interactive systems based both on verification and validation methods. Our approach is formal and use at first the B Method. We show in this paper how formal B specifications can be derived from informal requirements in the informal notation UAN. Then, these B specifications are validated using the data oriented specification language EXPRESS. Several scenarios can be tested against these EXPRESS specifications.

Key words: B Method; EXPRESS; UAN; interaction properties; verification; validation; formal specification of interactive systems.

1. INTRODUCTION

Graphical user interfaces relying mostly on software, are being more and more used for safety-critical interactive systems –for example aircraft glass cockpits– the failure of which can cause injury or death to human beings. Consequently, as well as hardware, the software of these interactive systems needs a high level of dependability. Besides, on the one hand, the design process must insure the reliability of the system features in order to prevent disastrous breakdowns. On the other hand, the usability of the interactive system must be carefully carried out to avoid user misunderstanding that can trigger similar disastrous effects. So, the software dependability of these

safety-critical interactive systems rely as well on safety as on usability properties. Our work focuses on the use of formal techniques in order to increase the quality of HCI software and of all the processes resulting from the development, verification, design and validation activities.

In past workshops and conferences, we presented our approach through papers dealing with formal specifications of HCI software (Aït-Ameur et al. 1998a), formal verification of HCI software (Aït-Ameur et al. 1998), test based validation of existing applications (Jambon et al. 1999). This paper addresses another topic not tackled yet by our approach: design and formal validation of formal specifications with respect to informal requirements. This work completes the whole development process of a HCI software. Indeed, our approach uses the B formal technique for representing, verifying and refining specifications (Aït-Ameur et al. 1998a, Aït-Ameur et al. 1998, Jambon et al. 1999), test based validation of existing applications (Jambon et al. 1999), secure code generation (Jambon 2002) and integration of formal approaches (Girard et al. 2003).

This paper starts from the translation of the requirements in the UAN notation (Hix and Hartson 1993) and shows how B specifications can be derived from. Then, the EXPRESS formal data modeling language (EXPRESS 1994) is put into practice for the validation of the derived B specifications. We show how the B specifications can be translated to EXPRESS code which allows validation.

This paper is structured as follows. Section 2 reviews the different notations and formal techniques that have been experienced on HCI. Our approach and the case study –used to illustrate our approach– are also described in this section. Next section gives the UAN representation of the case study requirements. Section 4 presents the B technique and the specifications of the case study in B. Section 5 is related to validation. It presents the formal data modeling technique EXPRESS which allows the validation of the B specifications. We show how an automatic translation from B to EXPRESS can be performed and how this technique is applied to our case study. The result is a set of EXPRESS entities that are checked against various scenarios. Last, we conclude on the whole suggested approach.

2. NOTATIONS AND TECHNIQUES IN HCI: A BRIEF STATE OF THE ART

2.1 Notations & Formal techniques

In order to express HCI software requirements, several notations were suggested. As examples, MAD (for "Méthode Analytique de Description") (Scapin and Pierret-Golbreich 1990) and HTA (for Hierarchical Task Analysis) (Shepherd 1989) use a hierarchical decomposition of user tasks. On the other side, a notation like UAN (Hix and Hartson 1993) and its extension XUAN (Gray et al. 1994) allow the description of not only the interface feedback, but of the user behaviors as well. UAN specifications record the state of the interface and tasks are described as state evolutions. This state orientation of UAN facilitates translation to state based formal techniques –B for example.

Several techniques were used in the HCI area. These techniques differ from some point of views: semantics –algebraic or state based– verification –incremental proof or fully automatic proof– etc. Some of these techniques can be summarized in the following.

On the one hand, the first techniques are state based. They were based on automata through statecharts (Wellner 1989) and ATN (Waserman 1981) (Guittet 1995), Petri Nets (Accot et al. 1996) (Navarre et al. 2000). They have been extended to support temporal logics to allow automatic model checking like in CTL* (Paternò and Mezzanotte 1995), XTL (Brun 1997) and SMV (Clarke et al. 1986, McMillian 1992), or with the Lustre language (Roché 1998). The previous techniques support code generation and automatic proving. Other techniques supporting state based semantics and incremental proving and refinement like Z (Johnson 1995), VDM (Marshall 1986) or B (Aït-Ameur et al. 1998) were suggested.

On the second hand algebraic techniques have been applied with LOTOS (Paternò and Faconti 1992) for describing HCI software. The proofs are achieved by rewriting and refinement is performed by transformation. Other techniques based on higher order type systems have been experienced.

All these techniques cover a limited part of the development of an HCI. Our approach does not use only one technique, but it suggests to use several techniques which cooperate, choosing each technique where it has proved to be most efficient.

2.2 Our approach

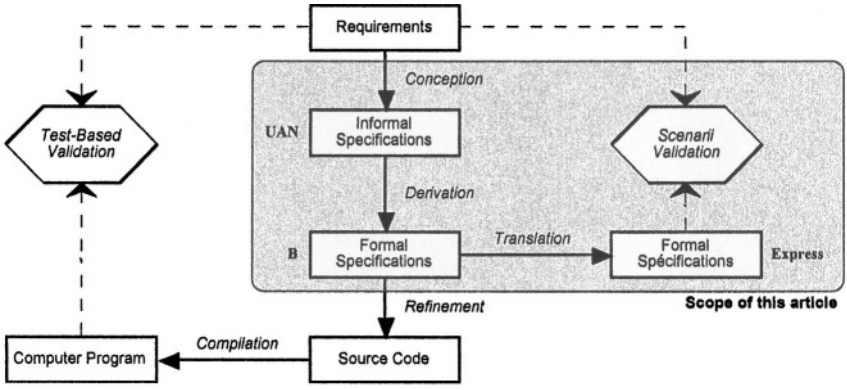

Figure 1: Scope of this article in the approach we suggest for handling the development and validation of HCI

Our approach uses the B technique. B supports formal specifications, refinement from specifications to code and property verification through the proof of the generated proof obligations. Specifications are derived from the informal UAN notation and are validated using the EXPRESS data modeling language.

Formal specifications, property verification and refinement from specification to code have been have been presented in (Aït-Ameur et al. 1998a ,Aït-Ameur et al. 1998, Jambon et al. 1999) respectively. This paper presents the last point: deriving specifications from semi-formal notations and their validation in EXPRESS. This paper completes the whole developed approach described in figure 1.

2.3 The case study: the *Rangeslider*

An usual slider –with a single cursor– is a graphical toolkit widget used by interface designers to allow the specification of a value in an interval. The *Rangeslider* (Ahlberg and Truve 1995) used by Spotfire™ (http://www.spotfire.com) is an enhanced version of this classical slider, i.e., it supplies two cursors –see fig. 2– in order to allow users to select not only a single value, but a range of values. This new widget is used by interface designers to implement easy-to-use zoom or filtering functions. A *Rangerslider* user can interact with the widget by the way of three different kinds of actions:

– **Move one cursor:** the user moves one of the two cursors, to the left or to the right. As a consequence, the area of the center zone expands or

reduces. The moved cursor cannot cover over the other cursor nor exceed the widget length.

- **Move the center zone:** the user moves the center zone, and at the same time both cursors come after it. So the area of the center zone remains unchanged. No cursor can exceed the widget length.
- **Select a value in outer zones:** the user clicks in one of the outer zones – MinZone or MaxZone– and the closest cursor moves at the selected point. As a consequence, the area of the center zone expands or reduces.

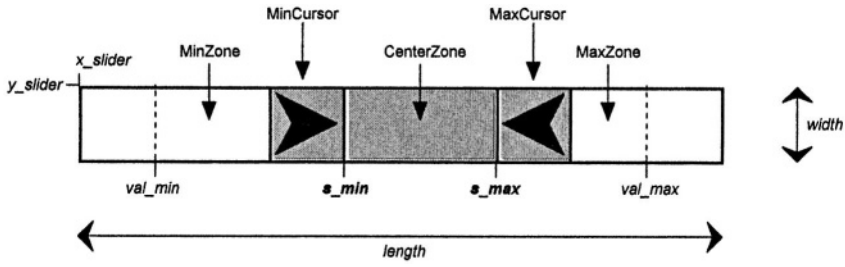

Figure 2: RangeSlider scheme —with variables names used both in the UAN, B and EXPRESS specifications.

This case study was proposed in the French working group ALF (Architectures, Languages and Formalisms) to study the expressiveness of a wide range of formalisms. Some interesting results, based on this case study, have already been proposed (Navarre et al. 2000).

3. THE USER ACTION NOTATION

The User Action Notation is an interaction-design notation. Hix et al. suggest that *"the UAN is intended to be written primarily by someone designing the interaction component of an interface, and to be read by all developers, particularly those designing and implementing the user interface software"* (Hix and Hartson 1993). The UAN is user- and task-oriented. A UAN specification describes, at the physical level, the user actions and their corresponding interface feedback and state changes.

The three tables below –table 2 to table 4– are the UAN specifications of the three user interactions described in §3.1. In fact, a full UAN specification must comprise five tables –one table for each user interaction. However the two pairs of UAN tables for cursor and outer zones are so similar that one table of each pair has been omitted. In these tables, *Rangeslider* is the name

of the whole slider object and Δx is the spatial increment on the abscissa axis.

In order to move the left cursor –MinCursor– the user must move the mouse button in the context of the MinCursor object. Then, he can depress the mouse button and drag the cursor. The MinCursor follows the mouse pointer and the center zone must be redisplayed. At each increment, the value of the `s_min` variable is updated.

Table 2: UAN specification of the task "move MinCursor"

TASK: **move MinCursor**		
USER ACTIONS	**INTERFACE FEEDBACK**	**INTERFACE STATE**
~[MinCursor] Mv	MinCursor !	
~[x,y in RangeSlider]*	0<x<s_max : MinCursor > ~ redisplay(CenterZone)	s_min=s_min+Δx
M^	MinCursor -!	

In order to move the center zone –CenterZone– the user must move the mouse button in the context of the CenterZone object. Then, he can depress the mouse button and drag the zone. The zone follows the mouse pointer and both cursors must be redisplayed. At each increment, the value of the `s_min` and `s_max` variables are updated.

Table 3: UAN specification of the task "move CenterZone"

TASK: **move CenterZone**		
USER ACTIONS	**INTERFACE FEEDBACK**	**INTERFACE STATE**
~[CenterZone] Mv	CenterZone !	
~[x,y in RangeSlider]*	s_min<x<s_max : CenterZone > ~ redisplay(MinCursor) redisplay(MaxCursor)	s_min=s_min+Δx s_max=s_max+Δx
M^	CenterZone -!	

In order to select a value in an outer zone –MinZone– the user must move the mouse button in the context of the MinZone object. Then, he depresses the mouse button. At this point, the left cursor –MinCursor– as well as the center zone must be redisplayed at the new position.

Table 4: UAN specification of the task "select a value in MinZone"

TASK: **select value in MinZone**		
USER ACTIONS	**INTERFACE FEEDBACK**	**INTERFACE STATE**

~[x,y in MinZone] Mv^	redisplay(MinCursor)@x,y redisplay(CenterZone)	s_min=s_min+Δx

These UAN tables, together with the figure 2 are the high level and informal specifications of the Rangeslider. These specifications are very useful for interface designers because they express in a rather short and precise way the behavior of the RangeSlider. However, the UAN is a notation to express requirements but cannot be used to prove or test the features of the interaction object we analyze. As an example, we cannot be sure that the cursor will never cover over the other cursor nor exceed the widget length. So we must now use formal methods to prove this kind of properties.

4. THE B TECHNIQUE: FORMAL SPECIFICATION

The B method, as VDM or Z, is based on model description. Like VDM, B uses preconditions and post-conditions (Hoare 1969, Hoare et al. 1987). Moreover, B is based on the weakest precondition technique of Dijkstra (Dijkstra 1976). Starting from this method, J.R. Abrial (Abrial 1996) has defined a logical calculus, named the generalized substitutions calculus. Our choice had been motivated by the fact that B is supported by tools (ClearSy 1997) which allow a complete formal development.

The following abstract machine describes what a set of range sliders is. It describes the set of all the sliders to be `SLIDERS` and two constants describing the length and the width of the screen. The `PROPERTIES` clause types these two constants and gives their corresponding values.

```
MACHINE
  the_slider
SETS
  SLIDERS
CONSTANTS
  screen_width, screen_height
PROPERTIES
  screen_width : NAT  ∧  screen_width = 800 ∧
  screen_height : NAT ∧  screen_height = 600
```

The model of this abstract machine is given by the attributes defined in the VARIABLES clause. The set `sliders` describes the set of the actually described range sliders. The other variables allow to access the attributes of a given range slider.

Informally, as described in figure 2, each range slider is characterized by:

- `x_slider` and `y_slider` are the coordinates of the up left corner of the window describing the range slider,
- `width` and `length` are respectively the width and the length of a given range slider,
- `val_min` and `val_max` are the minimal and maximal values associated to a range slider,
- and finally, `s_min` and `s_max` are the current low and up values of the described range slider.

```
VARIABLES
  sliders, x_slider, y_slider, width, length,
  val_min, val_max, s_min, s_max
```

All these variables are typed in the `INVARIANT` clause. This clause contains the properties that are always satisfied by the variables of the model. These properties shall be maintained by the operations that affect these variables. Two kinds of properties are described:

- typing properties that give types to the variables. The set `sliders` is declared as a subset of the set `SLIDERS`. Then, all the other variables are accessing functions and they are typed by their signature,
- safety properties which ensure a set of critical properties and model consistance. They are described in first order logic and are maintained by the B prover. They assert that the low (resp. Up) value of a slider shall be greater (resp. Lower) or equal to the minimal (resp. maximal) value of the range slider. Moreover, it states that the whole range slider is contained in the screen dimensions. This last assertion ensures visibility and reachability properties.

In the B language, these properties are described by:

```
INVARIANT
  sliders ⊂ SLIDERS ∧
  x_slider  ∈ sliders-->NAT ∧ y_slider  ∈ sliders-->NAT ∧
  width     ∈ sliders-->NAT ∧ length ∈ sliders-->NAT ∧
  val_min   ∈ sliders-->NAT ∧ val_max   ∈ sliders-->NAT ∧
  s_min     ∈ sliders-->NAT ∧ s_max     ∈ sliders-->NAT ∧
/* Safety properties of the slider */
  ∀ sl.(sl:sliders => (val_min(sl) >= 0)) ∧
  ∀ sl.(sl:sliders => (val_min(sl) <= s_min(sl))) ∧
  ∀ sl.(sl:sliders => (s_min(sl)<s_max(sl))) ∧
  ∀ sl.(sl:sliders => (s_max(sl) <= val_max(sl))) ∧
  ∀ sl.(sl:sliders => (val_max(sl) <= length(sl))) ∧
  ∀ sl.(sl:sliders => (x_slider(sl) ∈ 1..screen_width)) ∧
  ∀ sl.(sl:sliders => (y_slider(sl) ∈ 1..screen_height)) ∧
```

```
∀ sl.(sl:sliders => (x_slider(sl)+length(sl) ∈ 1..screen_width)) ∧
∀ sl.(sl:sliders => (y_slider(sl)+width(sl) ∈ 1..screen_height))
```

The first operation allows to create a range slider with XX, YY as coordinates of its left up corner. Its length and width are respectively given by the parameters LENGTH and WIDTH. Finally, VMIN and VMAX parameters indicates the minimal and maximal values of the range. The slider is created with VMIN and VMAX as initial minimal and maximal values. A preconditon ensures that the parameters are correctly typed and the invariant is maintained. It ensures that the creation of a range slider is correctly performed.

```
OPERATIONS
 create(XX,YY,LENGTH,WIDTH,VMIN,VMAX)=
 PRE
    sliders ≠ SLIDERS ∧
    XX ∈ NAT ∧ YY ∈ NAT ∧ WIDTH ∈ NAT ∧ LENGTH ∈ NAT ∧
    VMIN ∈ NAT ∧ VMAX ∈ NAT ∧ VMIN>=0 ∧ VMIN<VMAX ∧ VMAX<=LENGTH ∧
    XX ∈ 1..screen_width ∧    YY ∈ 1..screen_height ∧
    XX+LENGTH ∈ 1..screen_width ∧ YY+WIDTH ∈ 1..screen_height
 THEN
    ANY sl
    WHERE sl ∈ SLIDERS - sliders
    THEN
       sliders := sliders ∪ {sl}  ||
       x_slider(sl):=XX  ||  y_slider(sl):=YY  ||
       length(sl):=LENGTH  ||  width(sl):=WIDTH  ||
       val_min(sl):=VMIN  ||  val_max(sl):=VMAX  ||
       s_min(sl):=VMIN  ||  s_max(sl):=VMAX
    END
 END;
```

In order to keep this paper in a reasonable length, we show only one operation that manipulates the range slider. It allows to move the left value of the range slider to the left. In B this operation is described by:

```
move_left_slider(one_slider, new_left_min_value)=
PRE
   one_slider ∈ sliders ∧ new_left_min_value ∈ NAT ∧
   new_left_min_value > val_min(one_slider) ∧
   new_left_min_value < s_max(one_slider)
THEN
   s_min(one_slider) := new_left_min_value
END;
```

Other operations related to the range slider have been described in this abstract machine. Moreover, the whole application is represented by several

abstract machines not presented in this paper. Indeed, abstract machines related to the mouse management, to the direct manipulation and so on have been described. Finally, notice that the abstract machine described in B and presented in this paper has shown that it is possible to:

- ensure that a range slider remains in the screen limits,
- ensure that the low and up values of a range slider respect the definition of a range,
- move the low value, of a range slider to the left in order to decrease its left value, by running the corresponding operation.

For the whole developed abstract machine, the proof obligations have been generated. They all have been automatically proved. However, this specification has not been built at the first attempt. We had to enrich the preconditions and to remove other preconditions. Indeed, the prover behaves following:

- preconditions are not complete, therefore the proof cannot be achieved,
- preconditions are contradictory, then the user has to make new choices and to check the requirements.

Finally, about 40 proof obligations are generated for this application. We had to prove only 2 proof obligations using the interactive prover, i.e., "by hand". This shows that when the application is well specified following sound software engineering concepts, the proof phase can be considerably reduced.

All these properties are safety properties. In the next section we address the problem of the validation of such formal specifications that is not supported by the B formal technique.

5. THE EXPRESS LANGUAGE: VALIDATION

The EXPRESS language specifies formal data models. The language focuses on the definition of entities –types– which represent the objects –classes– we want to describe. EXPRESS is type oriented: entity types are defined at compile time and there is no concept of meta-class. Each entity is described by a set of characteristics called attributes. These attributes are characterized by a domain and constraints on these domains. An important aspect of these entities is that they are hierarchically structured allowing multiple inheritance as in several object oriented languages. This part of the specification describes the structural and the descriptive parts of the domain knowledge.

On the other hand, it is possible to describe processes on the data defined in the entities by introducing functions and procedures. These functions are used to define constraints, pre-conditions and post-conditions on the data. They are also used to specify how the values of some properties that may be derived from the values of other properties. This part of the specification describes the procedural part of the domain knowledge.

Finally, in EXPRESS, entities –data– and functions –processes– are embedded in a structure called a SCHEMA. These schemes may reference each other allowing a kind of modularity and therefore specification in the large possibilities. More details about the definition of this language can be found in (Schenck, Wilson 1994, Bouazza 1995).

5.1 Translation of B specifications to EXPRESS

The translation from B specifications to EXPRESS code is based on the semantics of generalized substitutions on which B is built. The idea consists in:

- representing the state variables of the model by an EXPRESS entity. This entity describes a state in the underlying transition system. According to the B semantics, this transition system describes the semantic model of the developed application,
- representing the invariant properties by global EXPRESS rules. Indeed, the properties that are described in the `INVARIANT` B clause are global properties that need to be satisfied at each state,
- and finally, representing operations by entities expressing the initial and the final states with local rules that express the relationship between the initial and the final states.

All the objects that are defined in an abstract machine are translated into EXPRESS. Each abstract machine corresponds to one EXPRESS schema.

5.2 The case study in EXPRESS

The following EXPRESS entity defines the model associated to the abstract machine described in B. It is obtained by a translation of all the variables that are described in the `VARIABLES` B clause.

```
SCHEMA The_Slider;

ENTITY Slider;
  x_slider,y_slider  :INTEGER;
  width, length      :INTEGER;
```

```
  val_min, val_max  :INTEGER;
  s_min, s_max      :INTEGER;
END_ENTITY;
```

For a given range slider, the previous entity describes the `x_slider` and `y_slider` representing its coordinates, its width and length, its minimal and maximal values and finally its low and up values. The instanciation of this entity allows to create a range slider. This entity preserves the identifiers introduced in the B abstract machine. Moreover, it encodes all the invariant properties which are related to typing of the variables.

The other invariant clauses that are related to the universally quantified properties, which express safety properties, are represented by a global EXPRESS rule. This rule expresses that all the instances of the entity `slider` satisfy the expressed logical properties. It states that the set of all the instances of a range slider satisfying these properties is exactly the set of all instances of a range slider. It is given by:

```
RULE coord FOR (Slider);
LOCAL
  sliders_ok, ens_sliders :
  SET OF Slider := [] ;
END_LOCAL;
  sliders_ok := QUERY(s <* Slider | ((s.val_min >=0) AND
  (s.val_min <= s.s_min) AND (s.s_min < s.s_max) AND
  (s.s_max <= s.val_max) AND (s.val_max <= s.length) AND
  (s.x_slider >= 0) AND (s.x_slider + s.length < 800) AND
  (s.y_slider >= 0) AND (s.y_slider + s.width < 600)));
  ens_sliders := QUERY(s <* Slider | true);
WHERE
  sliders_ok = ens_sliders ;
END_RULE;
```

Finally, operations are also transformed into an EXPRESS entity. The translation principle is based on the semantics of B. Indeed, the entity slider expresses the state of the described system (state based formal semantics). So, an operation, acting on a state, transforms an initial state `Ei` to a final state `Ef`.

The operation `move_left_slider` considers two states: the initial state `Ei` and the final state `Ef` and its input parameter, namely `new_left_min_value`. The description of this entity is given by:

```
ENTITY Move_Left_Slider;
  -- states
  Ei, Ef : Slider ;
  -- input parameter
  new_left_min_value : INTEGER;
```

The next part completes the description of an operation by an entity. It translates the precondition part (expressed by the B keyword `PRE`), the effect of the operation by expressing the change of `s_min` in the final state and finally it states the unchanged attributes in final state. The result gives the following `WHERE` rules.

```
WHERE
-- Translation of preconditions
      pre1: new_left_min_value >= Ei.val_min ;
      pre2: new_left_min_value < Ei.s_max ;
-- Translation of perations
    ope1: Ef.s_min = new_left_min_value;
-- Translation of unchanged state variables
      cst1: Ef.x_slider = Ei.x_slider ;
      cst2: Ef.y_slider = Ei.y_slider ;
      cst3: Ef.width = Ei.width ;
      cst4: Ef.length = Ei.length ;
      cst5: Ef.val_min = Ei.val_min ;
      cst6: Ef.val_max = Ei.val_max ;
      cst7: Ef.s_max = Ei.s_max ;
END_ENTITY;
....
END_SCHEMA;
```

This approach shows that it is possible to automatically translate B specifications into EXPRESS data modeling specifications. This translation will allow to give data models that represent specification tests.

5.3 Validation scenarios

In order to describe tests of B specifications –recall that validation and test are not supported by B– we need to describe instantiations of the EXPRESS data model.

As an illustration consider two rangesliders that are described by the same coordinates (`x_slider` =20 and `y_slider`=30), the same length and width (equal to `length`=100 and `width` = 10), the same minimal and maximal values (equal to `val_min`= 40 and `val_max`= 80) and the same up value (equals to `s_max` = 60). Consider that the first range slider RS1 corresponding to the initial state has a low value (equals to `s_min`= 50) and the second range slider RS2 has a low value (equals to `s_min` = 45). In fact this situation corresponds to a moving of the left value of a range slider. It can be expressed as `move_left_slider (RS, 45)`. Here we consider that the range sliders `RS1` and `RS2` corresponds respectively to the range

sliders of the initial and final states. In EXPRESS, this situation corresponds to the description of the three following instances:

```
#1=SLIDER (20, 30, 10, 100, 40 , 80, 50 , 60) ;
#2=SLIDER (20, 30, 10, 100, 40 , 80, 45 , 60) ;
#3=MOVE_LEFT_SLIDER(#1, #2, 45) ;
```

The previous set of instances represent a test case for the `move_left_slider` operation. The method can be generalized to other operations and to compositions of these operations that allow the description of a wide range of user scenarios. The test sequences can then be produced using the UAN specifications described in §3.3. Thanks to these specifications, a wide coverage can be achieved.

6. CONCLUSION

This paper shows a formal technique that allows to derive, verify and validate formal B specifications of HCI software. The informal requirements are expressed using the semi-formal notation UAN which is used as the basis for writing formal specifications. This process is proved helpful for writing formal specifications. Indeed, the direct derivation of these specifications from informal requirements is a hard task. This approach bridges the gap between user oriented specifications which feed the formalization process, the B formal development and verification techniques.

As a second step this paper addresses a crucial issue related to formal validation of formal specifications. It suggests to use a data oriented modeling language, namely EXPRESS, which allows to represent validation scenarios. This approach increases the efficiency of the HCI software development process since validation is not performed at the programming language level but at higher and abstract specifications. This approach allows to validate scenarios of application earlier in the development process. The result increases the efficiency of the development and decreases its cost.

Finally, to end the whole development process we suggest there is a need for taking into account user tasks descriptions and user tasks validations. This topic has not been addressed in this paper but it will be tackled in future developments. Indeed, we think that task representations and validations are possible within the framework we have developed.

REFERENCES

Abrial, J.-R. (1996) *The B Book: Assigning Programs to Meanings.* Cambridge University Press.

Accott, J., Chatty, S. and Palanque, P. (1996) A formal description of low level interaction and its application to multimodal interactive systems. In *Proceedings of Eurographics Workshop on Design, Specification, and Verification of Interactive Systems (DSV-IS'96)* (5-7 June, Namur, Belgium), Springer-Verlag, pp. 92-104.

Ahlberg, C. and Truve, S. (1995) Tight Coupling: Guiding User Actions in a Direct Manipulation Retrieval System. In *Proceedings of HCI'95 Conference on People and Computers X,* pp. 305-321.

Aït-Ameur, Y., Girard, P. and Jambon, F. (1998a) A Uniform approach for the Specification and Design of Interactive Systems: the B method. In *Proceedings of Eurographics Workshop on Design, Specification, and Verification of Interactive Systems (DSV-IS'98)* (3-5 June, Abingdon, UK), pp. 333-352.

Aït-Ameur, Y., Girard, P. and Jambon, F. (1998) Using the B formal approach for incremental specification design of interactive systems. In *Proc. of Engineering for Human-Computer Interaction,* Kluwer Academic Publishers, pp. 91-108.

Bouazza, M. (1995) *Le langage EXPRESS.* Hermès, Paris.

Brun, P. (1997) XTL: a temporal logic for the formal development of interactive systems. Palanque, P. et Paternò, F. (Ed.). In *Formal Methods for Human-Computer Interaction,* Springer-Verlag, pp. 121-139.

Clarke, E.M., Emerson, E.A. and Sistla, A. P. (1986) Automatic Verification of Finite-State Concurrent Systems using Temporal Logic Specifications. *ACM Transactions on Programming Languages and Systems.* 2, 8, pp. 244-263.

ClearSy.(1997) *Atelier B - version 3.5.* 1997.

Dijkstra, E. (1976) *A Discipline of Programming.* Prentice Hall, Englewood Cliff (NJ), USA.

EXPRESS. (1994) *The EXPRESS language reference manual.* ISO, 1994 ISO 10303-11.

Girard, P., Baron1, M. and Jambon, F. (2003) Integrating formal approaches in Human-Computer Interaction. In *Proceedings of INTERACT 2003 - Bringing the Bits togETHer - Ninth IFIP TC13 International Conference on Human-Computer Interaction - Workshop Closing the Gaps: Software Engineering and Human-Computer Interaction,* (September 1-5, Zurich, Switzerland).

Gray, P., England, D. and McGowan, S. (1994) *XUAN: Enhancing the UAN to capture temporal relation among actions.* Department of Computing Science, University of Glasgow, February, Department research report IS-94-02.

Guittet, L. (1995) *Contribution à l'Ingénierie des Interfaces Homme-Machine - Théorie des Interacteurs et Architecture H4 dans le système NODAOO.* Doctorat d'Université (PhD Thesis): Université de Poitiers.

Hix, D. and Hartson, H.R. (1993) *Developping user interfaces: Ensuring usability through product & process.* John Wiley & Sons, inc., Newyork, USA.

Hoare, C.A.R. (1969) An Axiomatic Basis for Computer Programming. *CACM.* 12, 10, pp. 576-583.

Hoare, C.A.R., Hayes, I.J., Jifeng, H., Morgan, C.C., Sanders, A.W., Sorensen, I.H., Spivey, J.M. and Sufrin, B.A. (1987) Laws of Programming. *CACM.* 30, 8.

Jambon, F. (2002) From Formal Specifications to Secure Implementations. In *Proceedings of Computer-Aided Design of User Interfaces (CADUI'2002)* (May 15-17, Valenciennes, France), Kluwer Academics, pp. **43-54.**

Jambon, F., Girard, P. and Boisdron, Y. (1999) Dialogue Validation from Task Analysis. In *Proceedings of Eurographics Workshop on Design, Specification, and Verification of Interactive Systems (DSV-IS'99)* (2-4 June, Universidade do Minho, Braga, Portugal), Springer-Verlag, pp. 205-224.

Johnson, C.W. (1995) Using Z to support the design of interactive, safety-critical systems. *IEE/BCS Software Engineering Journal.* 10, 2 (March), pp. 49-60.

Marshall, L.S. (1986) *A Formal Description Method for User Interface.* Ph.D Thesis : University of Manchester.

McMillian, K. (1992) *The SMV System.* Carnegie Mellon University, 1992.

Navarre, D., Palanque, P., Bastides, R. and Sy, O. (2000) Structuring interactive systems specifications for executability and prototypability. In *Proceedings of 7th Eurographics workshop on Design, Specification and Verification of Interactive Systems, DSV-IS'2000* (Limerick, Ireland), Springer Verlag.

Paternò, F. and Faconti, G.P. (1992) On the LOTOS use to describe graphical interaction. In Cambridge University Press, pp. 155-173.

Paternò, F. and Mezzanotte, M. (1995) Formal verification of undesired behaviours in the CERD case study. In *Proceedings of IFIP TC2/WG2.7 Working Conference on Engineering for Human-Computer Interaction (EHCI'95)* (14-18 August, Grand Targhee Resort (Yellowstone Park), USA), Chapman & Hall, 1995, pp. 213-226.

Roché, P. (1998) *Modélisation et validation d'interface homme-machine.* Doctorat d'Université (PhD Thesis) : École Nationale Supérieure de l'Aéronautique et de l'Espace.

Scapin, D.L. and Pierret-Golbreich, C. (1990) Towards a method for task description : MAD. Berliguet, L. et Berthelette, D. (Ed.). In *Working with display units,* Elsevier Science Publishers, North-Holland, pp. 371-380.

Schenck, D. and Wilson, P. (1994) *Information Modelling The EXPRESS Way.* Oxford University Press.

Shepherd, A. (1989) Analysis and training in information technology tasks. Diaper, D. (Ed.). In *Task Analysis for Human-Computer Interaction,* Ellis Horwood, Chichester, USA, pp. 15-55.

Waserman, A. (1981) User Software Engineering and the design of Interactive Systems. In *Proceedings of 5th IEEE International Conference on Software Engineering* ,IEEE society press, 1981, pp. 387-393.

Wellner, P. (1989) StateMaster : a UIMS based on Statecharts for prototyping and target implementation. In *Proceedings of Human Factors in Computing Systems (CHI'89)* (30 April - 4 May, Austin, USA), ACM/SIGCHI, pp. 177-182.

MODELLING INCIDENT SCENARIOS

To enrich User Interface Development

Claudia V. S. Guerrero[1], Maria de F. Q. V. Turnell[1], Jean-Marc Mercantini[2], Eugène Chouraqui[2], Fernando A. Q. Vieira[3] and Madson R. B. Pereira[4]

[1]Laboratório de Interfaces Homem Máquina da Universidade Federal de Campina Grande - Caixa Postal 10053 CEP 58109-970 Campina Grande PB - Brazil; [2]Laboratoire des Sciences de l'Information et des Systèmes, de l'Universitè d'Aix-Marseille III, Domaine Universitaire de Saint-Jerôme, Avenue Escadrille Normandie-Niemen 13397 Marseille Cedex 20 –France; [3]DOMO-DOS-SOC-DO and [4]SLOG-DO at CHESF - Companhia Hidro Eletrica do Sao Francisco, Rua Delmiro Gouveia, 333 – Bongi, CEP 50761-901, Recife PE, Brazil.

Abstract: This paper presents the process employed in obtaining a conceptual model of human errors scenarios for the electrical power industry. The model presented results from the analysis of the industry's reports on human and operational errors using a knowledge acquisition method (KOD). These scenarios will be used to build a process control simulator with which it will be possible to study the user behaviour when dealing with safety critical situations. From these studies it will be extracted a cognitive model of the user behaviour when working under critical situations to be incorporated into a method for the conception of user interfaces based upon ergonomic concerns (MCIE).

Key words: User interface Design, Process control, Human Error analysis, knowledge acquisition

1. INTRODUCTION

As result of technology development, industrial machinery and tools have reached a high degree of perfection in their performance transferring to the human operators the responsibility of almost all of the failures that happen during the interaction between them. According to Amalberti[1] human errors have almost always been considered the main cause of accidents. This is the result of differences in work pace and in representation languages which lead into misunderstandings, responsible for the majority of

accidents reported[2]. On the other hand, when attempting to manage their own abilities and error rate, the operators risk to increase their fatigue and stress.

This problem reaches higher proportions in industrial applications supported by complex systems considered safety critical from the viewpoint of the consequences of errors and faults, whether in financial terms or in terms of their catastrophic consequences. For those systems, beyond precision and functionality, it is imperative to offer their users: built in safety, adaptability to different degrees of expertise and work situations and support to easy the learning.

According to Amalberti[1] the causes of human errors can be classified into two categories: (i) internal causes such as stress, fatigue, high cognitive loads associated to time pressure or lack of knowledge about the task and/or the system; and (ii) external causes such as badly conceived aids and problematic systems. A badly designed user interface may lead the operator into misinterpretations, causing decision making errors and putting at risk the system's operation. For these reasons and in these contexts, user interface adequacy becomes even more important and critical, since it is possible through its design to easy task completion and to reduce the cognitive loads.

The user always makes mistakes; it is necessary to accept them and try to ensure that they will not lead into accidents. The operator's risk perception causes an increase in safety margins when planning activities, an increase in the expected performance levels thus interfering with the way of thinking and decision making[3]. From the designer's point of view, it means to build systems and user interfaces more robust from the viewpoint of error tolerance, acknowledging the user's right to make mistakes. So, it is essential to increase error visibility in order to facilitate its correction. In spite of the designer's efforts, in offering manuals, help assistants, and other kinds of help, errors will always occur. So it is important to research help strategies which are closer to the user's mental representation and language[1].

Difficulties in the communication between the designer and the final user have lead to considerable distances between the understanding of the task which the user actually performs and that which the designer assumes that the user must perform[3]. This is a vicious cycle which can only be broken by the ergonomical analysis of the user activity, by the positive feedback, and by the ergonomical validation of the tools developed to support the user activity. This ergonomical view of user interface design can be achieved by means of a user centred approach to interface design, which is based on the task analysis and on prototyping building and validation.

This paper presents how to obtain a conceptual model for human errors scenarios, for safety critical systems in the electric power industry, using the method KOD[4]. This work is part of a joint research project supported by CAPES-COFECUB, which involves the LIHM-UFCG in Brazil, the LSIS-Universite d'Aix-Marseille III, in France and the Brazilian Electric Power Company, CHESF.

The scenario model to be presented is the result of a preliminary study on the industry's database of incident and accident reports. This model will be used as the basis for building a process control simulator, with which it will be possible to study the user behaviour when dealing with safety critical situations. This approach is in accordance with Hollnagel's point of view which prescribes to consider the working context (critical situations) when modelling the user activity[5]. In these studies the user will be confronted with critical and atypical situations and his behaviour will be registered and analysed in order to build a user behaviour cognitive model when dealing with these situations. The observations will take place in the LIHM, with an infrastructure adequate for registering in different media the interactions between operator and simulator. From these studies it will be extracted a cognitive model of the user behaviour when working under critical situations to be incorporated into the MCIE Method for the Conception of Ergonomical Interfaces *(Método para a Concepção de Interfaces Ergonômicas)*[6], which is based upon ergonomic principles.

This paper is organised as follows. Section 2 gives an overview of the methods and tools employed in this research. Section 3 introduces the case study and the criteria adopted to choose the *corpus* of reports from the industry's database. Section 4 describes how the incident scenario model was built using the method KOD[4]. Section 5 discusses the results of the study and presents directions for future work.

2. FORMALISMS AND TOOLS

When developing safety critical systems in particular, the Ergonomics focus lies on the user interface design. In this context, optimizing the cooperation between the user and the system to perform a task implies in reducing the semantical difference between the user's real world and the application's abstract world which not necessarily share the same working logic[6]. In the attempt to produce better user interfaces various methods propose to integrate the ergonomical knowledge earlier in the process. Amongst the methods which are based on task models are: ERGO-START[7], MACIA[8], ALACIE[9], MEDITE[10] and MCIE[6].

The conception of ergonomic human interfaces for industrial applications is the focus of the research at the Human Interface Group (GIHM) at UFCG. The method employed is the MCIE. This method has already been applied to the development and usability evaluation of industrial user interfaces [11,12,6]. At the centre of this group's research is the project which aims to extend the MCIE and its supporting environment to enable designers of interfaces for safety critical applications to be able to conceive and evaluate the design's ergonomical adequacy.

Each of the MCIE phases is supported by model building and the design process is centred on evaluation. Thus the result of each conception phase must be evaluated before proceeding to the following one. It adopts a user centred approach in order to achieve an ergonomic result adequate to the expectations and abilities of the users when performing their tasks under critical situations. However, the user cognitive behaviour is still to be explored by the current development methods which tend to concentrate on the information related to *ergonomic* work analysis such as age, sex, knowledge, background, work strategies, etc. It is also necessary to consider those related to the cognitive abilities, more specifically to understand and consider the user behaviour when facing risk and critical situations.

The MCIE research project aims to incorporate such knowledge into the UI interface design process by means of incorporating a model of the cognitive user behaviour into the requirement phase. The knowledge to be considered is specific to the context of operating industrial systems under critical situations. With this knowledge added to the requirements it is expected to arrive to more ergonomically adapted user interfaces thus reducing and even eliminating a great deal of incidents in the industry. In figure 1 it is presented the MCIE process, its phases, related models and tools, highlighting the introduction of the cognitive model.

The conceptual model of error scenarios presented in this paper was built using the method KOD[4]. This method uses a bottom up approach in order to maximise the data extraction both from the specialist and text documents, reordering those under a model of homogenous structure. The choice of the method KOD is due to the following features:

- Based upon linguistic engineering, well adapted to extract knowledge from text expressed in natural language (such as the incident reports).
- Follows a bottom up approach so the model is gradually constructed from the raw data and knowledge.
- Guides the engineer from the knowledge extraction to the software model.

KOD requires elaborating three successive models: the practical model, the cognitive model and the software model. Each one is based on the paradigm

<*representation, action, interpretation*>. A system development process using KOD results in the models represented in figure 2.

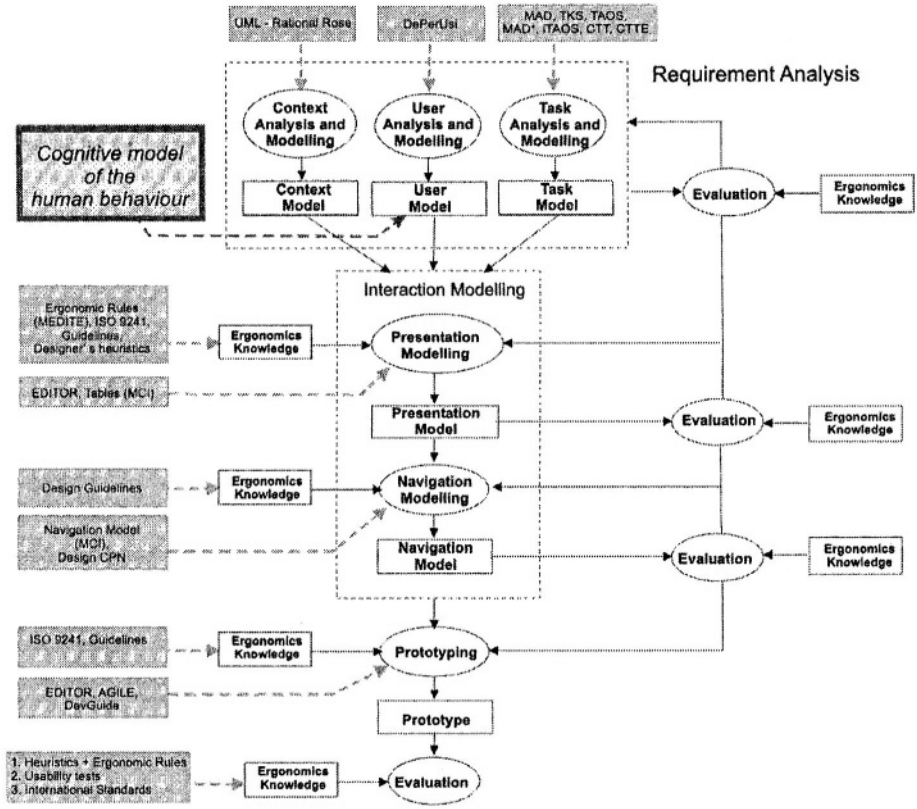

Figure 1, MCIE and the cognitive model of the user behaviour

This application of this method consists on the following steps:

- From each incident report a practical model (P.M) is constructed. This is the knowledge extraction operation.
- Based on the practical models, a cognitive model is elaborated. This is the abstraction operation.
- The software model is obtained through to the use of a formal language. This is the formalisation operation.

The KOD method constitutes a powerful framework to structure the domain knowledge. This method has already been applied to the domains: traffic accident modelling[13] and urban industrial site simulation [14].

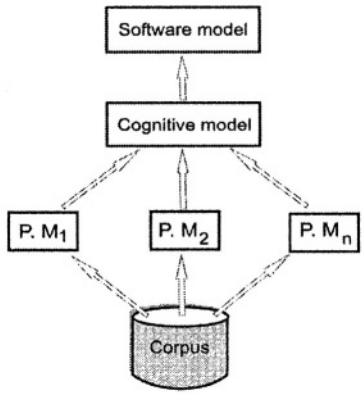

Figure 2. KOD Models

3. CASE STUDY

In the context of safety critical systems, a case study was chosen related to the electric power industry, in particular to the managing of the process involved in the transmission and distribution of electric power.

In the electrical Power Industry, according to annual reports on failures, a significant proportion of the incidents are due to human error. This industry is particularly interesting as a case study, not only because of its safety critical characteristics, but also because of its systematic approach in analysing and documenting failures. Also, it is an industry which mixes a high degree of automation in some of its operation centres but also keeps a good share of decision making and task performance on the hands of the operators. There, human errors may bring internal consequences such as material losses and incidents which may even endanger the lives of those who work for the industry, as well as the interruption of the service known as power supply cuts with the well known consequences to other industrial clients and to the society in general.

Typically, this process happens in a network of substations hierarchically organised into geographical regions, each of which associated to a control centre. The control centres manage the flow of electricity by supervising it and occasionally controlling specific substations, either remotely or via co-ordinated actions with the substation operators. Each substation has a set of input lines which brings in the electric power from the distribution network, which is then processed and passed on as outputs into the distribution network. The process may consist in changing the levels of electric tension or simply switching it between nodes of the network. The electric power

supplied in the output lines of the substation can feed other nodes in the network or be directed to consumers of the industry.

With the technology development, in the higher levels of automation, the task performed by the operators of the supervisory and control systems in this industry has increased in complexity. During system operation the demand for almost immediate responses and fast decision making, with little or no tolerance to errors, leads to an increasing cognitive load.

In spite of all efforts, errors will always occur during the operation of such systems. In this study, we concentrated our efforts on incident reports due to human errors. The aim is to understand the cognitive behaviour during task execution under critical situations and thus develop human interfaces which account for cognitive ergonomics thus leading into higher levels of safety for both the operators and the system.

3.1 Defining the Corpus

Due to safety regulations the power supply industry keeps a detailed record of incidents and accidents along many years of operation. Initially, as a part of a preliminary study, 21 reports were chosen from the industry's database. These reports are related to different scenarios of incidents which happened in different locations in the industry's network of substations, in the past two years of operation. This period was chosen to ensure updated information in respect to work patterns and tasks. From the set of 21 reports it was extracted the corpus which constituted this preliminary study. In this section we synthesise the results of the corpus analysis.

The typical scenarios of incidents were related to system operation during the so-called manoeuvres which are classified in this industry as: routine, maintenance and emergency. From the preliminary study it was found that the causes associated to human errors can be classified as: (i) internal, such as stress, fatigue, and lack of attention and excess self-confidence; and (ii) external such as insufficient documentation, faulty equipment and inadequate work conditions such as insufficient lighting.

A set of criteria was then adopted in order to select the documents relevant to the purposes of this study. The reports of interest for this preliminary study were about human errors related to tasks of control and supervision, performed either in the control room or in the patio of equipments. These criteria emerged from a first analysis resulting in an abstract written for each report. The next step consisted in synthesising all the relevant information on a table format highlighting: the task performed, the cause of the error (according to the point of view of industry's experts), the consequences of the error and a brief description of the scenario before and during the task execution which lead into error. This analysis made

possible to identify the corpus relevant to this preliminary study. The corpus of this study consists of 8 reports on human errors and equipment failures which occurred on 8 different sites of the industry.

4. BUILDING THE CONCEPTUAL MODEL FOR THE HUMAN ERROR SCENARIO

In this section we introduce the process used in order to obtain the conceptual model of human errors using the method KOD[4] (figure 3).

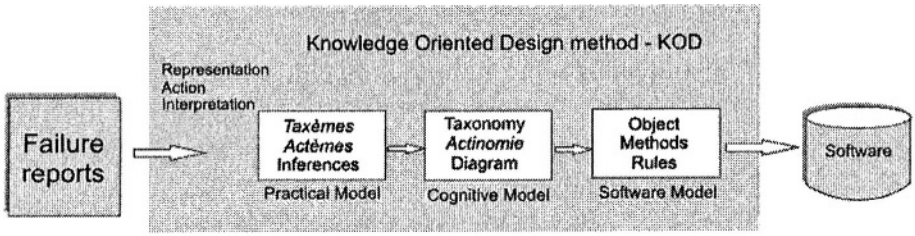

Figure 3. KOD process

The first step consisted in extracting from each raw document, from the corpus, all the elements relevant to the context of study. The terms extracted consisted in all the objects used for the task completion. Each one of those elements had to be described in a later step. The documents consisted of incident reports as well as documents from ANEEL, the Brazilian Electricity Regulatory Agency, which establishes the operator's tasks and the necessary tools and equipments.

The following step consisted in listing all the activities and objects mentioned in each document analysed, according to the *action* and *representation* aspects of the KOD paradigm. Elements from *interpretation* aspect of the KOD paradigm were not available on the reports analysed.

4.1 Practical Model

The practical model represents the discourse by means of the elements which belong to the domain of the problem to be solved. In this phase the aim is to formalise the elements extracted from the text and represent the discourse originally expressed in natural language by means of *taxèmes, actèmes* and inferences.

4.1.1 The *taxèmes*

The *taxèmes* are items extracted from the reports, which define the physical and conceptual objects used by experts in the domain. A list with 44 *taxèmes* was built. These have been formalised in triplets <object, attribute, value>. The *taxèmes* characterise an object from the real world, manipulated by the expert, performed by means of a relation (attribute) which links the object to a value. There are five types of relations: classifying (is-a, type-of, ...), identifying (is), descriptive (position, failure mode,...), structural (composed-of) and situational (is-in, is-below, is-above,...). To illustrate the *taxemes,* the list below refers to a type of object from the class - switch.

***Taxèmes* for the switch 101-cp:**

```
<Switch, type of, Interaction Device >
<Switch 101-cp, type of, Switch>
<Switch 101-cp, composed of, red lamp >
<Switch 101-cp, composed of, id-label>
<Switch 101-cp, composed of, green lamp >
<Switch 101-cp, composed of, id-label>
<Switch 101-cp, composed of, id-label>
<Switch 101-cp, composed of, handle >

< Switch SW13-cp, is a, Switch 101>
< Switch SW15-cp, is a, Switch 101>
< Switch SW14-cp, is a, Switch 101>

< handle, position, open >
< contact, position, open >
< handle, position, closed >
< contact, position, closed >

<Switch SW13-cp, is on, Equipment Control Panel>
<Switch SW14-cp, is on, Equipment Control panel>
<Switch SW15-cp, is on, Equipment Control panel>
< Switch 101-cp, failure mode, broken handle >
< Switch 101-cp, failure mode, red-lamp always on >
< Switch 101-cp, failure mode, red-lamp always off >
< Switch 101-cp, failure mode, green-lamp always on >
< Switch 101-cp, failure mode, green-lamp always off >
< Switch 101-cp, failure mode, red-lamp burnt>
< Switch 101-cp, failure mode, green-lamp burnt>
< Switch 101-cp, failure mode, contact always opened>
< Switch 101-cp, failure mode, contact always closed>
```

4.1.2 The *actèmes*

The *actèmes* are textual items extracted from reports, which describe the change of state of an object or concept used by the domain experts. In this case study a list with 52 *actèmes* has been built. Once identified, the *actèmes* are translated into a 7-tuple: <*Action Manager, Action, Addressee, Properties, State 1, State2, Instruments*>

- An Action Manager who performs the action
- An Action which causes the change
- An Addressee who undergoes the action
- The properties represent the way the action was executed
- The State 1– is the state of the addressee before the change
- The State 2– is the state of the addressee after the change
- Instruments – one or a set of instruments used to cause the change.

Figure 4 illustrates one of the *actèmes* of the model - 'To Close'.

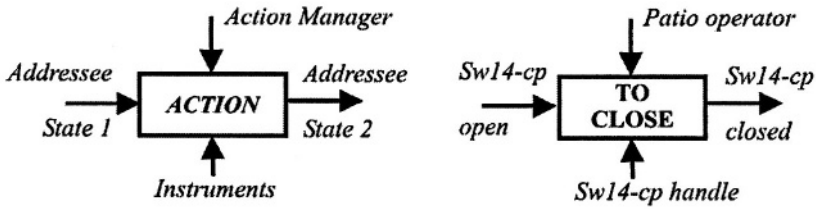

Figure 4. Representation of the *Actème* 'TO CLOSE'

Table 1. Actème 'TO CLOSE'

TO CLOSE (action on the switch 101)	
Components	Values
Action Manager	[control room operator; patio operator; regional centre operator]
Addressee	[Switch 101 of SW13-cp; [Switch 101 of SW14-cp; Switch 101 of SW15-cp]
State 1 (addressee)	
1) handle-position	1) [closed ; opened]
2) contact-position	2) [closed ; opened]
3) Red-Lamp state	3) [on, off , burnt]
4) Green-Lamp state	4) [on, off , burnt]
State 2 (addressee)	
1) handle-position	1) [closed ; opened]
2) contact-position	2) [closed ; opened]
3) Red-Lamp state	3) [on, off , burnt]
4) Green-Lamp state	4) [on, off , burnt]

TO CLOSE (action on the switch 101)	
Instruments	[operator's hands, electrical command]
Properties	Unknown (not mentioned in analyzed reports)

The next step, after building a practical model for each report, consisted on building the incident conceptual model or cognitive model, which will be presented in the following section.

4.2 Cognitive Model

The cognitive model is the abstraction of the practical models. It is composed of: *Taxonomies, Actinomies and Interpretation schemas.* The taxonomies are the result of the *taxéme* classification. It is presented as a tree structure showing the connections between each concept and related objects, as illustrated in Figure 5.

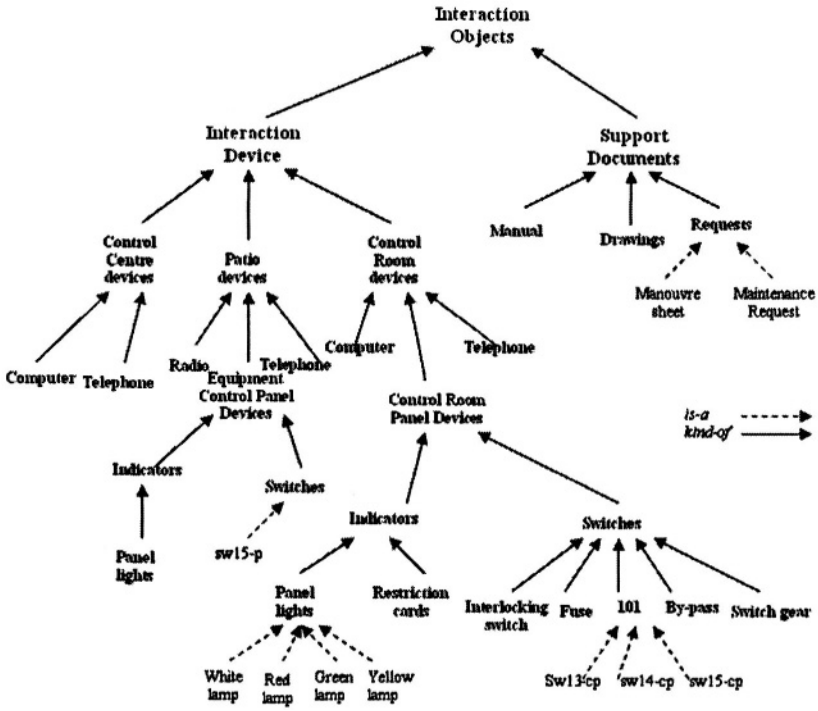

Figure 5. Example of Taxonomy

Each concept (represented in bold) must be defined, as exemplified in Tables 2 and 3. Switch is an interaction object used by the operator to select one amongst the possible states of a system element. A switch might also be associated to indicators which allow the operator to identify its current state.

Table 2. Switch Concept

Name: Switch
Reference:
Composed of: {handle, lamp, red lamp, green lamp, contact, id-label}
Located: {on equipment control panel, on control room panel }
Failure_mode: {broken handle, lamp always on, lamp always off, lamp burnt, contact always opened, contact always closed }

Table 3. Switch-101 concept

Name : Switch 101-cp
Reference: SW13-cp, SW14-cp, SW15-cp
Composed of: {handle, red-lamp, green-lamp, id-label, contact}
Localisation : on control room panel
Failure_mode: {broken handle, red-lamp always on, red-lamp always off, green-lamp always on, green-lamp always off, red-lamp burnt, green-lamp burnt, contact always opened, contact always closed }

The *actinomies* are the result of the *actémes* organisation according to scenarios of human errors. It follows the textual description of an incident scenario taken from one report of corpus analysed in the case study:

'During a manoeuvre to release the switch SWI5-p, it ignored the electrical command for opening, nor remote, or local. So, the patio operator, performing the manoeuvres on the switches, moved to the control room, and there arriving, received a request from the control room operator to agree the position of the switch SW15-cp with that of the red lamp (closed) and open it after, since he was at the telephone with the region's control operator agreeing on other manoeuvres. The patio operator moved to the control panel and unduly moved the handle of switch SW13-cp opening it. Noticing the error closed it after. Soon afterwards he moved the handle of switch SW14-cp opening it, interrupting the load of 1.7 MW, noticing his second error closed the switch. Continuing, he moved the handle of switch SW15-cp in order to open it, but it did not respond to his action remaining

closed. Finally, he communicated his colleague (control room operator) all his errors and problems with the switches.'

It follows the *actèmes* used to represent this particular scenario:

- To open <operator, to open, switch, property, switch closed, switch open, switch handle, hands >
- To close <operator, to open, switch, property, switch open, switch closed, switch handle >
- To move <operator, to move, operator, property, place of departure, place of arrival, feet>
- To request <operator1, request, operator2, property, operator2 without request, operator2 with request, communication tool>
- To communicate <operator1, communicate, operator2, property, operator uninformed, operator informed, communication tool)>
- To identify a problem<operator, to identify a problem, switch, property, unidentified problem, identified problem, cognitive system>

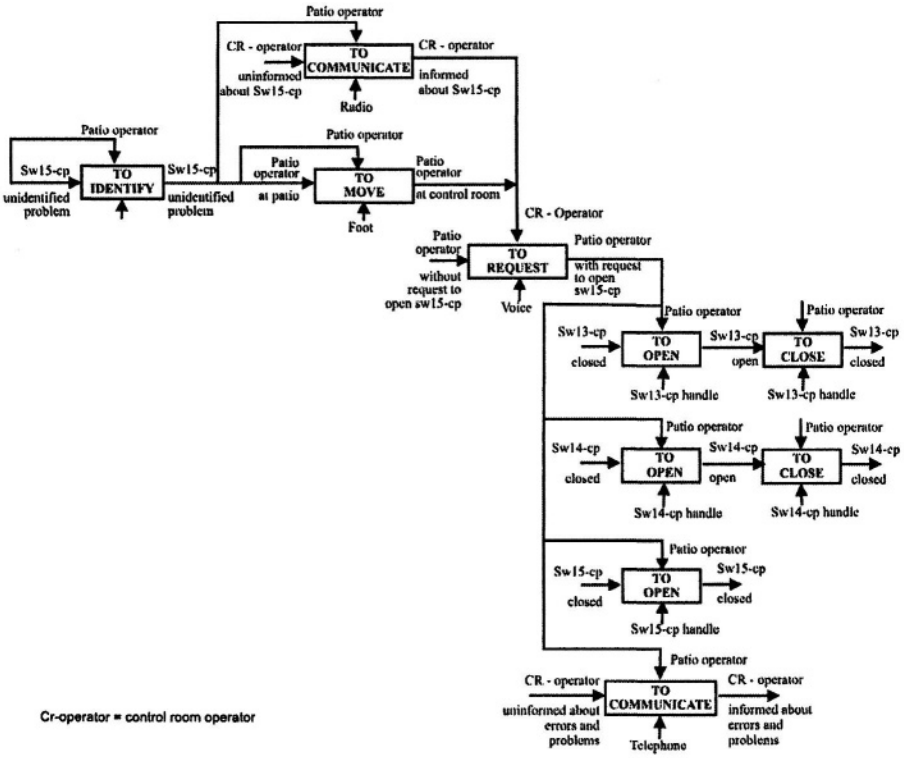

Figure 6. Example of *actinomie* for the case study scenario

4.3 Software Model

The software model is the formalisation of the conceptual model expressed in a formal language, independent of any programming language or computer platform. It must be expressed in high level and adapted to the nature and complexity of concepts to be represented. It consists on the integration of the taxonomy and concepts into the definition of classes and objects to be used in the development of the software. This model will only be built after completing this preliminary study with the analysis of a more significant number of reports.

5. CONCLUSIONS AND FUTURE DIRECTIONS

The use of KOD in this new domain has proven to be adequate to the extraction and organisation of the knowledge required to continue our research work.

One difficult faced consisted in understanding the documents composing the reports, since the industry's objective when writing them was to register the incident causes in order to access responsibilities and propose new work practices. Thus, some of the information there mentioned was not relevant from our research point of view whereas the relevant information was not always clear or complete. So, during the process of knowledge extraction the assistance of the technical personnel from the industry was crucial to understand the description of the systems and tools used to perform the tasks, the context in which the incidents happened, the description of environment where they occurred and particularly the terminology used in the reports (jargon, abbreviations,...).

Once the knowledge extraction is completed, one intended contribution from our work to the industry will consist in proposing the inclusion of aspects related to the operator's cognitive load and the explanation of how the experts (psychologists and engineers) who analyse the causes of incident arrive to the conclusions which are stated on the reports.

It is yet too soon to draw conclusion based upon the sample of reports used in this preliminary study. Thus it is evident the need for a wider analysis of the incident data base in order to represent the incident scenarios more precisely and thus build a simulator capable of better representing the reality of the industry. The model presented in the previous section represents in a structured way the essential knowledge for building the incident simulator, since it provides information on the scenarios and on the objects and actions performed by the operator during task execution.

On the other hand, although the corpus analysed is still very modest compared to number of cases available in the industry, it already allows us to identify the critical aspects of the operator's interfaces with the system, giving insights on how to improve their design in order to avoid the reappearance of problems. This is in itself a contribution to the design of interfaces for critical systems in the studied domain. However, a cognitive model of the operator's behaviour when dealing with incidents is still needed in order to understand the reasons which lead the user into carrying out wrong actions. This understanding will allow us to conceive systems which empower the user avoiding the occurrence of these errors.

Once this model is available the next step of this research project will consist in building the simulator and defining an experimental protocol for the tests which will be initially performed in the controlled environment of the lab, and later in the real work environment of the industry. After being validated, this model will be used in the conception process of user interfaces for critical applications, using the method MCIE. This approach of user interface development is expected to make them more adequate to the user cognitive needs and thus result in a reduction on the number of errors related to the interaction between operators and safety critical systems.

ACKNOWLEDGEMENTS

The authors would like to thank the agencies CAPES and COFECUB for partially financing this project.

REFERENCES

1. Amalberti, R. La conduite de systèmes à risques. Collection Le Travail Humain. 1996
2. Ganascia, J. G. Securité et Cognition. Editions HERMES. 1999
3. Guillermain, H. Ferrer, P. S. Contribution à l'identification des risques facteurs humains dans la conduite des processus à haut niveau de sûreté de fonctionnement, Securité et Cognition. Editions HERMES. 1999
4. Vogel, C. Le genie cognitif. Masson. 1988.
5. Hollnagel E. Cognitive Reliability and Error Analysis Method CREAM. Elsevier. 1998
6. Turnell, M. F. Q. V., Scaico, A., Sousa, M. R. F.; Perkusich, A. "Industrial User Interface Evaluation Based On Coloured Petri Nets Modelling and Analysis". Lecture Notes in Computer Science - Interactive Systems, LNCS 2220, p. 69-87, Germany, 2001.
7. Hammouche, H. De la modélisation des tâches utilisateurs au prototype de l'interface homme-machine, Thèse de Docteur, Université Paris VI, Décembre, 1995.
8. Furtado, M. E. S. Mise en oeuvre d'une méthode de conception d'interfaces adaptatives pour des systèmes de supervision à partir de spècifications conceptualles, PhD thesis, Doctorat de productique et Informatique à l'Univcrsité Aix Marseille III, France, 1997.

9. Gamboa, F. R. Spécification et Implémentation d'ALACIE: Atelier Logiciel d'Aide à la Conception d'Interfaces Ergonomiques, Thèse de Doctorat, Paris XI, Octobre, 1998.
10. Guerrero, C. V. S., Lula, B. Jr. Model-Guided and Task-Based Approach to UI Design Centered in a Unified Interaction and Architectural Model, CADUI'2002 - 4th International Conference on Computer-Aided Design of User Interfaces, Valenciennes, FRANCE, May 2002
11. Turnell, M.F.Q.V., Farias, G.F. The use of Supervisory Software in the Industrial Automation Process Control from the User Interface Perspective. 1996 IEEE International Conference on Systems, Man and Cybernetics; Beijing China, October, 1996.
12. Sousa, M. R. F., Turnell, M. F. Q. V. "User Interface Design Based on Coloured Petri Nets Modelling and Analysis". Proceedings of the 1998 IEEE International Conference on Systems Man and Cybernetics, San Diego, USA, 1998. S 2220, p. 69-87,
13. Mercantini, J.M., Capus, L., Chouraqui, E., Touriny, N., Knowledge Engineering contributions in traffic road accident analysis. In Innovations in Knowledge Engineering , pp 211-244, Ed. Ravi K. Jain, Ajith Abraham, Collette Faucher, Berend Jan Van der Zwaag. 2003.
14. Mercantini, J.M., Loschmann, R., Chouraqui, E. A provisional analysis method on safety of an urban industrial site, In Safety in the Modern Society, People and Work Research Report 33, pp 105-109, ISBN 951-802-338-7, 2000.

AUTOMATIC DEPENDENT SURVEILLANCE - BROADCAST / COCKPIT DISPLAY OF TRAFFIC INFORMATION: PILOT USE OF ELECTRONIC VS PAPER MAP DISPLAYS DURING AIRCRAFT NAVIGATION ON THE AIRPORT SURFACE

O. Veronika Prinzo, Ph.D.
Federal Aviation Administration, Office of Aerospace Medicine, Civil Aerospace Medical Institute, AAM-510, 6500 S MacArthur Bvd, Oklahoma City, OK 73179

Abstract: The Federal Aviation Administration is making a concerted effort to reduce runway incursions. A 5-day operational evaluation, conducted in October 2000, assessed pilot use of varying types of CDTI devices. Structured and unstructured taxi routes examined how well pilots navigated their aircraft using an electronic surface-map display (north-up, track-up) or a paper surface map. An analysis of 15 hours of communication data was performed to determine how the use of these displays might aid situation awareness and influence operational communications. A Type-of-Route x Type-of-Map ANOVA revealed more problems occurred and more messages were exchanged for structured taxi routes. A statistically significant interaction indicated most problems occurred for the north-up map during structured taxi routes and the number of problems encountered was comparable for the other maps when pilots navigated along unstructured taxi routes. Avionics developers may want to reconsider north-up surface moving map displays airport surface navigation tasks.

Key words: CDTI, ADS-B, moving map display

1. INTRODUCTION

Recreational and professional pilots form a diverse population of aviators who vary in piloting skills, experience with airport operations, and familiarity with the surface geography of their departure and destination airports. At one time or another, they – like all of us – make mistakes.

Sometimes, adverse weather or poor visibility add complexity and contribute to human error. The more serious mistakes can result in runway incursions, surface incidents, near-collision ground incidents, and fatal runway collisions.

In its special-investigation report entitled *Runway Incursions at Controlled Airports in the United States* (May 6, 1986), the National Transportation Safety Board (NTSB) noted a significant increase in collision ground incidents[13]. That report included several new safety recommendations to reduce the frequency of runway incursions. Some of these recommendations remained open when, on January 18, 1990, a fatal runway collision involving a Boeing 727 and a Beechcraft King Air A100 occurred at Atlanta, Georgia. As a result, the NTSB placed airport runway incursions on its "1990 Most Wanted Transportation Safety Improvements List," where it still remains.

The FAA is working diligently to address NTSB Safety Recommendation A-00-66 (NTSB, 2000): "… require, at all airports with scheduled passenger service, a ground movement safety system that will prevent runway incursions; the system should provide a direct warning capability to flight crews. In addition, the FAA should demonstrate through computer simulations or other means that the system will, in fact, prevent incursions." [14] A critical component of Safety Recommendation A-00-66 is that runway incursion prevention technologies should "provide a direct warning capability to flight crews."

In 2000 and again in 2002, the FAA's Office of Runway Safety made a concerted effort to reduce runway incursions. Several technologies that are being developed will provide a direct alerting capability to flight crews include ground markers, addressable signs, and surface moving maps. Under the Safe Flight 21 Program, contracts were awarded for avionics development and demonstration that included a surface moving-map capability. This capability was demonstrated (along with several others) in October 2000 during an operational evaluation of the automatic dependent surveillance broadcast (ADS-B) and cockpit display of traffic information (CDTI).

The FAA's intent in undertaking operational evaluation activities is to refine, standardize, and certify a set of tools that airports can acquire to

[13] In 1987, the FAA Administrator approved the definition of the term "runway incursion" as "any occurrence at an airport involving an aircraft, vehicle, person, or object on the ground that creates a collision hazard or results in loss of separation with an aircraft taking off, intending to take off, landing, or intending to land." This definition was clarified in 1996 to refer only to airports with operating control towers (Order 7050.1 2002).

[14] Letter of recommendation dated July 6,2000, to the FAA addressing runway incursions.

address their specific runway safety issues. The stated purposes of the operational evaluation were to develop and evaluate specific ADS-B air-air and air-ground applications, evaluate controller use of ADS-B, and demonstrate ADS-B technology. It also provided an opportunity to collect field data that could be used to guide the development of the ADS-B airport surface movement applications. These applications would improve surface surveillance and navigation through enhancements to airport surface situation awareness.

One goal associated with the airport surface situational awareness application was to enhance safety and mitigate occasions for runway incursion by providing pilots with tools that graphically display the proximate location of other surface aircraft and vehicles. Another was to enhance positional awareness by providing them with tools that displayed real-time information to supplement out-the window speed, direction, and position information.

To evaluate how ADS-B and surface-map information could be used to aid pilot situation awareness, very specific and complex taxi routes were created to examine how well flight crews navigated their aircraft along the assigned taxi routes using either an electronic surface-map display or a paper surface map. During the five-day event, objective (air traffic control voice tapes and radar data) and subjective data (surveys, questionnaires, jump-seat observer reports, small-group interviews) were collected. This report provides a general description of the communication findings.

2. METHOD

2.1 Participants

Twenty-five paid pilot volunteers flew 16 different aircraft. Two controllers and a coordinator (also volunteers) provided local- and ground-control services. They were on a temporary detail during training and on a regular schedule during the evaluation.

2.2 Materials

2.2.1 Experimental Structured Taxi Routes and Taxi-Route Cards

The experimentally constructed taxi routes (structured taxi routes) described the routes for pilots to navigate a defined course segment to or from the assigned runway. Pilots received individual uniquely labeled cards with these “canned” taxi routes presented in text format. Each card had a

named taxi route associated with it (e.g., CUPS1, FBO1, ANG1) that provided very specific, and often complex, taxi instructions. Each structured route was presented on a single sheet of paper, as in the example presented in Figure 1 (left panel). Ground controllers received these structured routes as graphical images with the name of the taxi route clearly labeled across the card, as shown in the right panel of Figure 1.

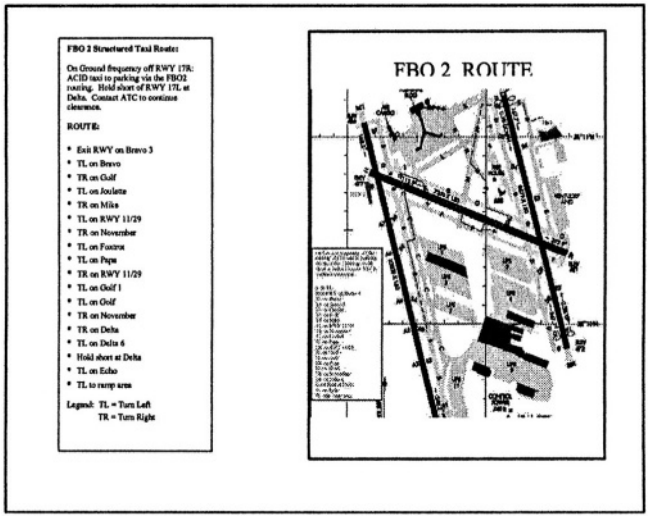

Figure 1. An Example of a Structured Inbound Taxi Route to Fixed Base Operations (FBO)

2.2.2 Traditional Unstructured Taxi Routes

A majority of the airport surface operations were performed using established taxi patterns, procedures and operations to and from the assigned runway and designated parking areas. For these unstructured, typical taxi routes, the ground controllers verbally provided pilots with the instructions necessary to taxi their aircraft to or from the assigned runways. Pilots did not know in advance the taxi routes they would be given.

2.2.3 Digitized Audiotapes

The Terminal Radar Approach Control (TRACON) facility provided one digital audiotape (DAT) for each test period. Separate voice records of all the transmissions made to the Ground East position were on the left channel. The right channel contained the Universal Time Coordinated (UTC) time code expressed in date, hour (hr), minute (min), and whole second (s).

2.3 Procedure

During the operational evaluation, the tower was divided into two sections, with the West portion of the airspace dedicated to the evaluation. In addition, a portion of the airfield was set apart from normal operations and tower controllers limited access to the West runway to participating aircraft. Flight periods lasted between 2 hr 19 min and 2 hr 59 min. The experimental flight periods were scheduled during normally low airport activity.

A majority of the airport surface operations were performed using customary taxi procedures[15] with the unstructured taxi routes — following initial call-up ground controllers issued taxi clearances such as the one in *FAA Order 7710.65M Air Traffic Control* (2000): "American Four Ninety Two, Runway Three Six Left, taxi via taxiway Charlie, hold short of Runway Two Seven Right." However, for some portions of the taxi route, they instructed pilots to proceed according to the script-defined taxi routes — using the structured taxi-route cards. The distance, the numbers of turns, and the complexity of the inbound and outbound taxi routes were controlled.

Prior to each flight period, ground controllers were instructed to clear participating aircraft via customary taxi routes (i.e., unstructured taxi routes) or defined structured taxi routes and monitor the aircraft's movement along its assigned taxi route to ensure compliance with the scripted scenario and FAA procedures. During pre-flight briefings pilots received sets of taxi-route cards to use when issued structured taxi-route clearances. They were instructed to interpret the textual route information presented on their taxi-route cards to determine the route to taxi.

Pilots taxied their aircraft along their assigned routes using Paper-Charts (35 segments), Track-up (11 segments) or North-up (22 segments) surface map overlays to find their way to the runway, ramp, or transient parking area. Each outbound taxi segment lasted between 530.0 s and 1763.0 s (M=1289.0 s, SE=89.11 s) during 9 structured routes and from 292.0 s to 1652.0 s (M=717.4 s, SE=59.8 s) during 20 unstructured routes. Each inbound taxi segment lasted between 520.0 s and 1321.0 s (M=734.82 s, SE=57.0 s) for 22 structured routes and from 134.0 s to 470.0 s (M=280.1 s, SE=64.8 s) for 17 unstructured routes.

[15] *FAA Order 7110M Air Traffic Control,* 3-7-2. TAXI AND GROUND MOVEMENT OPERATIONS was current at the time of the evaluation. "Issue, as required or requested, the route for the aircraft/vehicle to follow on the movement area in concise and easy to understand terms. When a taxi clearance to a runway is issued to an aircraft, confirm the aircraft has the correct runway assignment."

2.4 Experimental Design

This study used a two-factor, between-groups design. The between-groups factors were Taxi Route (Structured, Unstructured) and Type of Surface Map (Paper-Chart, North-up, and Track-up). Each structured and unstructured taxi route segment was assigned to a different, pre-selected flight-crew as part of their outbound or inbound taxi segment.

The type of ADS-B equipment installed in each aircraft determined its assignment to a Type of Map group. Nine aircraft comprised the Paper-Chart Group. They could display ADS-B equipped aircraft on their CDTI, but no map overlay was available of the airport surface. Five of the aircraft had scanned Jeppesen airport surface map overlays on their CDTI, always depicted in a north-up orientation. They were classified as the North-up Group. The remaining two aircraft were classified as the track-up group, since their aircraft had a CDTI with a vector-based moving map of the airport surface map available for display. The messages recorded during the structured and unstructured routes allowed for a comparison with taxi performance by the Paper-Chart, North-up, and Track-up Groups.

2.5 Dependent Measures

Operational efficiency for each structured and unstructured taxi segment was of primary interest. It consisted of communication workload and operational communications. Measures of communication workload included number and duration of communication. Measures of operational communication included problems and operational concerns.

To measure changes in communication workload and operational communication, the messages transmitted between the ground controller and pilot of each aircraft were grouped into transactional communication sets (TCSs) that included the pilot's first message to the ground controller and the last message that either switched the pilot to local control (outbound) or terminated at the ramp or transient parking area (inbound). TCSs are made up of communication sets that comprise all the messages between a controller and pilot that share a common goal or purpose (Prinzo 1996).

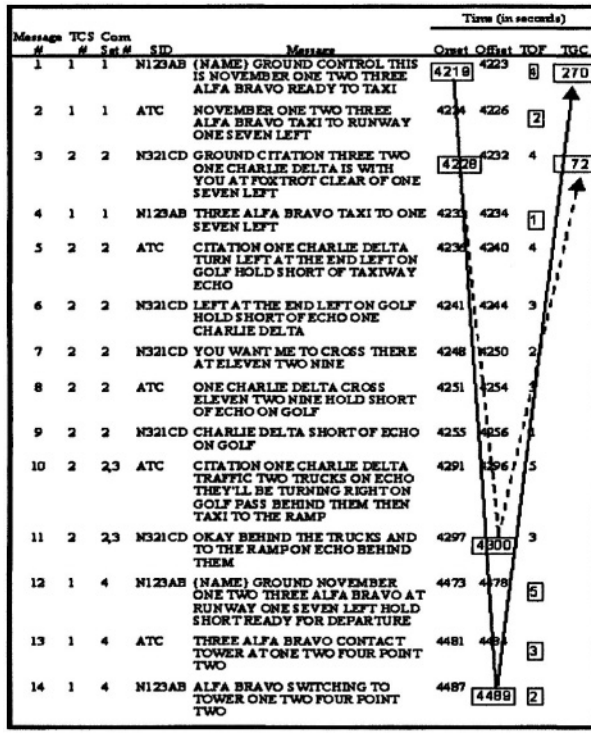

Figure 2. Example of a Transcript Encoded into Transactional Communication Sets

To illustrate, consider the partially encoded transcript presented in Figure 2. There are two TCS, one for each of two different taxi operations. TCS #1, which is an outbound taxi, consists of two communication sets: a taxi route clearance (messages 1, 2, and 4) and transfer of communications (messages 12 through 14). TCS #2 is comprised of three communication sets: position report (message 3), taxi route clearance (messages 5 through 11), traffic advisory (messages 10 and 11). Message 10 is complex, in that the first part of the message is a traffic-advisory while the latter part is a taxi instruction.

2.5.1 Objective Measures of Communication Workload

Four measures of communication workload were examined for each TSC. They included (1) number of messages transmitted, (2) time on frequency per message (TOF), (3) frequency occupancy time (FOT), and (4) time under ground control (TGC).

As shown in Figure 2, six messages involved N123AB (TCS #1). The TOF for the first message was 4 s. Frequency occupancy tune for TCS #1 was computed as the sum of the TOF. In the example, FOT was 17s (FOT =

Σ TOF = 4+2+1+5+3+2). As illustrated by the solid-line arrow, the time N123AB spent under ground control (time ground control, TGC) was computed as the time lapsed from the onset of the pilot's initial call-up in message 1 (at 4219 s) to the closing of the transaction in message 14 (at 4489 s). In the example, TGC was 270 s (4489 s - 4219 s). N321CD's TGC is shown by the dashed-line arrow and it was 72 s. A taxi segment typically began with a pilot checking in and ended in the transfer of communication to local control (outbound route), as was the case with TCS #1, or with the last recorded transmission as the pilot navigated back to either the ramp or transient parking area (inbound), which was the case with TCS #2.

2.5.2 Measures of Operational Communications

Communications that have the potential to adversely affect operational efficiency were identified and classified as problems and operational concerns. Problems included message reception (say again, did you copy), misunderstanding (readback error, stolen transmission, intentional repetition of a previous message for emphasis), erroneous information (incorrect call sign, can't find route segment provided in taxi instructions), and message production (self-correction of the call sign or another piece of information). Operational concerns involved spatial and positional awareness. Spatial awareness includes a general understanding of the airport's surface geography (aircraft is not on its assigned route, correction made to a previous taxi instruction, incorrect taxi clearance issued, instructions given to rejoin route, confusion, lost, missed turn), whereas positional awareness concerns the temporal and relational factors associated with maneuvering about the airport (maneuver around aircraft, possible conflict, request clearance to cross an active runway).

3. RESULTS

Operational communications were evaluated from verbatim transcripts and digitized voice recordings provided by the TRACON facility. Although requests were made during the planning of the event that baseline circuits be included, none were conducted. The analysis of voice tapes did allow for preliminary comparisons between flight crews who had access to traditional paper-charts and electronic airport surface maps, in either a north-up or track-up orientation, as aids to their surface situational awareness.

The analyses were restricted to taxi routes that either began or ended at the ramp or transient parking areas. Since progressive ground movement instructions include step-by-step routing directions, these taxi routes were

excluded because pilot variance in navigating on the airport surface would be restricted as ground control would be providing detailed instructions to guarantee safe and expeditious flow to the destination point. There were 727 messages (pilots=401, controllers=326) transmitted between participating pilots and controllers during the 31 structured and 37 unstructured taxi routes that involved 39 inbound and 29 outbound taxi segments.

3.1 Communication Workload

Multivariate Analysis of Variance (MANOVA) was performed on the taxi-segment means for each objective measure of communication workload presented in Table 1 (standard errors (SE) are enclosed in parentheses). Univariate Analysis of Variance (ANOVA) was used to assess the statistically significant findings. The Tukey Honestly Significant Difference (HSD) statistic was performed on statistically significant main effects and interactions. An alpha level of .05 was set for all statistical tests.

Table 1. Objective Measures of Communication Workload

	Measures of Communication Workload, in seconds (Standard Error)			
	TOF	N Messages	FOT	TGC
Structured Taxi Route				
Paper-chart	3.24 (.25)	11.00 (1.13)	35.89 (3.69)	716.67 (63.88)
North-up	3.51 (.37)	16.50 (1.70)	52.63 (5.54)	1383.88 (95.81)
Track-up	3.59 (.47)	10.60 (2.14)	35.20 (7.00)	759.20 (121.19)
Unstructured Taxi Route				
Paper-chart	3.64 (.25)	11.24 (1.16)	36.82 (3.80)	688.24 (65.73)
North-up	3.67 (.28)	7.71 (1.28)	27.86 (4.18)	299.36 (72.43)
Track-up	4.28 (.43)	7.50 (1.96)	32.33 (6.39)	536.50 (110.63)

A two-way Type-of-Route by Type-of-Map MANOVA revealed a statistically significant main effect for Type of Route [F (4,59)=9.76] and Type-of-Route by Type-of-Map interaction, [F(8,118)=6.85]. Subsequent Univariate ANOVAs revealed that not only were more messages transmitted during structured (M=12.70 SE=.99) compared with unstructured (M=8.82 SE=.87) taxi routes [F(1,62)=8.72] but more time was spent under the authority of ground control (Structured M=953.25 SE=55.73, Unstructured M=508.03 SE=49.22) [F(1,62)=35.86]. Although the time on frequency to transmit individual messages did not vary with the type of route navigated [F(1,62)=2.09], the overall frequency occupancy time increased by 9 s during the structured (M=41.24 SE=3.22) compared with unstructured (M=32.34 SE=2.84) taxi routes, [F(1,62)=4.29].

The statistically significant interaction revealed that the type of route navigated, in combination with the type of map available on the flight deck, affected communication workload for the number of messages transmitted

[F(2,62)=5.70], frequency occupancy time [F(2,62)=4.48] and time spent under the authority of ground control [F(2,62)=25.02]. In particular, Tukey post hoc comparisons clearly indicated that controllers and pilots in the north-up surface map group exchanged twice as many messages during the structured routes, as compared with unstructured routes.

The north-up surface map group spent nearly twice as long communicating during the structured, as opposed to unstructured taxi routes, and was under the authority of ground control for an additional 18 mins. Tukey results also showed that when the north-up surface map group traveled via structured taxi routes they exchanged more messages than the track-up surface map group that navigated the unstructured taxi routes. They also spent more time on frequency than the track-up surface map group during structured taxis and spent more time under the authority of ground control than the other participating groups, regardless of their assigned type of taxi route. The north-up surface map group that navigated unstructured taxi routes spent the least time under the authority of ground control.

3.2 Operational Communication

Previous research has demonstrated that when attentional resources are taxed, people may fail to detect that another person is talking, they may misspeak or mishear, or experience other problems identified from communication (Navarro, 1989). To gain some insights as to how the combination of the type of route navigated and type of map available affected communication workload, an examination of the operational communications was initiated.

3.2.1 Types of Problems

As shown in Table 2, 40% of the problems involved message reception (request from a pilot to have a transmission repeated, controller request for confirmation that a message was received). Another 16% of the problems involved misunderstandings (pilots incorrectly repeating back information or responding to an instruction meant for someone else). Erroneous information (5.2% of the problems) involved an incorrectly spoken aircraft call sign and pilot failure to find a taxi intersection. Mid-streamcorrections, consisted of mid-utterance repairs that involved either the aircraft's call sign or taxi information, and they included 34.2% of the identified problems. Lastly, intentional repetition or restatement of an earlier transmission occurred 5.3% of the time. In each recurrence, the controllers instructed the pilots to navigate their aircraft along the assigned taxi routes or runways.

Table 2. Frequency of Problems Presented by Type of Route by Type of Map

Types of Problem (n=38)	Type of Map						Total %
	Paper-chart		Track-up		North-up		
	S	U	S	U	S	U	
Message Reception	5.3%	10.6%	5.3%	2.6%	7.9%	7.9%	39.4%
Misunderstanding	5.3%	5.3%			2.6%	2.6%	15.8%
Erroneous Information	2.6%	2.6%					5.3%
Message Production	5.3%	10.6%	2.6%	2.6%	13.1%		34.2%
Intentional Repetition	2.6%	2.6%					5.3%
Total %	21.1%	31.6%	7.9%	5.3%	27.3%	10.5%	100.0%

3.2.2 Types of Operational Concerns

The types of operational concerns, along with their frequency of occurrence, are presented in Table 3. The operational concerns noted in the data involved either spatial awareness (aircraft is not on its assigned route, correction to taxi instructions, incorrect taxi clearance issued, instructions given to rejoin route, lost, missed turn) or positional awareness (maneuver around aircraft, possible conflict, request cleared to cross a runway). Approximately 75% of the operational concerns were related to spatial awareness, while the remaining 25% centered on position awareness.

3.3 Prevalence of Problems and Operational Concerns

Before examining the data for problems and operational concerns, a chi-square (χ^2) test revealed no statistically significant difference in the number of structured, compared with unstructured taxi routes completed, when pilots had available either the paper-chart, track-up or north-up surface maps, [$\chi^2(2)=1.24$]. Of the 68 taxi routes, 37 contained one or more problem. Subsequent chi-square tests revealed a significant difference in the number of problematic routes among participants in the north-up surface map group [$\chi^2(1)=12.57$]— 100% of their structured (8/8) and 21% of their unstructured (3/14) taxi routes were problematic. The number of problematic routes was equivalent when pilots navigated their assigned structured or unstructured taxi routes with paper-charts (Structured=11/18 Unstructured=10/17) or track-up (Structured=3/5 Unstructured=2/6) surface-map displays.

Table 3. Frequency of Operational Concerns Presented Type of Route by Type of Map

Types of Operational Concerns (n=33)	Type of Map						Total
	Paper-chart		Track-up		North-up		
	S	U	S	U	S	U	
Spatial Awareness							
Aircraft is not on its	3.0%		6.1%		3.0%		12.1%

	Type of Map						
assigned route							
Corrected taxi instructions			3.0%		3.0%		6.1%
Incorrect taxi clearance						6.1%	6.1%
Instructions given to rejoin route	3.0%					6.1%	9.1%
Lost	6.1%						6.1%
Missed turn	3.0%		3.0%		6.1%		12.1%
Position Awareness							
Maneuver around aircraft		3.0%					3.0%
Possible conflict	3.0%	3.0%		3.0%		3.0%	12.1%
Request cleared to cross a runway	15.2%	9.1%	3.0%		3.0%	3.0%	33.3%
Total	33.3%	15.1%	15.0%	3.0%	27.3%	6.0%	100.0%

A two-way Type-of-Route by Type-of-Map ANOVA was performed on the total number of problems and operational concerns associated with each assigned taxi-route clearance. As shown in Figure 3, more problems and operational concerns were present during the structured, as compared with the unstructured taxi routes [F(1,62)=9.82].

The Type-of-Route by Type-of-Map interaction also was statistically significant [F(2,62)=3.90]. Subsequently, the Tukey HSD statistic revealed significantly more overall problems; operational concerns resulted only for the north-up surface-map group during structured taxi routes, compared with the north-up and track-up surface map groups during unstructured taxi routes. Five of the 9 identified problems occurred during eight taxi operations and involved mid-stream corrections, three centered on problems in message reception, and one involved an incorrect readback of a taxi clearance. Eight of the 9 identified operational concerns involved spatial awareness (missed turn = 2, aircraft not on its assigned route =1, ATC issued instructions to rejoin a route = 2, incorrect taxi clearance issued = 2, and correction to taxi instructions = 1), and one concerned position operation (e.g., a pilot request to cross an active runway).

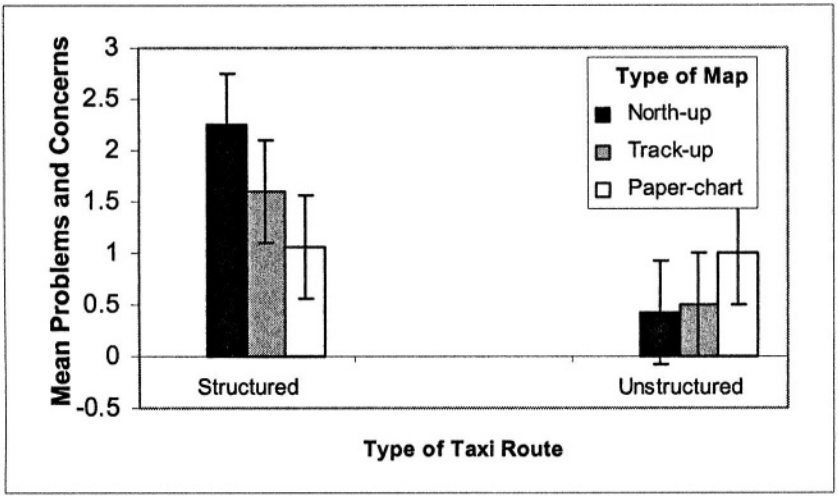

Figure 3. Problems and Operational Concerns Presented by Type of Route and Type of Map

4. DISCUSSION

The analysis of voice communications revealed that the combination of the type of assigned taxi route and surface map capability onboard the aircraft affected communication workload. Notably, participants in the north-up surface map group a) exchanged more messages than the track-up surface map group when they navigated the unstructured taxi routes, b) spent more time on frequency than the track-up surface map group during the structured taxi condition, and c) spent more time under the authority of ground control than the other groups regardless of their assigned taxi route (structured or unstructured).

Of the 68 taxi routes completed, 54% contained one or more operational concern or problem — with more overall problems in the north-up surface map group during the structured taxi routes, when compared with the north-up and track-up surface map groups during unstructured taxi routes. Notably, all of the structured and 21% of the unstructured taxi routes that involved the use of a north-up display were problematic. The number of problematic routes was equivalent when pilots navigated their assigned structured or unstructured taxi routes using paper-charts or track-up surface-map displays.

The results of the Wickens, Liang, Prevett, and Olmos (1996) simulation study, which investigated pilot use of either a rotating or fixed map display, reported that access to a north-up display was not advantageous for pilots flying southerly headings — any changes to their flight path would require complex mental rotations. Their interpretation correlates well to the findings

reported here in that about 75% of the identified operational concerns involved spatial awareness. The north-up map group experienced more problems that related to missing a turn or attempting to rejoin a taxi route while problems for the paper-chart group took the form of getting lost; for the track-up map group, it was not being on their assigned route.

It would seem that having a north-up map display for airport surface navigation provided no additional benefit over a paper chart. Furthermore, pilots in the paper-chart and north-up map groups may have been busier performing complex mental operations (i.e., making left to right transformations) while navigating. Pilots in these map groups requested more repetitions and were more likely to incorrectly read back messages. Consequently, fewer attentional resources may have been available to actively listen for their aircrafts' call signs.

This is similar to driving on an unfamiliar metropolitan interstate highway requiring extra vigilance to make a timely and correct turn when approaching a fast-moving, cloverleaf intersection. If a passenger should attempt to engage the driver in casual conversation, the driver might miss the turn or not hear the passenger. The driver and pilot alike can request a "say again" or ask for assistance – both requiring additional communications.

Although some pilots (like drivers) turn their maps to be congruent with the direction they are going, Joseph et al. (2002) did not mention whether the pilots in the paper-chart group had rotated their maps or not. If they had, their performance should have been more aligned with the pilots in the track-up group instead of the north-up group.

When evaluating emerging avionics devices that aid navigation, consideration as to the format of these displays must be deliberated in light of the piloting task the operator is expected to perform (see Aretz, 1991, for a summary of previous research; Carel, McGarth, Hersherber, & Herman, 1974, for early research on design criteria). The format in which a map is presented can, and does, affect some aspects of pilot performance — north-up displays are better for some tasks (planning), and track-up displays are better at others (turning). In fact, Clarke, McCauley, Sharkey, Dingus, and Lee (1996) suggest that, when both north-up and track-up displays are available, pilots typically select north-up map displays when planning routes and track-up display when flying. Some developers are making both north-up and track-up modes available on some of their CDTI devices, and this provides the pilot with the option to select one mode for some piloting tasks and the other mode for others. Of course, some pilots still may choose to use paper charts as their primary source of airport information.

REFERENCE

Aretz, A. (1991). The design of electronic map displays. *Human Factors,* **33**:85-101.

Carel, W.L., McGarth, J.J., Hershberger, M.L. and Herman, J.A. (1974). (DOD-74-731101). *Design criteria for airborne map displays volume I: Methodology and research results.* Washington DC: Department of Defense, Department of the Navy, Office of Naval Research.

Clarke, D.L., McCauley, M.E., Dingus, T.A., and Lee, J.D. (1996). Development of human factors guidelines for advanced traveler information systems and commercial vehicle operations: Comparable systems analysis. Washington, DC: Federal Highway Administration (FHWA-RD-95-197).

Federal Aviation Administration. (November 2002). *Order 7050.1 Runwa Safety Program;* http://www.awp.faa.gov.

Federal Aviation Administration. (July 2002). *Runwa Safety Blueprint 2002-2004;* http://www.faa.gov/runwaysafety/publications.cfm.

Federal Aviation Administration. (2001). *Operational Evaluation-2 Final Report;* http://www1.faa.gov/And/AND500/docmgr/docs/T0247.pdf.

Federal Aviation Administration (2000). *Air Traffic Control* (7110.65M). Washington DC: U.S. Government Printing Office.

Federal Aviation Administration (1998). *Airport Surface Operations Safety Action Plan;* http://www1.faa.gov/ats/ato/150_docs/RSP_AP.DOC.

Joseph, K.M., Domino, D. Battiste, V., Bone, R., and Olmos, O. (2003). *A summary of flightdeck observer data from SafeFlight 21 OpEval-2.* (DOT/FAA/AM-03/2). Washington, DC: Federal Aviation Administration, Office of Aerospace Medicine.

National Transportation Safety Board. (July 2000). *NTSB Safety Recommendation A-00-66;* http://www.ntsb.gov/Recs/letters.

National Transportation Safety Board. (1986). *Runway Incursions at Controlled Airports in the United States.* NTIS Report PB86-917003.

Prinzo, O.V. (1996). *An analysis of approach control/pilot voice communications.* (DOT/FAA/AM-96/26). Washington, DC: Federal Aviation Administration, Office of Aerospace Medicine.

Wickens, C.D., Liang, C., Prevett, T., and Olmos, O. (1996). Electronic maps for terminal area navigation: Effects of frame of reference and dimensionality. *The International Journal of Aviation Psychology,* **6:** 241-71.

This page intentionally left blank

TASK PATTERNS FOR TAKING INTO ACCOUNT IN AN EFFICIENT AND SYSTEMATIC WAY BOTH STANDARD AND ERRONEOUS USER BEHAVIOURS

Philippe Palanque & Sandra Basnyat
LIIHS-IRIT, University of Toulouse 3, 118, route de Narbonne, 31062 Toulouse cedex, France

Abstract: While designing interactive software, the use of a formal specification technique is of great help because it provides non-ambiguous, complete and concise notations. The advantages of using such a formalism is widened if it is provided by formal analysis techniques that allow to prove properties about the design, thus giving an early verification to the designer before the application is actually implemented. However, formal specification of interactive systems (even though aiming to produce reliable software) often does not address the issues of erroneous user behaviour. This paper tackles the problem by proposing a systematic way of dealing with erroneous user behaviour. We propose to extend task models (describing standard user behaviour) with erroneous user behaviour. Without appropriate support, incorporating erroneous user behaviour in task models requires much effort from task analysts. We thus propose the definition of patterns of user errors in task models. These patterns of errors are then systematically applied to a task model in order to build a task model covering both standard and erroneous user behaviour. These task models can then be exploited towards system models to provide a systematic way of assessing both system compliance to user tasks and system tolerance to user errors.

Key words: Formal specification, human error, UI design, Petri nets, tasks models

1. INTRODUCTION

While designing interactive software, the use of a formal description technique is of great help by providing non-ambiguous, complete and concise notations. The advantages of using such a notation is widened if it is provided by formal analysis techniques that allow proving properties about the design, thus giving an early verification to the designer before the application is actually implemented. However, formal description of an interactive system (even though aiming at producing reliable interactive software) often do not address the issue of erroneous user behaviour. Indeed, the focus is mainly on describing 'normal' behaviour of users while difficulties mainly arise due to unexpected or invalid (according to the system's description of valid behaviour) user actions performed.

This paper reports work done in combining issues relating to formal description techniques for interactive systems and issues raised by techniques for human error analysis and categorisation. The basic idea is to bring together in a unifying framework, three aspects of the user-centred development process for reliable interactive software i.e. task analysis and modelling, formal description of the system and human error categorisation.

Task analysis and modelling approaches have always focussed on standard behaviour of users leaving user error analysis for later phases in the design processes (Barber & Stanton 2004). This is part of the rationale underlying task analysis which is to provide an *exhaustive* analysis of user behaviour. This comprehensivity (of analysis) is critical as it is meant to provide the basics for a global understanding of user behaviour and tasks that will serve as a basis for driving evolutions of the interactive system. However, practice shows that reaching this exhaustivity is very difficult in terms of availability of resources and economy. These aspects drive people responsible for task analysis and modelling to focus on most frequent and standard activities, thus leaving the infrequent or erroneous ones unconsidered. However, this is precisely where the emphasis should be placed in order to deal efficiently with error tolerance.

In this paper we propose to use task patterns as a way of dealing exhaustively with potential user errors. These patterns of tasks have been modelled and can be directly reused within a task model in order to represent potential deviations of user behaviour. These task models (including representations of possible user errors) can then be tested over a system model in order to verify that the system under development is able to tolerate such erroneous user behaviour. The principle is similar to the work presented in (Fields et al.,, 1999) in terms of objective. However, in this paper we focus not only on ways of identifying erroneous user behaviour, but also on

representing such behaviour and on integrating them with standard user behaviour.

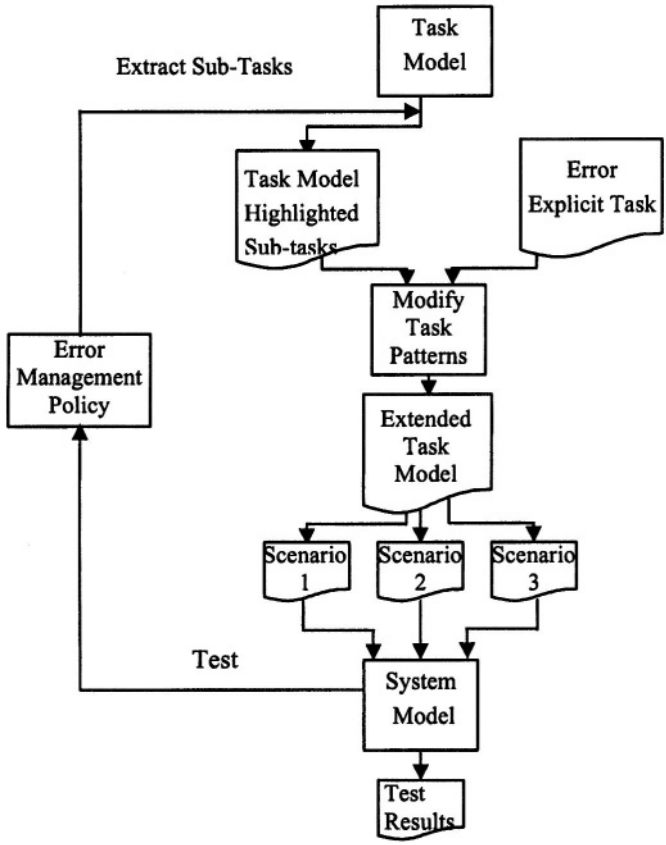

Figure 1. Overview of the research

The paper is structured as follows. Section 2 presents the domain of task analysis and modelling. It presents related work in this field and introduces the notion of task patterns as shown above. Section 3 deals with human error issues. It presents a subset of an extensive classification we have defined in order to identify the set of possible erroneous user behaviour that could occur while interacting with an interactive system. The classification builds upon previous work in the field of human error analysis and is then combined with the task patterns presented in section 2. Section 4 shows, on a case study, how this framework can be used and what it brings to the design and verification of error-tolerant safety critical interactive systems. Section 5 briefly presents some ways for relating this work on task modelling to work on the system side modelling.

2. TASK MODELLING

Tasks analysis and modelling is a central element to user centred design approaches. For this reason a lot of work has been devoted to it and to its integration in the development process of interactive systems. This section first presents an informal presentation of tasks modelling. A summary of task modelling techniques is then provided followed by a third section dealing with previous work addressing the combination of user error and task modelling. Finally, we present the notion of patterns and how this notation can be applied to task modelling.

2.1 Informal Presentation

A task model is a representation of user tasks often involving some form of interaction with a system influenced by its contextual environment. We use the word "influenced" (as opposed to "driven" by the environment) to highlight our thoughts on user's having an underlying goal and hence plan in their mind before attempting to perform a task. This contrasts Suchman's 1987 proposal that plans are representations of situated actions produced in the course of action and therefore they become resources for the work rather than they in any strong sense determine its course.

Users perform tasks, which are structured sets of activities (Preece 1994) in order to achieve higher-level goals. Tasks can be further decomposed corresponding to lower level sub goals. This notion of decomposition naturally results in tree-like structures and thus hierarchical representation of the model. More recently, task modelling tools such as Paternò's ConcurrentTaskTrees CTT (Paternò 1999) have enabled the distinction of abstract, user, interaction and application tasks as well as the possibility to model temporal aspects of activities when specifying a model. The CTT notation is described further in this section. The typical characteristics of a task model include its hierarchical structure, the task decomposition and sometimes the temporal relationship between elements.

Table 1. Summary of Task Analysis Techniques

Acronym	Full Name
HTA	Hierarchical Task Analysis (Annett & Duncan, 1967)
TKS	Task Knowledge Structure (Johnson, 1989)
MAD	Méthode Analytique de Description de tâches (Scapin et al.,, 1989)
UAN	User Action Notation (Hix & Hartson 1993)
GTA	GroupWare Task Analysis (van der Veer, 1996)
CTT	ConcurTaskTrees (Paternò, 1999)
GOMS	Goals, Operators, Methods and Selection rules (Baumeister, 2000)

The above mentioned models differ in their type of syntax (textual vs graphical), the level of formality and the richness of operators offered to designers. The work presented here could fit any of these notations even though our focus is on CTT (because of its tool support). More recent approaches to task modelling have emphasised the context in which the interaction between human and system is taking place with respect to user characteristics, organisations, artefacts to be manipulated and actions required. This has resulted in many task models having hybrid conceptual frameworks.

2.2 CTT and why it is not enough

ConcurTaskTrees (CTT) is a graphical notation used for specifying task models of cooperative applications in a hierarchical structure while also denoting temporal operators. The task models can then be simulated to study different possible paths of interaction. The notation is based upon four types of tasks, abstract tasks, user tasks, application tasks and interaction tasks as shown in Table 2.

Table 2. CTT Tasks

Graphical Symbols	Description
	Abstract Tasks: Tasks that require complex activities whose performance cannot be univocally allocated.
	User Tasks: Usually they are important cognitive activities.
	Application Tasks: Can supply information to the user.
	Interaction Tasks: Between the user and the system.

The operators provided within the CTT notation are described in Table 3. Figure 2 depicts the "choice" operator in which the interactive "Task" is performed by either interactive task 1 or interactive task 2. Figure 2 illustrates the use of an "abstract" task which is performed firstly by a user task, perhaps cognitive, which enables the interactive task.

Figure 4 is a simple example using the CTT notation taken from the ConcurTaskTree Environment (CTTe) tool for accessing an ATM with emphasis and expansion on the withdraw cash task.

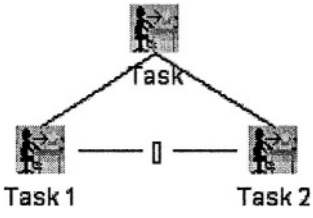

Figure 2. Choice Operator

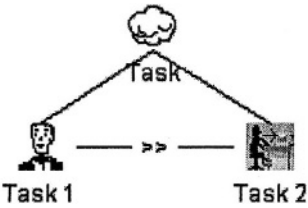

Figure 3. Enabling Operator with varied task type

Table 3. Operators used in the CTT notation

Notation	Description
T1 >> T2	Enabling
T1 [ ]>> T2	Enabling with information processing
T1 [> T2	Deactivation
T1 [] T2	Choice
T1 *	Iteration
T1 [I] T2	Concurrency with information exchange
T1 |> T2	Suspend resume
T1 ||| T2	Independent concurrency
T1 (n)	Finite iteration
[T1]	Optional task

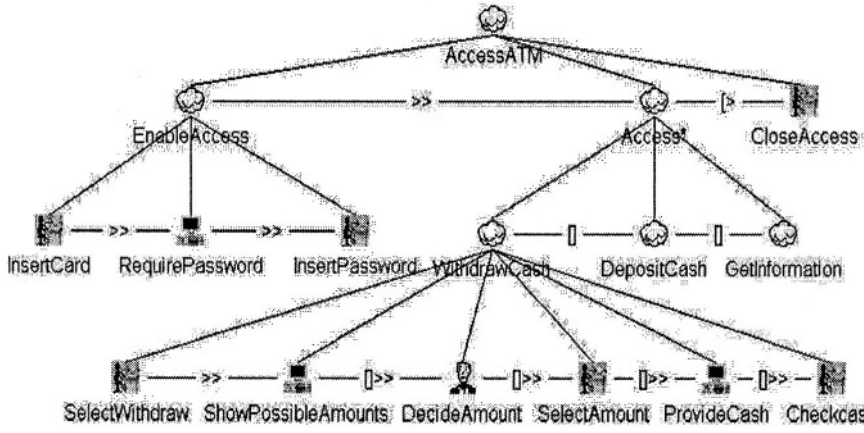

Figure 4. CTT ATM Example provided within CTTe

Figure 4 shows, at a level of description of most task analysis methods, with addition of temporal operators, the task of accessing an ATM. The diagram is read from left to right. A task is not complete until its necessary sub-tasks are addressed. Thus in the above example, EnableAccess must

first be completed by performing of its subtasks. That is, the user inserts the card (an interactive task) which enables the request of the PIN (a system task) which in turn enables the entering of the PIN (a interactive user task). The success of the EnableAccess task enables the Access task which is composed of WithdrawCash or DepositCash or GetInformation. Please note however, that the Access task is iterative which means that each of the Access subtasks can be performed in any order. The iteration of the Access task is aborted by the interruption of the CloseAccess task.

We are considering CTT as the most appropriate notation for this research because of its graphical appearance and tool support.

2.2.1 Why CTT is not enough

There are also downsides to the CTT notation. The context and environmental conditions within which the activities are taking place are not considered when modelling tasks in CTT. Surrounding circumstances could have effects on the process. Also, details of artefacts being manipulated during tasks could be useful because the time spent performing a task could be dependent on the artefact in hand. The cognitive workload and users' current state is also not detailed. For example, conditions such as stress or tiredness or being under pressure may effect the way in which actions are performed with respect to their efficiency and effectiveness. CTT does not allow the designer to detail the type of low-level interaction taking place such as a mouse click or a keyboard entry. Furthermore, the "shift of focus" of the user when interacting with an interface is not taken into account. This level of detail could be crucial in safety-critical interactive systems.

2.3 Task modelling and user errors

As stated in the introduction, when modelling user behaviour, an error-free perspective is usually employed. It is normally during the testing phase of the system development cycle that errors are realised and taken into account. Task modelling as yet, does not allow for the description, representation and analysis of unexpected eventualities that may occur including human error. Since the task model influences the system design it is important to understand how to manage and overcome possible errors.

In their paper, Baber and Stanton (Barber & Stanton 2004) propose that a system image implies a set of routines that a person may use and that the selected routine will be based on the user's goal and previous experiences. They suggest a need to represent interaction between user and product to consider possible mental model mismatches between a given system image

and user representation during initial design activity. The proposed technique, Task Analysis for Error Identification (TAFEI) is based on the assumption that interaction is goal-oriented and passes through a series of states. The TAFEI approach consists of three stages, a Hierarchical Task Analysis (although any technique is acceptable), construction of a State Space Diagram (SSD) mapped with the HTA plans, finally construction of a transition matrix to display state transitions during device use. This work tries to address the same goal as ours. However, there are three main differences:

- Our work, exploiting classification of user error provides a systematic way of dealing with possible user errors,
- Our proposal to define and exploit task patterns for user errors provides a way of coping in an efficient and reliable way with the complexity of the task models,
- The authors consider as erroneous only the paths in a users task that are provided by the system but do not support the achievement of the user's goal. We consider user error in a broader sense including errors such as mistakes and slips.

Paternò and Santoro (2002) also suggest that a goal is a desired modification of the state of an application. Their work describes how task models can be used in an inspection based usability evaluation for interactive safety-critical applications.

A key aim to their method is to provide designers with help in order to systematically analyse what happens if there are deviations in task performance with respect to what was originally planned during the system design. Building upon the HAZAOP family of techniques a set of predefined classes of deviations are identified by guidewords such as "none", "other than" and "ill-timed".

The method incorporates three stages:

1) Development of the task model for the application considered.
2) Analysis of deviations related to the basic tasks.
3) Analysis of deviations for high-level tasks.

The analysis, which is recommended be carried out by interdisciplinary groups, follows a bottom-up approach considering basic and then high-level tasks. Documentation of the analysis takes the form of tables. As we suggest in this paper, (Paternò & Santoro 2002) mention that the interpretation of every guideword for every task of the prototype addresses completeness issues however has the drawback of taking time and effort.

This work tries to address the same goal as ours. However, there are two main differences:

- Our proposal to define and exploit task patterns for user errors provides a way of coping in an efficient and reliable way with the complexity of the task models,
- The authors consider error analysis based only on a set of high-level guidewords (such as "other-than") and thus leaving detailed analysis to the analyst's discretion. Our proposal is much more concrete and grounded on previous work in the field of user error identification and classification. For instance we have more than 20 types of errors that would fit within the "other-than" guideword (for instance "branching error", "environmental capture", "order error"…).

The Technique for Human Error Assessment or THEA (Pocock et al., 2001) is aimed at helping designers of interactive systems anticipate human errors resulting in interaction failures. As we suggest, this is also a technique intended for use during the initial phases of the development lifecycle. With its foundations laying in human-reliability analysis (HRA) (Kirwan 1994) it aims to establish requirements for "error resilient" system design. In their paper, it is noted that errors can be regarded as failures in cognitive processing (Pocock et al., 2001). The process of analysing a system's vulnerability to human error is performed by posing provided questions about a given scenario, identifying possible causal factors of potential problems identified and finally identifying consequences and their impact on task, work, user, system, etc. The results are recorded in tables for further analysis.

The THEA approach shares a common perspective with our work, that is, we aim to assist designers to produce usable error tolerant interactive systems based on systematic techniques. However, our work differs in that we intend the task patterns to be applicable to more than one system design, possibly of various domains. This means that once a sound solution has been modelled, there will be less repetition of work.

2.4 Task patterns

Task patterns for interactive systems design is a relatively new concept with the aim of solving design problems using existing knowledge of previously identified patterns and solutions. The majority of research to date focuses on user interaction with software and interfaces providing interface design patterns and task based patterns to improve usability rather than task

based patterns that focus on user behaviour intended for testing the compatibility with system design.

Indeed, the focus is mostly on what we could call generic user behaviours. This work proposes then a set of user interface design solutions to such user behaviours. Work presented in (Sinnig et al., 2003) focuses on establishing and integrating patterns as building blocks for the creation of the task model in order to merge software and usability engineering. Work presented in (van Welie et al., 2000) also addresses the issues raised by user interaction stating that patterns for user interface design should help make systems more usable for humans.

Task patterns were first introduced by Paternò (Breedvelt et al., 1997 & Paternò 1999) as reusable structures for task models. The patterns were described as hierarchical structured task fragments that can be reused to successively build the task model.

(Sinnig et al., 2004) present a model-based approach to development in which they refer to the task, user, business, object, dialogue, presentation and layout models. In (Sinnig et al., 2003) they describe two types of patterns: task and feature. Task, which detail activities the user has to perform while pursuing a certain goal and feature patterns that are applied to the user-task model describing the activities the user has to perform using a particular feature of the system. Task patterns are said to be composed of sub-patterns which can be task or feature patterns. A four-sage strategy for the process of pattern application is also detailed. Steps include Identification, Selection, Adaptation and Integration.

Such task patterns are context specific and problem centred as opposed to guidelines which can often be too simplistic, too abstract and difficult to interpret. The task patterns proposed in these works only model error-free behaviour, that is, the possibility of human-error would be considered later in the design lifecycle during the testing phase of the system for example.

3. USER ERRORS

Human error plays a major role in the occurrence of accidents in safety-critical systems such as in aviation, railways systems, or nuclear power plants (Reason 1990). The recent rail disaster of North Korea has been officially blamed by their Government on human error. (BBC News, 2004).

Interactive systems particularly those that are safety-critical need to be designed with the eventuality of human error in mind to prevent catastrophes. This means, early in the design process and as well as during the testing phase. Although the term “human error” appears very

controversial, theories of human errors such as Rasmussen's (Rasmussen 1983) SRK, Hollnagel's (Hollnagel 1991) Phenotypes and Genotypes and Norman's (Norman 1988) classification of slips can be considered widely acceptable.

Table 4 presents a subset of the Skill-based Human Error Reference Tables (we will refer to as HERT from this point onwards). Within Skill-based errors are two main failure modes, inattention and over attention (Reason 1990). This table details those errors found within the inattention failure mode.

3.1 Classification of user errors

Hollnagel's (Hollnagel 1991) error classification scheme is an example of a behaviour-based taxonomy of human error (Reason 1990). Hollnagel identifies eight simple phenotypes and five complex phenotypes. It starts with the observable phenomena, such as errors of omission or commission, rather than cognitive theories. The observable phenomena, he states, make up the empirical basis for error classification. He refers to these behavioural descriptions as the *phenotype* of human error. The error *genotype* denotes the mental mechanism assumed to underlie the observable erroneous behaviour.

Rasmussen (Rasmussen 1983) proposed the SRK theory in which he distinguished three levels of human processing each with its associated error types: (1) skill based level, for activities performed automatically, (2) rule based level, for circumstances in which our intuition provides an applicable response, and (3) the knowledge based level, for new situations in which there are no rules.

–

Based on Rasmussen's SRK theory, Reason developed the Generic Error Modelling System (GEMS) (Reason 1990) which can be summarised as:

- SB: Unintended deviations from the procedures that are conducted automatically by the individual.
- RB: Associated with those activities in which the individual has to consciously choose between alternative courses of action.
- KB: The individual attempts to define a new procedure on the basis of knowledge about the system they are using.
-

Using the above-mentioned classifications, we have produced detailed tables, grouping together many user oriented error types appropriate for our studies based fundamentally on the SRK theory.

The tables are decomposed at the following levels: (Please note part of sections 1.3, 1.4 and 1.5 are shown in Table 4)

1 Rasmussen's Skill-based level
  - 1.1 Rasmussen's SRK
  - 1.2 Reason's failure modes (inattention & over attention)
  - 1.3 Reason's 6 common mechanisms
  - 1.4 Error name & definition
  - 1.5 Example of error

2 Rasmussen's Rule-based level
  - 2.1 Reason's failure modes (misapplication of good rules & application of bad rules)
  - 2.2 Reason's 9 common mechanisms
  - 2.3 Error name & definition
  - 2.4 Example of error

3 Rasmussen's Knowledge-based level
  - 3.1 Reason's failure modes (selectivity, workspace limitations etc)
  - 3.2 Error name & definition
  - 3.3 Example of error

Some of the examples identified in the HERTs are those presented by authors of the classifications, others have been supplemented. Within the framework, further classifications are identified such as Hollnagel's eight simple phenotypes and five complex phenotypes (Hollnagel 1991), Norman's six categories of slips (Norman 1988), HAZOP causes of deviations (HAZOP 1989) etc., which were not necessarily located together within the tables. The benefit of producing such reference tables enables the exact identification of very precise error types when analysing human behaviour associated to particular tasks of a task model.

3.2 Relating types of user error to task patterns

Since the HERTS have been decomposed to the level of an example for each type of error, it is possible to relate every classified error to a particular task. Thus for each task of a task analysis model, it is possible to determine, by means of elimination, which human errors are applicable.

Once specific error-related situations for each task of a task model have been identified, it is possible to re-design the model to support and make explicit these types of errors. At this stage, if there is an existing system in place, its constraints will need to be considered. This ensures, that when the system or future system is modelled, for example as a Petri-net, the task model can be tested over the system model with the human errors already

identified. This would avoid the discovery of problems during later testing phases of the development lifecycle.

Finally, the re-designed error explicit parts of the task model can be 'plugged in' to the relevant areas of the task model creating task patterns which can be applied to other domains involving the same or similar activities.

Table 4. **Subset of the Skill-Based Inattention Human Error Reference Tables**

Reason's 6 Common Mechanisms	Error Name & Definition	Examples of errors	Error Type Ref
Double-capture slips Involve two distinct, though casually related, kinds of capture. (Reason 1990) (Norman's 1989 1st category of slips)	Strong-habit intrusion The unintended activation of the strongest action scheme beyond the choice point. (Reason 1990) Hollnagel 1993, simple phenotype, Intrusion. Error mode: action not included in current plans. Corresponding complex phenotype, Hollnagel, 1993 capture, branching and overshoot.	Unintended activation of the strongest action schema beyond the choice point. (Reason 1990) On starting a letter to a friend, I headed the paper with my previous home address instead of my new one. (Reason 1990)	Reason 1990
	Strong-habit capture (Reason 1990)	I intended to stop on the way to work to buy some shoes, but 'woke up' to find that I had driven right past. (Reason 1990)	Reason 1990

4. CASE STUDY (CASH MACHINE)

In order to demonstrate our ideas, we take the example of an Automated Teller Machine (ATM). ATMs have often been used to demonstrate task analysis since they are widely used systems demonstrating clear human-computer interaction. We are using the ATM as an example because we are focusing on repetitive, highly structured, situated based systems involving less decision making, however errors can still occur.

4.1 Task Models

Since the possible interactions with an ATM may sometimes be complex and plentiful, we have focused our attention on producing a task model for the process of withdrawing cash using the CTT notation (See Figure 5).

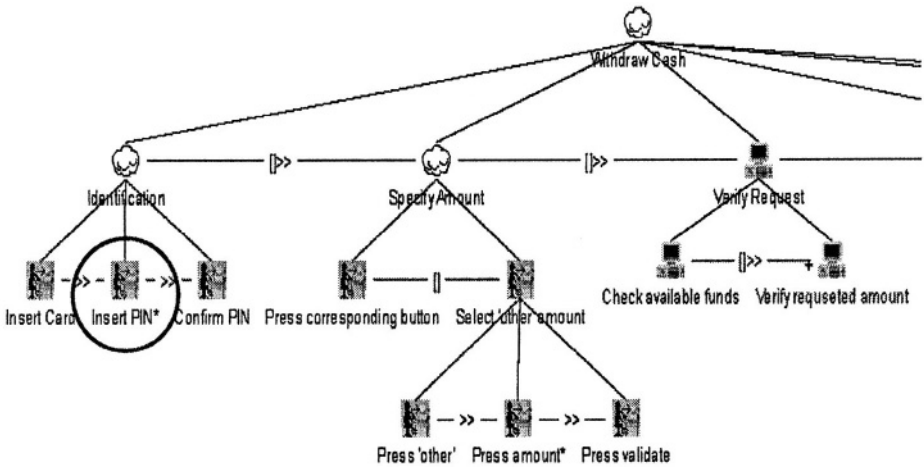

Figure 5. Sub-section of CTT Task Analysis for Withdrawing Cash at an ATM (Highlighting the "Insert PIN" activity for analysis during the case study)

The CTT diagram for cash withdrawal at an ATM shows part of the process of being identified, selecting an amount through to the withdrawal of the money. We are referring to this CTT as a general representation of task analysis because it describes error free behaviour and "normal" flow of activities. For example, if the same task was analysed using the HTA approach, the resulting textual or graphical representation would be somewhat similar. However CTT allows for temporal operators to be described, which is particularly useful for modelling interaction with an ATM, resulting in a more realistic model in terms of possible user activities.

4.2 Identification of possible deviations

Following the task analysis and modelling, each low-level sub-task can been considered for possible human error events that may occur. This can be done by referring to the HERTs and determining whether or not each particular type of error could occur to each CTT task. In this paper, we present the systematic analysis of possible human deviations while interacting with an ATM based on the "insert PIN" sub-task of Identification highlighted in Figure 5 and the Skill-Based HERTs (a subset of which is

shown Table 4). Where applicable, examples of human deviation relevant to the Insert PIN activity have been identified. This can be seen in Figure 6.

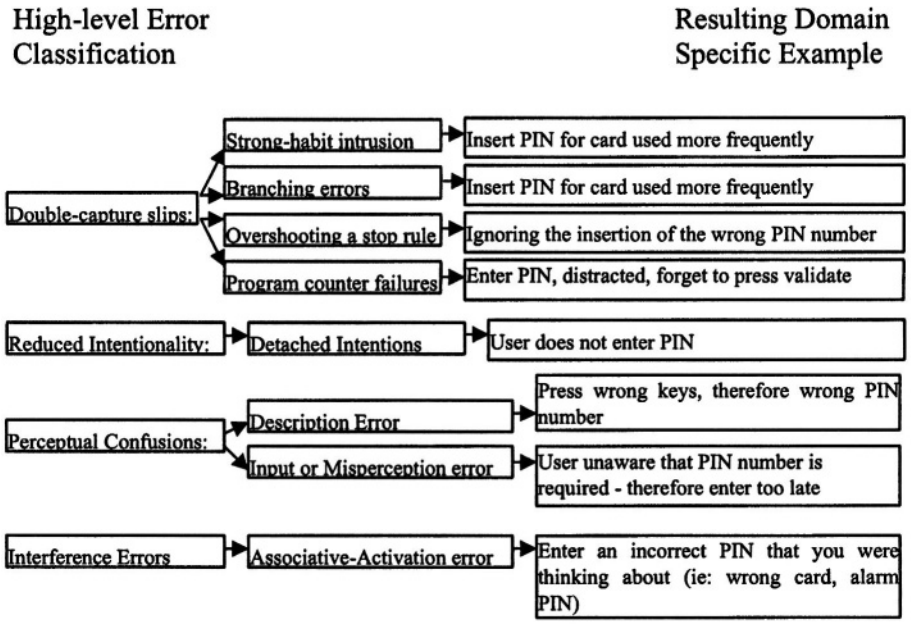

Figure 6. Subset of the systematic analysis of possible deviations while inserting a pin at an ATM based on a CTT task analysis model and skill-based human error classification for the inattention failure mode

It must be noted however, that although many possibilities have been considered, we have sensibly limited our analysis to avoid considering natural disasters such as "dropping down dead" at the cash machine.

4.3 Including task patterns for errors

Some of the errors identified in Figure 6 above can be summarised as they result in the same task modelling. For example, 'Insert PIN for card used more frequently' and 'Press wrong keys, therefore wrong PIN number' can both be considered as entering the wrong PIN. Figure 7 to Figure 17 illustrate the re-designed task models for the Insert PIN sub-task which make explicit the possible deviations. The task models contain repetitions which can now be re-labelled as task patterns.

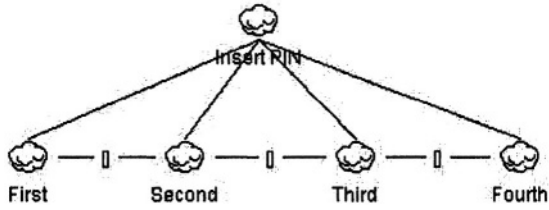

Figure 7. Four possibilities of interaction when entering PIN

Taking into account the potential errors identified, there are four possibilities of interaction with the system when entering the PIN shown as four abstract tasks, first, second, third and forth in Figure 7. The abstract tasks are made up of combinations of five task patterns labelled P1, P2, P3, P4 and P5. These patterns can be seen in Figures 8 – 12.

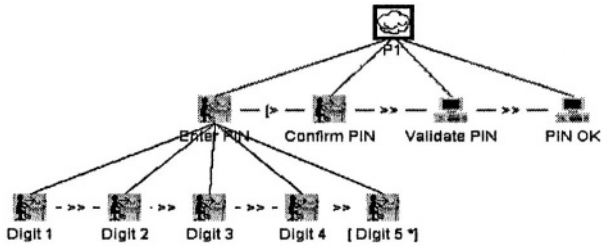

Figure 8. Pattern 1: PIN OK

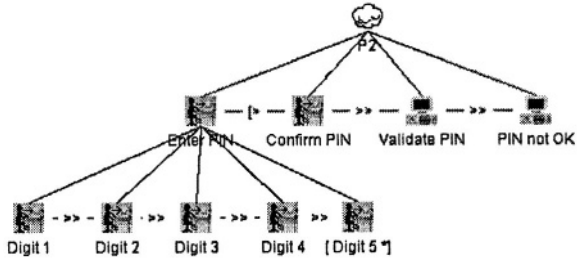

Figure 9. Pattern 2: PIN not OK

–

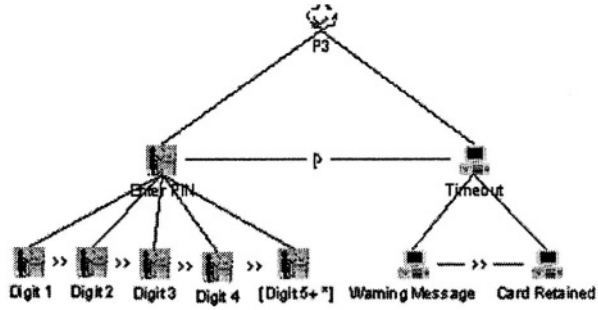

Figure 10. Pattern 3: Too long entering PIN

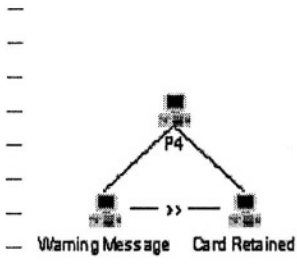

Figure 11. Pattern 4: Simple Timeout

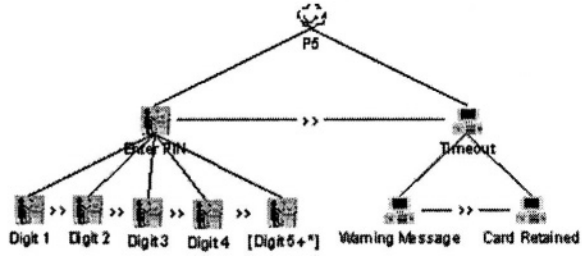

Figure 12. Pattern 5: Timeout on PIN confirmation

1) For the first possibility of interaction (refer to Figure 7), either:

- The PIN is correct on the 1st try OR
- The user takes too long to enter the PIN and the system times out OR
- The user does not enter a PIN at all and the system times out OR
- The user takes too long confirming the PIN and the system times out

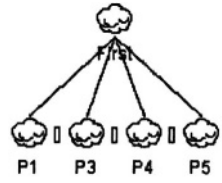

Figure 13. [First possible interaction]

2) For the second possibility of interaction (refer to Figure 7), either:

- The PIN is incorrect on the 1st try and correct on the 2nd try OR
- The PIN is incorrect on the 1st try and the user takes too long to enter the PIN on the 2nd try therefore the system times out OR
- The PIN is incorrect on the 1st try and the user does not enter a PIN at all on the 2nd try and therefore system times out OR
- The PIN is incorrect on the 1st try and the user takes too long confirming the PIN on the 2nd try and the system times out

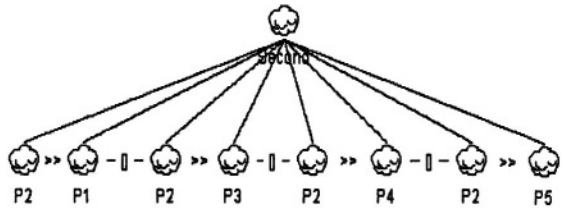

Figure 14. Second possible interaction

4) For the third possibility of interaction (refer to Figure 7), either:

- The PIN is incorrect on the 1st try, incorrect on the 2nd try and correct on the 3rd try OR
- The PIN is incorrect on the 1st try, incorrect on the 2nd try and the user takes too long to enter the PIN on the 3rd try therefore the system times out OR
- The PIN is incorrect on the 1st try, incorrect on the 2nd try and the user does not enter a PIN at all on the 3rd try and therefore system times out OR
- The PIN is incorrect on the 1st try, incorrect on the 2nd try and the user takes too long confirming the PIN on the 3rd try and the system times out.

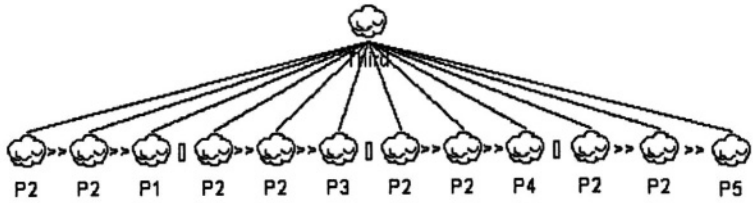

Figure 15. Third possible interaction

5) For the fourth possibility of interaction (refer to Figure 7):

The PIN is incorrect on the 1st try, incorrect on the 2nd try and incorrect on the 3rd try

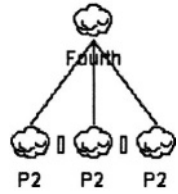

Figure 16. Fourth possible interaction

It can be seen from previous models that, taking into account possible erroneous behaviour increases significantly the size of the task models. For instance there are 21 leaves in the full task model for normal user behaviour (some of which is represented in Figure 5) and 234 leaves in the task model describing both erroneous and normal user behaviour (merging of models illustrated in Figures 7 to 17).

5. TOWARDS ERROR-TOLERANT SYSTEMS

This paper has presented the process of transforming a simple task analysis model to a more complex one making domain specific human errors explicit by means of task patterns that could be applied to other domains.

It is intended that the idea of identifying human errors early in the development process will enable the design of error-tolerant systems. We have previously studied different ways of taking into account task models in interactive systems development.

For space reasons, we only present in this section some ongoing work we are carrying out to exploit the results presented in this paper.

The process of relating task models and system models extends previous work presented in (Navarre et al., 2001) where task models are used in combination with system models in order to verify that both models were compliant with each other.

In this work we were only considering error-free user behaviour and were able to prove compatibility of tasks and systems at lexical, syntactic and semantic levels. More information about that can be found in (Palanque et al., 1997) and in (Palanque & Bastide 1997). Such verification allows designers to assess that all the tasks in the task model correspond to actions offered by the system (and represented in the system model). Similarly, the sequence of tasks in the task model must be compatible with the valid sequence in the system model.

6. CONCLUSION

In this paper we have presented a way of taking into account in a systematic way erroneous user behaviour. This work builds upon previous work in the field of task analysis, task modelling, human error analysis and identification. We have proposed the definition and use of task patterns for dealing with complexity and repetitions that frequently appear when modelling erroneous user behaviours. We have shown on a simple case study how task patterns have been identified and how we have modelled them using CTT notation and its related tool CTTE. Due to their intrinsic nature, patterns are good candidates for known and previously encountered problems. This is the reason why they have been successfully exploited on the ATM case study. Their application in other context where interaction techniques are more innovative has still to be studied. In the same way as we have studied the use of patterns for CUA interactors, we are currently working on their application for ARINC 661 user interface standard in cockpit displays in order to provide certification authorities in France DPAC (Direction des Programmes de l'Aviation Civile) with systematic error identification techniques.

ACKNOWLEDGMENTS

The work presented in the paper is partly funded by French DGA under contract #00.70.624.00.470.75.96 and EU via the ADVISES Research Training Network RTN2-2001-00053.

REFERENCES

Alexander, C., Ishikawa, S and Silverstein, M. (1977) A pattern language: towns, buildings, construction.

Annett, J. and Duncan, K., (1967) Task Analysis and Training Design, Occupational Psychology, 41,1967, pp.211-227.

ARINC 661. (2002) Cockpit display system interfaces to user systems. Arinc specification 661. Published: April 22, 2002. An ARINC document prepared by airlines electronic engineering committee Published by aeronautical radio, inc. 2551 riva road, annapolis, maryland 21401

Baber, C., and Stanton, N. (2004). Task Analysis for Error Identification. In D. Diaper & N. Stanton (Eds.) The Handbook of Task Analysis for Human-Computer Interaction. New Jersey: Lawrence Erlbaum Associates p.367-379

Baumeister, L.K., John, B.E., Byrne, M.D. (2000). A Comparison of Tools for Building GOMS Models Tools for Design. In: Proc. of ACM Conf. on Human Factors in Computing Systems CHI'2000, ACM Press, New York, 502–509

BBC News Website (2004) http://news.bbc.co.uk/2/hi/asia-pacific/3656853.stm (last accessed 30th April 2004)

Blandford A. (2000). Designing to avoid post-completion errors. PUMA working paper WP33. (http://www.cs.mdx.ac.uk/puma/).

Blandford, A., (2000) PUMA Footprints: linking theory and craft skill in usability evaluation. (http://www.cs.mdx.ac.uk/puma/).PUMA working paper WP26.

Blandford, A. (2000) Designing to avoid post-completion errors. WP33

Breedvelt, I., Paternò, F. & Sereriins, C. (1997). Reusable Structures in Task Models, Proceedings DSVIS '97, Springer Verlag, pp.251-265

Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal. (1996) Pattern Oriented Software Architecture. John Wiley & Sons, Inc.

Card, S.K., Moran, T.P., Newell, A. (1983). The Psychology of Human-Computer Interaction. Lawrence Erlbaum Associates, Hillsdale

Farenc C. & Palanque P. (1999). A Generic Framework based on based on Ergonomic Rules for Computer Aided Design of User Interface. J. Vanderdonckt, A. Puerta (eds.), Computer-Aided Design of User Interfaces II, Proceedings of CADUI'99, Kluwer Academics, Dordrecht, 1999.

Feldbrudge F., Jensen K. (1986). Petri net tool overview in Brauer W. (Editor). LNCS 254 & 255, Springer Verlag

Fields, B., Paternò, F., Santoro, C (1999). Analysing User Deviations in Interactive Safety-Critical Applications, Proc. of DSV-IS '99, pp. 189-204, Springer-Verlag.

Gamma, E., Helm, R., Johnson, R. Vlissides, J. (1995). Design Patterns: Elements of Object-oriented Software. Book published by Addison-Wesley

Hartson, R., and Gray, P. (1992). Temporal Aspects of Tasks in the User Action Through Product and Process. New York. John Wiley.

HAZOP (1996). MOD: Studies on systems containing programmable electronics. UK Ministry of Defence Interim Def Stan 00-58, Issue 1. Available from http://www.dstan.mod.uk/dstan_data/ix-00.htm

Hix, D. and Hartson, H. R. (1993) Developing User Interfaces.

Hollnagel, E., (1991). The Phenotype of Erroneous Actions: Implications for HCI Design. In: Weir, G.R.S. and Alty, J.L., (Eds.), Human-Computer Interaction and Complex Systems, Academic Press.

IBM (1989) Common User Access: Advanced Interface Design Guide. IBM, SC26-4582-0

Johnson, P., Johnson, H. (1989). Knowledge Analysis of Task: Task Analysis and Specification for Human-Computer Systems. In: Downton, A. (ed.): Engineering the Human Computer Interface. McGraw-Hill, Maidenhead 119–144

Jordan, B. (1996). Ethnographic Workplace Studies and CSCW. In D. Shapiro, M.J. Tauber and R. Traunmueller (eds). The design of computer supported cooperative work and groupware systems. North-Holland, Amsterdam, 17-42.

Kirwan, B (1994). A guide to practical human reliability assessment. Taylor and Francis.

Nardi, B. (1995). Context and Consciousness: Activity Theory and Human Computer Interaction. MIT Press, Cambridge MS.

Navarre D., Palanque P., Bastide R. Paternò F. & Santoro C. (2001). A tool suite for integrating task and system models through scenarios. In 8th Eurographics workshop DSV-IS'2001 LNCS, no. 2220. Springer, 2001

Norman Donald A. (1988). The design of everyday things New York: Currency-Doubleday, 1988.

Palanque P, Bastide R. & Paternò F. (1997). Formal Specification as a Tool for Objective Assessment of Safety-Critical Interactive Systems Interact'97 conference, Chapman et Hall. pp. 463-476.

Palanque Ph. & Bastide R. Synergistic modelling of tasks, system and users using formal specification techniques. Interacting With Computers, Academic Press, 9,12, pp. 129-153

Paternò, F. (1999) Model Based Design and Evaluation of Interactive Applications. Springer Verlag, Berlin

Paternò F. and Santoro C. (2002). Preventing user errors by systematic analysis of deviations from the system task model. International Journal Human-Computer Studies, Elsevier Science, Vol.56, N.2, pp. pp. 225-245, 2002.

Pocock, S., Fields, B., Harrison, M and Wright, P. (2001). THEA – A Reference Guide. University of York Computer Science Technical Report 336, 2001.

Rasmussen, J. (1983). Skills, rules, knowledge: Signals, signs, and symbols and other distinctions in human performance models. IEEE Transactions on Systems, Man, and Cybernetics, 13(3):257-267

Reason, J. (1990). Human Error, Cambridge University Press

Smith, D.J., (2002). Human Error (and how to prevent it) e-learning resource. Available from http://www.smithsrisca.demon.co.uk/unitHE2.html

Scapin, D., Pierret-Golbreich, C. (1989) Towards a Method for Task Description: MAD. In: Berlinguet, L., Berthelette, D. (eds.): Proc. of Conf. Work with Display Units WWU'89, Elsevier Science Publishers, Amsterdam (189) 27–34

Sinnig D., Forbrig P. and Seffah A. (2003). Patterns in Model-Based Development, Position Paper in INTERACT 03 Workshop entitled: Software and Usability Cross-Pollination: The Role of Usability Patterns.

Sinnig, D., Gaffar, A., Seffah, A., Forbrig, P. (2004). Patterns, Tools and Models for Interaction Design. MBUI Workshop 2004, P.09

Suchman, L. A. (1987). Plans and situated actions: The problem of human-machine communications. Cambridge, UK: Cambridge University Press.

van der Veer, G., van der Lenting, B.F., Bergevoet, B.A.J. (1996) GTA: Groupware Task Analysis - Modeling Complexity. Acta Psychologica 91 297–322.

van Welie M., van der Veer G.C., Eliëns A. (2000). Patterns as Tools for User Interface Design: In: International Workshop on Tools for Working with Guidelines, pp. 313-324, 7-8 October 2000, Biarritz, France.

A SAMPLING MODEL TO ASCERTAIN AUTOMATION-INDUCED COMPLACENCY IN MULTI-TASK ENVIRONMENTS

Nasrine Bagheri & Greg A. Jamieson
Cognitive Engineering Laboratory, University of Toronto, Ontario, CANADA

Abstract: This article discusses the development of a model that defines the optimal sampling behaviour of operators in a multi-task flight simulation, where one of the tasks is automated. The goal of this model is to assign a cost function to the attention allocation strategy of participants, allowing us to assess the efficiency of their overall strategy. The model revealed that the optimal sampling strategy should be the same regardless of the automation reliability. When applied to previously reported empirical data, the model revealed that participants using constant, highly reliable automation demonstrated more 'expensive' monitoring behaviour, However, their monitoring behaviour became more efficient over time, which is inconsistent with the conclusion that the poor overall monitoring performance was due to complacency. This model allowed us to define an optimal monitoring performance, which is an important step in being able to accurately assess "complacency".

Key words: Human-automation interaction, complacency, sampling strategy

1. INTRODUCTION

Operators of complex systems are often involved in multi-task environments where several information displays compete for their attention. One of the primary cognitive tasks of operators in such systems is managing their allocation of attention so that the displays are sampled at appropriate frequencies to ensure accurate system state identification (Moray & Inagaki, 1999). The attention allocation strategy of operators can have a large influence on the safety and the efficacy of man-machine systems.

There is a wide spectrum of tasks that operators can be asked to perform (e.g., continuous control, discrete control, monitoring, etc.) (Schumacher & Geiser, 1983). As a result of the increased use of automation, monitoring tasks have become prevalent in modern man-machine systems. In this context, an operator's role is to monitor displays to detect abnormal states of the automated system. The high reliability of modern automated systems means that such abnormal states occur only rarely, and it has been suggested that operators may become "complacent" in their monitoring (Wiener, 1981). Complacency has been defined as "self-satisfaction which may result in non-vigilance based on an unjustified assumption of satisfactory system state" (Parasuraman et al., 1993, quoting NASA definition).

Although complacency is considered to be a serious problem, little consensus exists as to how it can be measured (Prinzel et al., 2001). Previous research has concluded that operators were complacent based primarily on their automation failure detection performance over time (e.g., Parasuraman et al., 1993, Singh et al., 1997). Moray (2000, 2003) questioned whether such evidence adequately supports the presence of "complacency" as even optimal behaviour can result in missed signals. Complacency implies under-sampling, not missed signals. This emphasizes the need to assess operators' attention allocation strategies while addressing the issue of complacency.

Bagheri and Jamieson (2004) replicated a Parasuraman et al. (1993) study in which participants who interacted with a consistent and highly reliable automated system were said to show signs of complacency based primarily on detection performance. In both studies, automation reliability was varied as a between-subject factor. The reliability was either constant at a high level (87.5%), constant at a low level (56.25%), or changed every 10 minutes from high (87.5%) to low (56.25%). In addition to detection performance, Bagheri & Jamieson (2004) recorded participants' eye movements to determine whether attention allocation corroborated the conclusions reached by Parasuraman et al. (1993). Results confirmed the significant effect of automation reliability on detection rate; participants using automation of constant and high reliability had the poorest performance. Eye movements revealed that these participants sampled the monitoring task significantly less than participants in the constant low and variable reliability conditions. However, the evolution of their attention allocation patterns did not appear to support the attribution of their poor performance to complacency.

The observational data collected by Bagheri & Jamieson (2004) did not afford a conclusion as to whether participants' sampling behaviour was adequate. To reach such a conclusion, a formal method of evaluating participants' attention allocation strategy is required. Moray and Inagaki (2000) showed that the occurrence of complacency cannot be proved unless an optimal behaviour is specified as a benchmark. This article presents a

model that defines the optimal sampling behaviour based on the characteristics of the tasks that participants had to perform in the study with the goal of concluding whether participants showed 'complacent', 'eutectic' or 'sceptical' monitoring behaviour (Moray, 2003). The eye movement data collected by Bagheri and Jamieson (2004) are reanalyzed to determine which automation reliability condition leads to more effective performance.

2. METHOD

2.1 Apparatus

We present here a model of sampling behaviour for interaction with the Multi-Attribute Task battery used in Parasuraman et al. (1993) (Figure 1). The MAT Battery is a flight simulation that requires participants to perform three equally important tasks: (1) automated system-monitoring, (2) manual tracking, and (3) manual fuel management. The windows containing the information required to perform each task were defined as *lookzones.*

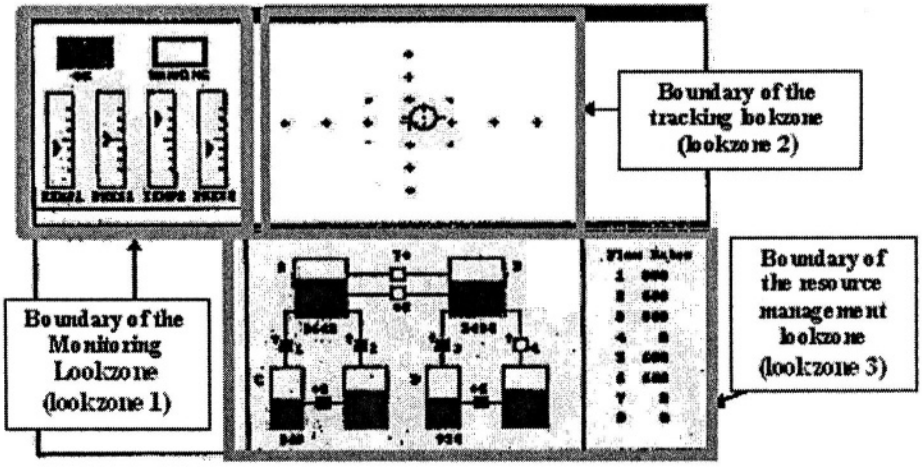

Figure 1. Picture of the MAT battery with the three lookzones of interest highlighted.

The system-monitoring task (lookzone 1) consisted of four engine gauges that participants had to monitor for randomly occurring abnormal values that represented system malfunctions. The monitoring task was automated so that a gauge showing an abnormal value would normally reset itself without participant intervention. Participants were advised that the automated system would sometimes fail to correct these malfunctions. Automation reliability

was defined as the percentage of malfunctions corrected by the automation. In case of an automation failure, participants were required to correct the malfunction manually. If they did not detect the failure within 10 seconds, the pointer was automatically reset and the event was scored as a "miss". Participants were not informed that they missed a failure.

The goal of the tracking task (lookzone 2) was to keep the aircraft within a rectangular area using a joystick (first-order control). The goal of the fuel management task (lookzone 3) was to keep the fuel level in tanks A and B between 2000 and 3000 gallons by controlling a network of eight pumps.

An Eye-gaze Response Interface Computer Aid (ERICA) system was also used to track participants' eye movements in the lookzones defined earlier. The eye-tracker used infrared light technology and was non-invasive to participants. Gaze location samples were taken 30 times per second.

2.2 Model of attention

Our goal was to develop a normative optimal sampling model for interacting with the MAT battery. As participants were advised that the tasks were of equal importance, the model would define the behaviour to optimize performance on the three tasks simultaneously. Our goal was to define a model to assess attention allocation strategy *globally,* as opposed to a local optimization approach wherein each eye movement is considered as a decision that aims at maximizing the marginal utility of the next fixation. For that purpose, we assumed that a functional relationship could be established between participants' average performance and their sampling strategy.

2.2.1 Definition of the problem constraints

There were three lookzones of interest corresponding to the three tasks performed by participants. Let x, y, and z be the sampling rates of the monitoring, tracking, and resource management lookzone, respectively. As a conservative practical approximation, we assumed that only one source of information can be monitored at a time (Moray and Inagaki, 1999). 0.5 sec was established as a lower bound on dwell time in "real-life" tasks for experimented operators (Moray, 1986). Since participants had never used the MAT battery, we assumed a dwell time of 1 sec, and thus an eye movement rate less than or equal to 1 fixation per second. This constraint is expressed below:

$$x + y + z \leq 1 \qquad \textit{Constraint (1)}$$

In this model, each lookzone is allocated a specific elementary cost function which depends only on the associated sampling rate. Hence, we defined $C_m(x)$, $C_t(y)$, $C_r(z)$ as the elementary cost function associated with the monitoring, tracking, and the resource task, respectively. The overall cost is the sum of the contributions of these three independent elementary costs:

$$\Omega(x, y, z) = C_m(x) + C_t(y) + C_r(z)$$

2.2.2 Definition of the cost function requirements

The elementary cost functions must meet three requirements. First, as the three tasks are equally important, missing all of the automation failures must have the same cost as always keeping the aircraft outside the rectangular area, which in turn must have the same cost as keeping the fuel level always beyond the limit. This requirement is expressed as follows:

Requirement (1): $$C_m(0) = C_t(0) = C_r(0)$$

The second requirement implies that detecting all automation failures should be as rewarded as always keeping the aircraft within the rectangular area, or keeping the fuel level within limits at all times. Assuming that given the greatest possible sampling rate of a given lookzone, participants are sure to perform the task perfectly, this constraint is expressed as:

Requirement (2): $$C_m(1) = C_t(1) = C_r(1)$$

Let $C(\eta)$ represent any of the three elementary cost functions. $C(\eta)$ has a direct physical meaning if *Requirements (1)* and *(2)* are translated into C(0) = 1 and C(l) = 0. It represents the percentage of what is missed over what must be achieved for the performance on a given task to be considered perfect. Thus, $C_m(x)$ is the percentage of automation failures undetected, $C_t(y)$ the percentage of time the aircraft spends outside the rectangular area, and $C_r(z)$ the percentage of time the fuel level is outside the targeted limit.

Another important requirement is that the elementary cost functions be decreasing. Indeed, the higher the sampling rate, the better the participant performs in the lookzone and the lower the corresponding cost. Hence the fundamental requirements for each of the elementary cost functions are:

Requirement (1): C(0) = 1
Requirement (2): C(1) = 0
Requirement (3): $C'(\eta) \leq 0$ for $\eta \in [0 ; 1]$

Two additional considerations were made to find the most realistic elementary cost functions. First, when the sampling rate is very low (close to 0), participants are expected to perform poorly. The elementary cost would thus remain very close to 1 for any sufficiently small values of η. This implies that the slope be more or less horizontal at $\eta = 0$ (i.e. $C'(0) \approx 0$).

Similarly, when the sampling rate is high enough, participants perform almost perfectly in the lookzone regardless of how close to 1 the sampling rate actually is. This implies $C'(1) \approx 0$ (i.e., the slope at $\eta = 0$ almost null).

2.2.3 Definition of the elementary cost functions and their parameters.

A continuous mathematical expression of $C(\eta)$ was needed for the optimization process of $\Omega(x,y,z)$. Given the requirements and considerations stated above, we made the assumption that $C(\eta)$ had the following form:

$$C(\eta) = 1 - \frac{1 - e^{-\alpha\eta^{\beta}}}{1 - e^{-\alpha}} \qquad \textit{Equation (1)}$$

where α and β have to be determined for each of the three lookzones.

Equation (1) satisfies *Requirements (1), (2)* and *(3)* automatically. For a given lookzone, parameters α and β were calculated so as to make $C(\eta)$ defined by *Equation (1)* fit a series of experimental data. To do so, three elementary costs, *d, e,* and *f* associated with three sampling rates, *a, b,* and *c* (arbitrarily chosen, and such that $a<b<c$) were experimentally determined. Given a sampling rate η, an estimate of the cost $C(\eta)$ can be obtained by averaging the percentage of what is missed in the zone over several test sessions when the operator looks at it for one second every $1/\eta$ seconds (periodic sampling). Given the experimental points *(a,d), (b,e)* and *(c,f),* we defined the slope parameter: $\mu = (f - d)/(c - a)$ (Figure 2).

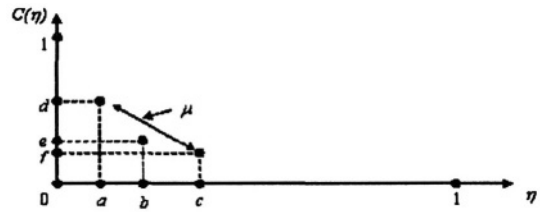

Figure 2. Construction of the elementary cost function based on three data points

α and β were determined to make $C(\eta)$ fit these three points as closely as possible using the following technique. First, we imposed that the actual

value of the cost function at the sampling rate b (middle point) be equal to its experimental estimation, e. Then, we assumed that the slope at this point could be approximated by μ. Hence, we solved for α and β in order to get:

$$C(b) = e \qquad \textit{Equation (2)}$$

$$C'(b) = \mu \qquad \textit{Equation (3)}$$

Using *Equations (1)* and *(2)*, β can be expressed in terms of α, b, and e:

$$\beta = \frac{1}{\ln b} \cdot \ln\left[-\frac{1}{\alpha}\ln\left(1-(1-e)(1-\exp(-\alpha))\right)\right]$$

Plugging this last expression into *Equation (3)*, α solves the equation below:

$$\mu = \frac{1}{b \cdot \ln b}\ln\left[-\frac{1}{\alpha}\ln\left(1-(1-e)(1-\exp(-\alpha))\right)\right]\ln\left(1-(1-e)(1-\exp(-\alpha))\right)\frac{1-(1-e)(1-\exp(-\alpha))}{1-\exp(-\alpha)}$$

Finally, a validation criterion was established to verify that the computed cost function $C(\eta)$ gave a fairly good representation of the problem. The cost estimation at a given sampling rate may greatly fluctuate from one test session to another. Given the sampling rate η, the radius $R(\eta)$ was defined as the greatest deviation to the mean value of cost estimate over the performed test sessions. Then, using *Equation (1)* for $\eta = a$ and $\eta = c$, we stated that:

If $C(a) \in [d - R(d) ; d + R(d)]$ and **if** $C(c) \in [f - R(f) ; f + R(f)]$,
Then, $C(\eta)$ given by *Equation (1)* was a sufficiently good representation.

Otherwise, the process had to be repeated and α and β had to be redefined, using other arbitrary sampling rates a, b, and c. The criterion states that $C(a)$ and $C(c)$ must remain within the acceptable range of variations about the estimated mean value of the cost at $\eta = a$ and $\eta = c$, respectively d and f.

2.2.4 Experimental determination of the cost functions' parameters

The next step in the model was to determine experimentally α and β for the three elementary cost functions. A pilot study was conducted where participants were asked to sample periodically one particular lookzone at a given rate. A metronome was used to pace the sampling. The performance cost associated with a given sampling rate was evaluated.

Cost associated with the monitoring task. The percentage of automation failures missed while periodically sampling the monitoring lookzone every $1/x$ seconds was determined. $1/x$ tests were performed corresponding to the different possible times the operator could start to sample in the [0 ; $1/x$ sec]

different possible times the operator could start to sample in the [0 ; 1/*x* sec] starting interval. The detection rate was then averaged over the number of tests. The three following sampling rates were chosen for both the high and the low reliability conditions: $a = 1/28\ s^{-1}$, $b = 1/20\ s^{-1}$, and $c = 1/14\ s^{-1}$. For the high reliability case, we found $\alpha = 85.08$ and $\beta = 1.57$. Hence:

$$C_m(x) = 1 - \frac{1 - e^{-85.08\,x^{1.57}}}{1 - e^{-85.08}}$$

For the low reliability case, we determined $\alpha = 97.52$ and $\beta = 1.63$. Hence:

$$C_m(x) = 1 - \frac{1 - e^{-97.52\,x^{1.63}}}{1 - e^{-97.52}}$$

Table 1. Experimental and analytical values of the monitoring cost function by sampling rate for the high reliability condition (87.5%) and for the low reliability condition (56.25%).

	Cost (% missed automation failures)			
	Constant High (87.5%)		Constant Low (56.25%)	
Sampling rate (x)	Experimental estimation (mean value ± radius)	Analytical value $C_m(x)$	Experimental estimation (mean value ± radius)	Analytical value $C_m(x)$
1/14	32% ± 7%	26%	26% ± 3%	27%
1/20	46% ± 4%	46%	48% ± 3%	48%
1/28	72% ± 11%	63%	67% ± 2%	65%

Cost associated with the tracking task. The cost of the tracking task was estimated by calculating the percentage of time spent outside the rectangular area for a given periodic sampling rate. The following sampling rates were tested: $a = 1/10\ s^{-1}$, $b = 1/8\ s^{-1}$, and $c = 1/5\ s^{-1}$. For each of them, a five-minute session was performed by five different operators. The corresponding costs were then averaged, and led to $\alpha = 10.86$ and $\beta = 1.04$. Hence:

$$C_t(y) = 1 - \frac{1 - e^{-10.86\,y^{1.04}}}{1 - e^{-10.86}}$$

Table 2. Experimental and analytical values of the tracking cost function by sampling rate

	Cost (% time outside of the rectangular area)	
Sampling rate (y)	Experimental estimation (mean value ± radius)	Analytical value $C_t(y)$
1/5	11% ± 2%	13%
1/8	29% ± 5%	29%
1/10	41% ± 6%	37%

Cost associated with the resource management task. The cost of the resource management task was established by evaluating the percentage of time the fuel level was beyond the targeted limit when the lookzone was periodically sampled at a given rate. This task greatly differed from the other two as the performance largely depends on the strategy of the operator. This led to large fluctuations around the mean value when trying to evaluate the cost. The following sampling rates were chosen: $a = 1/180\ s^{-1}$, $b = 1/120\ s^{-1}$, and $c = 1/60\ s^{-1}$. For each, three different operators performed 4 ten-minute blocks. Costs were then averaged and led to: $\alpha = 52.43$ and $\beta = 0.84$. Hence:

$$C_r(z) = 1 - \frac{1 - e^{-52.43\, z^{0.84}}}{1 - e^{-52.43}}$$

Table 3. Experimental and analytical values of the resource cost function by sampling rate

	Cost (% time outside of the targeted limit)	
Sampling rate (z)	Experimental estimation (mean value ± radius)	Analytical value $C_r(z)$
1/60	18% ± 12%	19%
1/120	39% ± 22%	39%
1/180	59% ± 31%	51%

2.2.5 Model verification

In order to verify that the monitoring cost function generated was sufficiently realistic, it was compared to additional experimental data. Table 4 summarizes the estimated monitoring cost for both reliability conditions and compares these experimental data to the analytical values obtained using the monitoring cost functions defined earlier. We can thus see that data obtained from the model are very close to those obtained experimentally.

Table 4. Percentage of automation failures missed by reliability and sampling rate determined experimentally and using the model

	Estimated percentage of failures missed			
	High reliability (2 failures)		Low reliability (7 failures)	
Sampling rate	*From the model*	*Experimental*	*From the model*	*Experimental*
1/5 s^{-1}	0.1	0	0.08	0
1/15 s^{-1}	29.8	33.33	30.7	34.76
1/20 s^{-1}	46.2	46.33	47.8	48.4
1/50 s^{-1}	83.3	79.83	84.7	80.31
1/100 s^{-1}	94	89.92	94.8	90.2
1/200 s^{-1}	97.9	94.96	98.3	95.52

2.2.6 Problem formulation and resolution

The optimal attention allocation strategy is given by the minimization of the overall cost function $\Omega(x,y,z)$ on the domain that satisfies the constraint: $x + y + z \leq 1$, where x, y and z are comprised between 0 and 1.

1. For the blocks with high reliability automation:

$$\Omega(x,y,z) = 3 - \frac{1-e^{-85.08\,x^{1.57}}}{1-e^{-85.08}} - \frac{1-e^{-10.86\,y^{1.04}}}{1-e^{-10.86}} - \frac{1-e^{-52.43\,z^{0.84}}}{1-e^{-52.43}}$$

2. For the blocks with low reliability automation:

$$\Omega(x,y,z) = 3 - \frac{1-e^{-97.52\,x^{1.63}}}{1-e^{-97.52}} - \frac{1-e^{-10.86\,y^{1.04}}}{1-e^{-10.86}} - \frac{1-e^{-52.43\,z^{0.84}}}{1-e^{-52.43}}$$

(x, y, z) are positive variables, and each of the elementary cost function has negative exponents. The magnitude of the exponent thus has to be as large as possible to minimize $\Omega(x,y,z)$, and we can expect *Constraint (1)* to become an equality which leads to $x + y + z = 1$. Minimizing $\Omega(x,y,z)$ becomes a 2-dimensional problem where y, and z are the two independent variables ($x = 1 - y - z$). The goal is to minimize $F(y,z) = \Omega((1-y-z), y, z)$ on the domain that satisfies the constraints presented above. Solving this minimization problem gives similar solutions for both reliability conditions:

- (x, y, z) = (0.232, 0.651, 0.117) for the Constant High condition,
- (x, y, z) = (0.228, 0.654, 0.118) for the Constant Low condition.

This solution indicates that, for both reliability conditions, the optimal strategy would be to sample the monitoring, tracking and resource lookzone every **4.3 sec**, **1.5 sec** and **8.5 sec**, respectively. This optimal solution exceeds what operators could achieve since the model assumes constant looking at the MAT battery. In reality, participants look away from the screen fairly regularly to perform control actions on the keyboard. This solution should thus not be regarded as a goal that participants should achieve but rather as a reference against which strategies can be compared.

3. APPLICATION TO PRIOR EMPIRICAL DATA

3.1 Summary of previous results

In Bagheri & Jamieson (2004), 24 participants completed four 30-minute sessions on the MAT battery for a total of 12 10-minute blocks. Automation reliability - defined as the percentage of the 16 malfunctions in each block that were corrected by the automation - was varied as a between-subjects factor. It was either constant at 87.5% (Constant High), constant at 56.25% (Constant Low), or changed every block from 87.5% to 56.25% (Variable Hi-lo), or from 56.25% to 87.5% (Variable Lo-hi).

The effect of reliability on detection rate was shown to be significant $F(3, 20) = 11.92$, $p < .001$. Post-hoc analysis revealed that the detection rate of Constant High participants' was poorer than that in any other condition.

3.2 Attention results

The sampling rates (x, y, z) were evaluated for each participant to determine the overall cost $\Omega(x, y, z)$ of his/her attention allocation strategy.

Overall cost function. Reliability had a significant effect on the log transformed cost function $F(3, 20) = 12.29$, $p < .001$. Constant High participants exhibited a significantly more costly behaviour than those in any other condition (Figure 3). They also had a sampling strategy significantly different from the optimal one $t(45) = 4.72$, $p < .001$. It should however be noted that this was also the case for the Constant Low and the Variable Hi-lo participants, $t(44) = 4.02$, $p < .001$, $t(50) = 3.13$, $p < .01$, respectively.

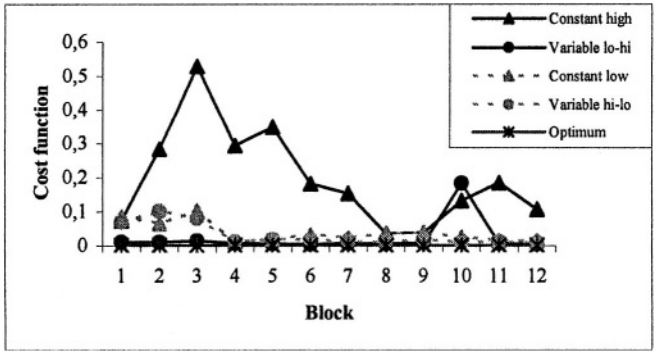

Figure 3. Overall cost functions by reliability group and block

The effect of block on the overall cost function approached significance $F(11, 121) = 1.66$, $p = 0.09$ (Figure 3). In each condition, participants developed a less costly sampling strategy over time.

Elementary cost functions C_m, C_t, C_r. Reliability had no significant effect on either the tracking or the resource cost function $F(3, 20) = 1.41$, $p > .05$, $F(3, 20) = 0$, $p > 0.05$, respectively. This was confirmed by participants' performance on both tasks, where no effect of reliability was found (see Bagheri & Jamieson, 2004). However, reliability had a significant effect on the cost function of the monitoring task $F(3, 20) = 19.50$, $p < .0001$. The difference observed in the overall cost function thus appears to be due to the sampling strategies for the monitoring task (Figure 4). Constant High participants had a significantly more costly behavior than what could be obtained when optimally sampling this task $t(45) = 4.11$, $p < 0.001$, whereas variable participants' behavior did not significantly differ from the optimum.

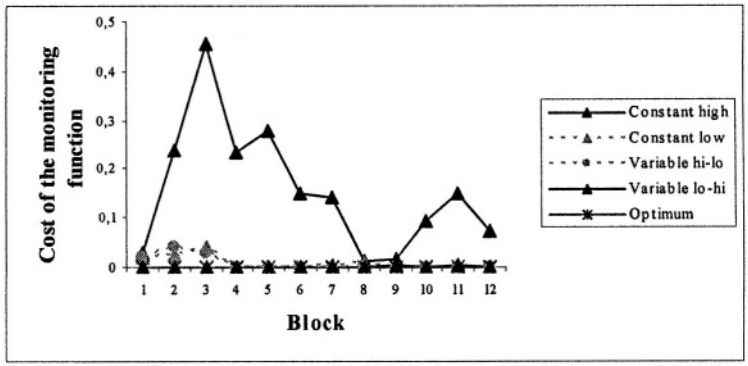

Figure 4. Cost of the monitoring task by reliability and block

4. DISCUSSION

The data from Bagheri and Jamieson (2004) showed that automation failure detection was significantly worse for Constant High participants, which could indicate that these participants were complacent. The model we presented here allowed us to consider this possible attribution more closely.

Eye point of gaze data first revealed that all participants (except those in the Variable Lo-hi condition) exhibited an overall strategy significantly more expensive than the optimal one. This difference was related to participants' tendency to sample the tracking lookzone less, and the resource lookzone more, than what was optimal (a trend observed in all conditions).

Moreover, the attention allocation strategy of Constant High participants was more costly than that of participants in all other conditions.

As shown in Table 5, very similar sampling rates are required for both reliability conditions to achieve a desired detection rate. This led to very similar monitoring cost functions and optimal sampling rates. As a result, Constant High participants' strategies could be compared not only to the optimal strategy but also to that of participants in other conditions. As shown in Figure 4, Constant High participants sampled the monitoring task significantly less than participants in the other reliability conditions, accounting for the higher cost of their overall strategy. The monitoring behavior of these participants could be interpreted as evidence of complacent behavior since better performance could be achieved with higher sampling rates as illustrated both by participants in the other conditions and the optimal solution. However, it is critical to note that after an increase of their monitoring cost function across the first three blocks, Constant High participants started to monitor more efficiently. By Block 8, they exhibited a monitoring behaviour comparable to that of participants in the other conditions and a trend converging toward the optimal behavior. This dynamic argues against the presence of complacency as complacent behavior is hypothesized to develop with prolonged interaction with highly reliable automation (Prinzel et al., 2001). This change in attention strategy was not observed from detection results (which remained significantly below that of participants in other conditions), which emphasizes the importance of measuring attention when evaluating monitoring performance.

It must be highlighted that a 100% detection rate of automation failures equates to detecting 7 failures in a block with low reliability automation and only 2 failures in a block with high reliability automation. However, for both reliability conditions, the same sampling rate is required to achieve this detection rate since automation failures occur randomly and unpredictably. Indeed, the MAT battery presents no local context that would allow participants to suspect that a failure might occur. Thus, sampling at a given rate is more rewarding for participants facing low reliable automation, or more 'laborious' for those facing highly reliable automation as they 'look for nothing' more often. It must be noted that the cost of 'looking for nothing' was only included as an indirect cost in the model, the cost of potentially be missing events in the unobserved lookzones. Jerison and Wing (1963) argued that detection rate provides reinforcement and hence control of the rate of the 'observing response'. Thus, detecting a failure could reinforce the sampling of the monitoring task. The design of this task, which requires a similar sampling rate regardless of automation reliability to reach a given detection rate, could thus partly explain the Constant High participants' poorer performance.

It can be seen that *Constraint (1)* is not that restrictive. Given the dynamics of the three tasks, participants could perform almost optimally on each of them as illustrated by the values of the cost function associated with the optimal solution: $C_m = 0.00018$, $C_t = 0.00095$, $C_r = 0.00017$. If the monitoring task was not automated, sampling the lookzone at the optimal rate previously defined would still allow participants to detect almost all of the malfunctions. This begs the question of the need to automate the task. Indeed, if the sampling rate necessary to detect all the failures is the same regardless of the reliability, then low reliable automation might lead to a more efficient behavior as it would better "reward" participants for looking at the monitoring task, and might prevent complacency from happening. This might even be a better option, since the sampling rate necessary to detect all of the failures does not seem to prevent efficient performance on the other tasks. If this holds across other studies of multi-task performance, it would have implication for the difficult problem of function allocation.

Model limitations. There are several limitations to the presented model. First, the operator was assumed to be a periodic sampler. Although, this might be true for the continuous tracking task, and perhaps the monitoring task, we would expect the operator to sample the resource lookzone more often when the fuel level approaches the boundaries. A model like Carbonell's (1966), in which the sampling interval depends on the distance from the boundary, might have been more appropriate, although much more complex.

A second limitation comes from the small sample used to verify that the equation of the different cost functions was sufficiently close to experimental data. A more accurate approximation could be obtained with more subjects and a wider range of sampling rates.

Third, the influence of control actions on sampling rate was not considered in our model. When control actions are performed, uncertainty due to the action is added to the existing background uncertainty (Crossman et al., 1974). Sampling intervals following control actions tend to decrease until the expected effects of the control changes have worn off.

5. CONCLUSION

We have developed a sampling model that allows us to determine the efficiency of a participants' attention allocation strategy. When applied to previously reported empirical data, the model demonstrated that users of constant, highly reliable automation exhibited monitoring behaviour that was more 'expensive' than (1) the optimal strategy and (2) that of participants in other reliability conditions. However, the evolution of their strategy does not

support the attribution of poor monitoring performance to complacency. This model also casts doubt on the need to automate the monitoring task in the MAT Battery and suggests that the simulation should be modified to more meaningfully assess monitoring of automation in multi-task environments.

ACKNOWLEDGMENT

Financial support was provided by NSERC. Frederic Winther greatly contributed to the development of the model. Roy Kwon provided valuable assistance in solving the optimization equation.

REFERENCES

Bagheri, N. & Jamieson, G. (2004). Considering subjective trust and monitoring behavior in assessing automation-induced "complacency". In *Proceeding of the Human Performance, Situation Awareness and Automation Technology II Conference,* Daytona Beach, FL.

Carbonell, J. (1969). A queuing model of many-instrument visual sampling. *IEEE Transactions on Human Factors in Electronics, HFE-5,*156-164.

Crossman, E., Cooke, J. & Beishon, R. (1978). Visual attention and displayed information in process control. In Edwards, E. & Lees, F. *The Human Operator in Process Control.*

Jerison, H. & Wing, J. (1963). Human vigilance and operant behavior. In D. Buckner & J. McGrath (Eds.), *Vigilance: a symposium.* New York: McGraw Hill.

Moray, N. (2003). Monitoring, complacency, scepticism and eutectic behaviour. *International Journal of Industrial Ergonomics, 31,*175-178.

Moray, N., T. Inagaki (2000). Attention and complacency. *Theoretical Issues in Ergonomics Science, 1,* 354-365.

Moray, N. & Inagaki, T. (1999). Laboratory studies of trust between humans and machines in automated systems. Trans. Inst. MC, *21,* No. 4/5, pp. 203-211.

Moray, N. (1986). Monitoring behavior and supervisory control. In Boff, K., Kaufman, L. & Thomas, J. (Eds), *Handbook of perception and human performance.*

Parasuraman, R., Molloy, R. & Singh I. L. (1993). Performance consequences of automation-induced "complacency". *The International Journal of Aviation Psychology, 3,* 1-23.

Prinzell, L. J., DeVries, H., Freeman, F.G., & Mikulka, P. (2001). Examination of Automation-Induced Complacency and Individual Difference Variates. (Tech. Memorandum No. 211413). Hampton, VA: NASA Langley Research Center.

Schumacher, W. & Geiser, G. (1983). Human control strategies in concurrent binary tasks under overload conditions. *Automatica,* Vol. 19, pp.723-727.

Singh, I., Molloy, R. & Parasuraman, R. (1997). Automation-induced monitoring efficiency: role of display location. *International Journal of Human-Computer Studies, 46,* 17-30.

Wiener, E. (1981). Complacency: Is the term useful for air safety? In *Proceedings of the 26th Corporate Aviation Safety Seminar,* pp. 116-125. Denver: Flight Safety Foundation, Inc.

This page intentionally left blank

DECISION MAKING IN AVALANCHE TERRAIN

A concept for an educational computer simulation tool for back-country ski guides with a special focus on human errors

Urs Gruber
WSL Swiss Federal Institut for Snow and Avalanche Research SLF, Flüelastrasse 11, CH-7260 Davos Dorf, Switzerland.

Abstract: Human errors are well recognised in avalanche education as one of the most important factors causing avalanche accidents. However, to date no adequate methods exist to enable people to learn about their own human weaknesses and ameliorate them. A second problem that contributes to many avalanche accidents is that it is not possible to exactly predict, when an avalanche will occur, since the triggering of an avalanche is a matter of probability and people are not well educated in dealing with probabilities. In order to overcome these two deficiencies, a concept for a role-playing computer simulation tool for backcountry ski guides is presented. Well-documented avalanche accidents, mixed with non-avalanche ski trips, are chosen as scenarios. At the start of the play, the role-player has to choose a role that is defined by two human factors, ambition and popularity, that aim to ensure that the player is acting in a realistic manner. During the simulation, the player has to make many decisions that influence his ambition and popularity as well as his safety. Avalanches are triggered based on a probability function such that the role-player can experience the consequences of relatively small occurrence probabilities and, subsequently, better understands the existing risk management rules.

Key words : avalanche education, human factor, decision-making, risk management, computer simulation

1. INTRODUCTION

Between 1980 and 1999 snow avalanches caused an average of 26 fatalities per year in Switzerland. Additionally, about 90 people on average per year were caught in an avalanche but survived (Tschirky et al., 2000).

Since 1992 avalanche education has started to focus not only on the snow cover analysis but also on risk management strategies to deal better with the uncertainties involved in the decision making process during back-country skiing (Munter, 1991). At the same time, the role of human factors has been recognised as one of the most important factors contributing to avalanche accidents. At present, several avalanche education books exist for back-country skiers (Engler, 2001; Larcher, 1999; Munter, 2003). All provide valuable risk management strategies about how to behave safely in avalanche terrain, including schemes and checklists that simplify the decision making process. These strategies are in good agreement with analytical decision making approaches (e.g. Dawes (1988), Yates (1990)) and include also elements of naturalistic decision making (e.g. Klein (1999)).

However, referring to the well-known quote "Tell me and I forget. Show me and I remember. Involve me and I understand." (either of Lao-Tse or Confucius), an educational tool is still missing that not only allows skiers to train the application of these strategies as often as possible in realistic situations but helps to thoroughly understand and subsequently accept these rules. The existing books are somewhere in between "tell me" and "show me" but are not at all "involve me". When it comes to human factors, it is especially important to be involved, in order to learn more about everyone's own human weaknesses. Schank (1997) who developed learning tools for business companies, stated that the best way of learning is by doing, failing, and practicing. Because most organisations can't afford massive on-the-job failure, Schank created a safe place to fail and learn: he rebuilt the reality with computer simulations and let the people learn using role-playing scenarios in this virtual reality.

Because avalanches accidents are often deadly, backcountry ski guides can not afford massive on-the-job failure either. Therefore, the goal of this paper is to propose a concept for a computer simulation tool that is based on role-playing scenarios in a virtual back-country-skiing environment. Within this computer simulation, backcountry ski guides have to make decisions that may lead to successful ski-trips or end in failures. Most fatal accidents are well documented in Switzerland (Winterberichte, 1936/37 - 1996/97). Information regarding location, the number of persons involved, the avalanche bulletin, detailed snow cover descriptions are available. These accident descriptions are a useful resource for developing challenging and close-to-reality scenarios. Within this concept of a role-play computer-simulation a special focus is directed to the integration of human factors.

The paper is structured as following. The next section gives a short summary about the existing rules how to behave in avalanche terrain in order to provide some basic understanding for readers that are unfamiliar with

snow avalanches. Afterwards, the most important existing deficiencies in teaching and implementing these rules for backcountry ski guides are described. The fourth section provides reasons, why a computer simulation tool is suggested to overcome the identified deficiencies. In the final section the basic ideas of the tool are presented.

2. STATE OF THE ART IN AVALANCHE DECISION MAKING: EXISTING RULES

Munter (1991) introduced a risk-based approach to avalanche education in 1991. The physical methods were supplemented by including typical human factors contributing to avalanche accidents. Later, Munter developed and refined the so called 3x3 scheme that systematically structures the back-country ski trip planning in order to reduce planning errors (Munter, 1997). Table 1 shows the core elements of this 3x3 matrix with the most important checks and questions.

Table 1. 3x3 backcountry trip planning scheme. Modified after (Munter, 2003)

	A)Snowpack/Weather	**B) Terrain**	**C) Human Factor**
1) At Home (Trip Planning) → Researchable information	Check: avalanche report, weather forecast, information from locals, etc.	Use: 1:25'000 map, guide books, photos, own knowledge of terrain	Who's coming? Skill level? Equipment? Who's responsible?
2) Local (At arriving at start location: Visible area) → Observations before setting out	Assess: general snow conditions, wind direction and loading, new snow amounts, oddities, visibility, temperature	Check info you've previously received, i.e. relief, slope angle, steepness, ski tracks, convex rolls etc., Are there existing ski tracks- how many and when made?	Who's in my group? Is the equipment and transceivers complete? Time plan for tour? Itinerary left with someone?
3) Zonal (Exact location of questionable slope) → Go or not to Go	Check new snow amounts, Visibility, Solar radiation, Assess possible slab potential, What's keeping the snow together?	What's above and below me?, Steepest part of slope, convex rolls? Near the ridge? Any wind pockets?	Tiredness of people, discipline, technique, are distance between each other necessary (precaution measures)?

Starting with the trip planning at home, the checks are refined in three steps until a decision has to be made whether or not to traverse a critical slope ((1) at home, (2) at the starting point of the ski trip (local), and (3) at critical locations during the trip (zonal)). At every level, three basic factors have to be checked: (A) The snow and weather conditions, (B) the terrain and (C) human factors.

Munter (1997) based his hazard assessment strongly on the hazard level H of the public avalanche bulletin (i.e. 1=Low, 2=Moderate, 3=Considerable, 4=High, 5=Very High) that is updated daily in Switzerland. Based on extended snow stability studies, he found that the danger potential D is a function of the hazard level according to Equation 1.

$$D \approx 2^H . \tag{1}$$

Based on these field studies, Munter developed a risk reduction method. He analysed avalanche accidents with respect to terrain parameters, group size and particular precaution measures for different avalanche hazard degrees and developed based on this study reduction factors RF that reduce the risk of an avalanche triggering. The idea is to reduce the damage potential D defined by the hazard degree H using these reduction factors RF to an accepted remaining risk level $R = 1$ according to Equation 2,

$$R = \frac{D}{RF * RF} \leq 1 . \tag{2}$$

Table 2 shows the reduction factors RF to be used in Equation 2.

Table 2. Reduction factors RF as stated by (Munter, 2003)

Category	Description of risk reduction	RF
Slope (first class)	1) Steepest slope portion between 35 and 40°, or	2
	2) Steepest slope portion 35°, or	3
	3) Steepest slope portion of less than 35°	4
Aspect and frequency (second class)	4) Avoid northern slopes: NW (incl.) - N - NE (incl.), or	2
	5) Avoid northern half of slopes: WNW (incl.) - N - ESE (incl.), or	3
	6) Avoid all critical slopes and altitudes specified in the avalanche bulletin	4
	7) Frequently skied slopes	2
Precaution, group size (third class)	8) Large groups with distance between each other person	2
	9) Small group (2-4 persons), or	2
	10) Small group with distance between each other person	3

Several derivates of the Munter approach were developed recently and tools such as colour-coded schemes, check-lists etc. were created that allow an even easier and more structured way to make decisions (Engler, 2001; Larcher, 1999).

For the purpose of the concept of an educational computer simulation tool it is important to understand, that there exist clearly defined rules about what risk level is acceptable. These rules can be used as objective safety measure to assess the decisions people made in the educational simulation tool.

3. IDENTIFICATION OF THE EXISTING DEFICIENCIES IN AVALANCHE EDUCATION

All the above mentioned new European methods as well as some North-American avalanche safety experts (Atkins, 2000; Fredston et al., 1994; Tremper, 2001) emphasise the importance of human failure as cause of avalanche accidents. Most of them deal with the "human factor" by mentioning, explaining and visualising as well as possible the most important human factor categories such as pride, "ego", hubris, "herding instinct", "testosterone effect", poor communication (Fredston et al., 1994), over confidence, complacency, poor group management (Atkins, 2000), "lion syndrome", "sheep syndrome", "horse syndrome" or cultural arrogance (Tremper, 2001). However, mentioning, explaining and visualising is often not sufficient to really change the way people behave. Fredston et al. (1994) provided a very illustrative example for a failure of theoretically teaching and discussing human factor aspects:

> "During one avalanche workshop with very unstable snow, the instructor picked a goal that was unrealistic for the conditions and knew that the group would have to turn around when they reached a certain crux spot. The group reached this last safe spot, ate lunch, and talked in great detail about all the clues to instability and the high avalanche potential. Everyone then put their packs back on and the group continued uphill. The instructor let them file out ahead of him, knowing that they could move about 40 meters before they were in real danger. The last person in line turned around, saw the instructor standing in place, and asked if he was coming. The instructor answered "hell no" and the group scurried back. They were asked why they decided to go in the face of all the data and were amazed when they discovered that even in an avalanche workshop, where communication is encouraged, they had fallen victim to peer pressure and the "sheep syndrome". They learned far more from falling into this trap than if they had just been told to turn around by the instructor." (Fredston et al., 1994).

In avalanche education there exists a deficiency of provoking traps, where human factor and other avalanche safety aspects can be learned by self-experience instead of only by reading about them in guidebooks or by being told by instructors. This experience is very important, since not all humans have the same human weaknesses. Therefore, it is crucial for every backcountry skier to identify and experience his or hers own human factor weaknesses and also to accept them in order to know in what kind of situations he or she has to take them into account. The problem is that such self-experience in real avalanche terrain is usually very dangerous.

Therefore, a method is needed that allow the participants of avalanche safety courses to make valuable experiences in a safe environment. In search of such techniques, Atkins (2000) found that in the aviation industry, fire fighting and military already several techniques have been developed to reduce the human factors in accidents. Orasanu and Martin (1998) provide a very comprehensive overview about errors in aviation decision-making and suggest strategies to improve the capabilities of aircrews. They split the problem into two parts: situation awareness and course of action. One point to improve the situation awareness is to provide to the pilots better diagnostic information and more accessible, comprehensible and integrated displays that show trends. Within the topic of avalanche safety, methods such as Munter's 3x3 checklist or the reduction method in combination with frequently updated avalanche information, that is now available on the internet exactly meet this demand.

As a second point, Orasanu and Martin (1998) mention the need to improve a decision makers experience by giving them better training: if they have a large number of exemplars to choose from, they will be able to select a model which more closely fits the problem. For avalanche education, we would have – as already mentioned – a lot of exemplars of such failures in the detailed accident descriptions (Winterberichte, 1936/37 - 1996/97), but they are not easy accessible by everyone.

Another deficiency is also related to the same problem: the consideration of likelihood of different course of actions is a problem. Orasanu and Martin (1998) stated that people are notoriously poor at integrating numerical probabilities. If an aircraft crew have been encountered in the past a somewhat similar risky situation and the crew has successfully taken a particular course of action, they will expect also to succeed this time. Given the uncertainty of outcomes, in many cases they will be correct, but not always. Reason (1990) called this "frequency gambling". Skiing a slope above 30° is always frequency gambling, since there is always a particular risk that an avalanche can occur. Since the exact instability pattern of a slope is unknown (Conway and Abrahamson, 1988; Landry et al., 2003), avalanche triggering is related to a probability. Part of the problem is that if nothing happens, you never know how close you have been to an avalanche triggering: you don't know how dangerous your "frequency gambling" was. A likelihood of 1:20 to trigger an avalanche in a particular slope is far above the accepted remaining risk, but in average this likelihood allows to ski this slope 19 times without that anything happens. Of course, the education methods will urge you not to ski such a slope, but if one sees others skiing this slope without anything happening your trust in these rules may be weakened. Therefore, it is important to have a tool that helps to understand better the essence of the likelihood.

4. OPPORTUNITIES OF COMPUTER SIMULATION TOOLS

Schank (1997) pictured an ideal computer simulation tool as follows:

> "You would be thrown into scenarios just as trainees are; you would be asked to make decisions and solve problems related to skills you are training for; and you would invariably make mistakes. When you messed up, you would have alternatives about what to do next. … You could hear an expert tell a story related to your failure; or you could start over and try again… Participants get angry, upset, confused, challenged, entertained and rewarded as they move through the plot."

If we compare this statement to the deficiencies mentioned in the previous section, we can recognise several connections: "Provoking traps" and "you would invariably make mistakes" or "lack of valuable role playing scenarios" and "you would be thrown into scenarios". With today's multimedia technologies it is possible to provide many scenarios derived from real accidents as well as non-accident trips. They can be elaborated to be very similar to reality in an attractive and entertaining way. Participants will be involved as interactive role-players and they are allowed to fail, since failing in a computer simulation game is safe and not embarrassing. One can also easily test the limits in order to better understand them. Barry LePatner once stated about learning by failing: "Good judgement comes from experience, and experience comes from bad judgement" (cited after (Tremper, 2001)).

Dörner (1989) used several computer simulations to scientifically study the behaviour of humans to solve complex problems such as to improve the welfare of the Moros, an African community. He underscored the fact that his models do not use "dirty tricks" to provoke failures of participants. Sometimes, the models are based completely on physics (i.e. the refrigerating storage house experiment, p. 201ff.). Other models are logically consistent and the participants mostly agree with the models when the driving model laws are disclosed to them after the computer simulation. Computer simulations have the advantage that there has to exist a well-defined model in the background that drives the simulation and evaluates the decisions of the participants. This model can be disclosed to participants after the computer simulation. The participants can later discuss whether or not the model is realistic.

Another advantage is the time-lapse capability of the computer simulations. In Dörner's Moro simulation, 20 years where simulated in 2 hours. This time-lapse capability provides the trainees with many more opportunities to make decisions than in real world experiences. Therefore, it

is possible to gather in a relative short time a lot of exemplars. These exemplars are not only presented to the user, but the user is directly involved in the decision making process.

Finally, computers may also help to ameliorate the human inability to deal with low propabilities. Since likelihood can be easily integrated into the simulation, people can experience the consequences of probabilities and start to learn to understand them.

5. CONCEPT OF A SIMULATION TOOL FOR BACKCOUNTRY SKI GUIDES

5.1 Principal goals of the concept

The concept of a computer simulation tool aims to overcome some of the identified deficiencies in avalanche education. It does not at all aim to compete with existing methods. On the contrary, it should be built directly on the existing knowledge and rules of how to behave in avalanche terrain. The target audience are not beginners, but backcountry ski guides, that already know the basic rules of backcountry ski safety. To them, the simulation game should provide an attractive and valuable training tool that:

a) allows to frequently apply the existing rules;
b) provides a realistic, but safe environment, where failures are possible;
c) enables experiences with a special focus of the recognition of everyone's own human factor weaknesses.

The bottom line is to involve the participant as much as possible to create a better understanding of the existing knowledge and rules.

5.2 Structure of the simulation tool

The 3x3 approach of (Munter, 2003) is used as a basis for the structure of the simulation, since it is well accepted within the back-country skiing community. Using the 3x3 approach the participants of the simulation proceed in the same way as they would on a real backcountry ski trip. Figure 1 shows the 3 spatial steps (1-3) as well as the 3 thematic factors (A-C) that have to be considered at every spatial step.

In the following, the three steps and the three factors are briefly outlined in order to provide a more detailed understanding of the procedure of the simulation structure. The factors that are involved in the simulation are indicated each time with the letters (A), (B) and (C) according to the legend in Figure 1.

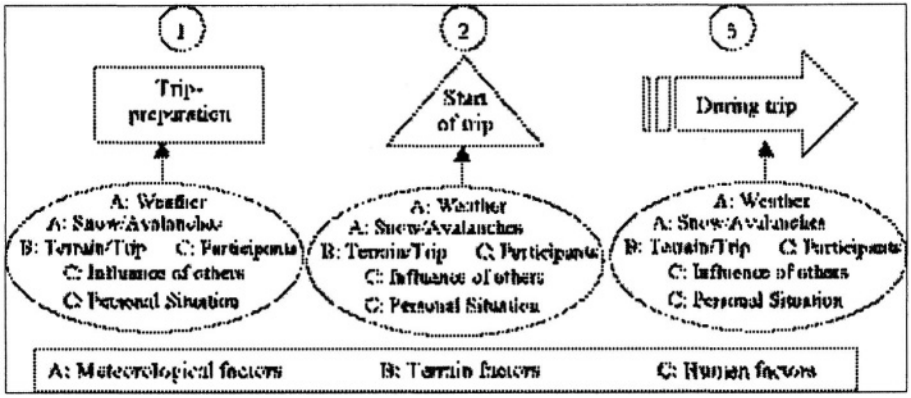

Figure 1. Structure of the backcountry skiing avalanche simulation game.

5.2.1 Preparation at home

The idea is to start the simulation with the planning process about 3 days before a ski-trip. At this time, the computer provides a randomly generated weather forecast for several areas (A). The person that plays the simulation (in the following named "actor" or "role-player") has access to the avalanche bulletin that is related to the given weather forecast (A). Based on this information, the actor – in the role of the back-country ski guide – has to make the decision whether or not to go on a ski trip and if he or she decides to go, where to go (B) and with whom to go (C). The terrain and trip descriptions (B) will be similar to those in reality (guidebooks, pictures, maps etc.). At this stage, the influence of others (C) can be: advice of colleagues, mountain guides etc. The potential participants are described in detail with respect to their skills, experience, fitness and behaviour (C). The actor has to define a time schedule for the ski trip and give instructions as to what equipment is necessary to take with.

5.2.2 Start of the ski trip

The computer simulates the travel to the ski tour departure point. Usually everything will go as planned and no tune delay will result. But the simulation may also introduce – driven by probability functions of various events – delays caused by reasons such as that one participant was still asleep at the appointed time or problems with cars, etc. Therefore, the proposed time schedule has to be compared with the real time spent with travelling and in case of a delay, its impact has to be assessed.

At the ski tour departure point, the actor receives an update of the weather and avalanche situation in the region of his ski trip by the simulation

(A). This information has to be evaluated with respect to the consequences for the planned ski trip. The actor has also to make a material check and security (beacon control). Also these tests will not always be perfect and the actor has to assess the consequences of failure with respect to material (C).

5.2.3 During the ski trip

Every ski trip has several decision points, i.e. points, where the actor has to decide whether or not to proceed the ski trip as planned, to take additional safety measures or to return. The decision will be influenced by the weather and avalanche situation that may or may not change during the trip (A), by the ability of the group to match the proposed time schedule (reasons for delays could be: not fit enough, material problems etc.) (C) and by the decisions of other groups (C). The decision will also be influenced on the terrain (B), i.e. the steepness, the aspect of a single slope portion. The terrain elements can be visualised using videos, images and digital maps.

5.3 Including the human factors in the simulation

As we have seen already, the human factor is a very important element in the decision making process. Tremper (2001) put it the following: "Human factors are woven into the fabric of every avalanche relevant decisions" or "Human beings have not only intelligence but also emotions and often, emotions are much stronger than intelligence". Another very interesting quote comes from the Canadian Mountain Helicopter ski guide Roger Atkins "Staying alive in avalanche terrain probably has more to do with mastering yourself than mastering any knowledge of avalanches" (cited after (Tremper, 2001)). All these human factors are well recognised as important, but unfortunately there does not exist an easy and convincing way to train the mitigation of these factors.

Making a training simulation game that tries to include the human factors appears to be difficult, since the actors will be very well aware of the fact, that the simulation may try to provoke human errors and thus, since the computer screen is not reality, the emotions that usually prevail will disappear and therefore every actor will try to play it safe, which should be quite easy with a normal intelligence and an average knowledge of the avalanche safety basics. Therefore, it will be one of the most difficult tasks of this simulation tool to introduce realistic human factors nevertheless, i.e. to provoke human error.

The basic idea to include the human factors in the simulation is to replace the emotions occurring in reality by artificial emotional parameters within the role-play. Before starting the simulation game, the actor has to choose a

role (i.e. mountain guide XY, mountaineering club guide WZ). Every role is defined by two emotional parameters: (1) ambition and (2) popularity. Ambition is meant as: "Am I myself satisfied with what I have reached" and popularity is meant as: "What are the others thinking of my decisions".

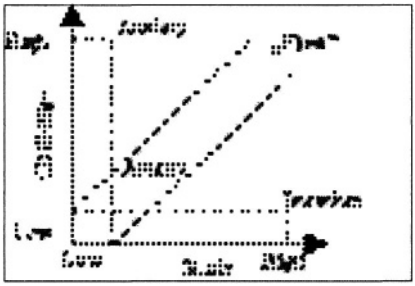

Figure 2. Sketch of the "flow" concept of (Csikszentmihalyi, 2002).

The ambition parameter is linked strongly to the "flow" concept of (Csikszentmihalyi, 2002). Following the core elements of this concept shown in Figure 2, every human being needs "flow experiences" in order to be satisfied with his or her living.

If someone is confronted with a lot of challenges but has no skills to manage them, he or she will be anxious. On the other hand, if someone has a lot of skills, but no challenge, it is boring. No skills and no challenge leads to apathy. What makes life really satisfying is when you are able (i.e. when you have developed the necessary skills) to manage even high challenges smoothly. This is what Csikszentmihalyi calls "flow experiences". Back-country skiing is challenging since it can be dangerous and as mentioned by Munter (2003) it requires a lot of skills to be a safe back-country ski guide. Thus a backcountry ski trip can provide "flow experiences". The link to the ambition parameter is that in order to be able to reach the level that allows flow experiences you need to be ambitious: You have to develop skills and you have to accept challenges. Of course not everyone has the same level of ambition, but without any ambition you usually don't lead ski trips. A person that always wants to climb the most difficult mountains and to ski down the steepest slopes with the best powder snow is considered to be more ambitious than a person that just likes to enjoy a sunny day. Ambition is also linked to the risk. The more ambitious you are, the more likely you are exposed to the avalanche danger.

Before starting the simulation, the actor must chose a back-country ski guide role with an associated ambition characteristics, e.g. a value within a range of 0 – 100, where 0 means: not ambitious at all and 100 means very ambitious. During the simulation (e.g. 4-5 different backcountry ski trip simulations) the actor must gather at least the points (e.g. 75) that are

associated with his role. He can gather these points by choosing appropriate trips, powder snow conditions and participants. The more difficult a trip is, the better the powder snow conditions and the more ambitious participants are the more points the actor will collect. If he fails to gather at the end enough ambition points, the simulation is over and the actor failed, because he was not satisfied with his own achievements. The need to achieve this predefined ambition limit should ensure, that the role-player has to take risks in a similar way as he would do it in reality. Since it is a role game one can choose different ambition levels and then experience the consequences of being very ambitious or only moderately ambitious. Finally, it should motivate the skiers to reflect on the own ambitiousness and the one of the colleagues in reality.

The second emotional parameter is "popularity". Decisions are often not independent of the opinions of colleagues or customers. Mountaineering club guides leading ski trips in their spare time like to be popular among their participants, as most humans do. However, mountain guides, that make their living guiding back-country ski trips need to be popular among their clients, otherwise they will not have any customers anymore. Therefore, as for the ambition parameter, you have to reach the number of popularity points that is associated with the role. If the role is the one of a mountain guide this value can be rather high. The gathering of popularity points depends strongly on the participants. If you have very ambitious participants, popularity points can be gathered for more or less the same decisions as ambition points. However, if your participants are very safety concerned you would collect only popularity points, if you make safe decisions.

5.4 Assessment of the decisions with respect to safety

The decisions of the actor will be assessed with respect to the safety in two ways. The first assessment will be, whether or not the group triggered an avalanche. When the group triggers during the simulation an avalanche, after the actor decided to ski a particular slope, it is a very strong indication, that some decisions were wrong with respect to the safety standards. The second safety assessment is based on the reduction method of (Munter, 2003).

We have seen that it is not possible to clearly state whether a slope will avalanche or not. We don't know a priori where the weak spots in a slope are located or whether or not the slope is homogenous. Consequently, it is a matter of probability if and where an avalanche is triggered. Even worse, the probability of an avalanche triggering is usually rather low which makes it more difficult for human beings to understand the risk of an avalanche event. A computer simulation is well suited to deal with probabilities, based on a

probability function as shown in Figure 3 in combination with Munter's reduction method.

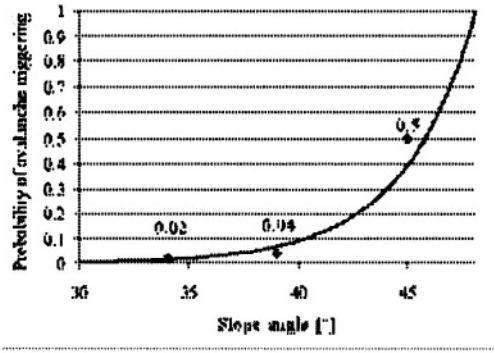

Figure 3. Possible avalanche triggering probability in function of the slope incline for an avalanche hazard level 3 without the application of reduction factors RF as stated in Table 2.

According to this probability function, a group of 10 people, skiing a northerly exposed slope of 45° without the application of precaution measures when the public bulletin hazard level is 3 (i.e. "considerable") will trigger an avalanche event with a probability of approximately 0.5, since no reduction factors RF according to table 2 are applicable. If reduction factors are applicable, e.g. the group size is less than 4 (provides a RF=2), the probability of an avalanche triggering will be 0.5/2 = 0.25 etc. The simulation will be able to determine all elements that are necessary to calculate this risk:

- The actor has to draw on a map exactly the path that he or she intends to go. Based on this path, the steepness and the aspect can be exactly derived;
- The avalanche bulletin provides the danger potential;
- The computer knows how many people are in the group and the simulation provides also information about how many existing trails are in a particular slope;
- Depending on the actor's request to apply precaution measures, the reduction factors will be taken into account.

For each slope, which the actor decides to ski, the triggering probability will be calculated and the simulation will trigger or not trigger an avalanche based on this probability.

The second safety assessment is based on the same parameters, but it does not provide information, whether or not an avalanche was triggered. Based on the same parameters, it is determined according to equation 2, whether or not the decision was above or below the accepted remaining risk:

For every decision point during the back-country ski trip, the reduction method will be used to assess the chosen decision with respect to its safety (point 1: R=0.2; point 2: R= 0.5; point 3: R= 0.2 → overall safety: 0.3 (i.e. safe at every point of the ski trip)). The idea of this second safety assessment is, that if the risk is above the accepted remaining risk, the safety assessment of the actor will be bad, even if he didn't trigger an avalanche during the first safety assessment method. In other words, if the actor decides to ski a slope with a triggering probability of 1:20, he will trigger an avalanche in average only once in 20 times, but he will nevertheless receive a bad safety rating, since he is above the accepted remaining risk.

5.5 Overall assessment at the end of the simulation

The concept of the assessment of the actor is to provide it at the time, when he would also be assessed in reality. During a learning sequence, the actor should play about 4-5 backcountry ski trips. During the ski-trip, he will be assessed immediately with respect to the following criteria:

- Whether an avalanche was triggered or not;
- Whether his decisions have been popular (direct feedback by participants);
- How many ambition points he gathered.

However, the results of the safety assessment by the reduction method will not be disclosed to him, but at the end of the learning sequence after he did several ski trips. The reason is, that in reality, you also don't have a direct feedback on your safety behaviour unless you trigger an avalanche. This aims to provoke the traps, i.e. to get the feeling that you can collect a lot of ambition and popularity points with skiing steep slopes without being reminded always by the reduction method that your risk is far above the accepted level.

The actor should realise, that ambition, popularity and safety are often contradictory and that he has to make decisions with the following goals:

- Satisfy the ambitions;
- Stay on the safe side;
- Be as popular as possible.

The actor should realise, that skiing a slope is always a sort of "frequency gambling" and therefore, learn to accept rules such as the reduction method for his risk management.

ACKNOWLEDGEMENT

The author is very grateful to the Swiss National Foundation that provided financing to elaborate the concept and to Werner Munter, whose contributions to avalanche education and risk management rules are crucial elements of this concept.

REFERENCES

Atkins, D., 2000. Human Factors in Avalanche Accidents. In: K. Birkeland (Editor), International Snow Science Workshop (ISSW). American Avalanche Association, Big Sky, Montana, USA, pp. 46-51.

Conway, H. and Abrahamson, J., 1988. Snow-slope stability - a probabilistic approach. Journal of Glaciology, 34(117): 170-177.

Csikszentmihalyi, M., 2002. Flow: The classic work on how to achieve happiness. Rider.

Dawes, R.M., 1988. Rational choice in an uncertain world. Harcourt Brace Jovanovich, San Diego.

Dörner, D., 1989. Die Logik des Misslingens: Strategisches Denken in komplexen Situationen. rororo science. Rowohlt, Reinbeck bei Hamburg, 320 pp.

Engler, M., 2001. SnowCard & Faktorencheck. Lawinenkunde vom Anfänger bis zum Profi. Berg & Steigen. Zeitschrift für Risikomanagement im Bergsport. Österreichischer Alpenverein(4/01).

Fredston, J., Fesler, D. and Tremper, B., 1994. The human factor - Lessons for avalanche education, International Snow Science Workshop (ISSW). American Avalanche Association, Snowbird, Utah, USA.

Klein, G.A., 1999. Sources of Power. How People Make Decisions. The MIT Press, Cambridge, Massachusetts, 330 pp.

Landry, C.C., Birkeland, K.W., Hansen, K., Borkowski, J.J. and Brown, R.L., 2003. Variations in snow strength and stability on uniform slopes. Cold Regions Science and Technology(Special Issue ISSW 2002).

Larcher, M., 1999. Stop or Go. Entscheidungsstrategie für Tourengeher. Berg & Steigen. Zeitschrift für Risikomanagement im Bergsport. Österreichischer Alpenverein(4/99).

Munter, W., 1991. Neue Lawinenkunde. SAC-Verlag, Bern.

Munter, W., 1997. 3x3 Lawinen: Entscheiden in kritischen Situationen. Agentur Phol und Schellhammer, Garmisch Partenkirchen, 229 pp.

Munter, W., 2003. 3x3 Lawinen: Risikomanagement im Wintersport. Verlag Pohl & Schellhammer, Garmisch-Partenkirchen, 223 pp.

Orasanu, J. and Martin, L., 1998. Errors in Aviation Decision Making: A Factor in Accidents and Incidents., 2nd Workshop on Human Error, Safety, and System Development,, Seattle, Washington, USA, pp. 100 -107.

Reason, J., 1990. Human Error. Cambridge University Press, Cambridge, UK.

Schank, R., 1997. Virtual learning: a revolutionary approach to building a highly skilled workforce. McGraw-Hill, New York, 185 pp.

Tremper, B., 2001. Staying alive in avalanche terrain. The Mountaineers Books, Seattle, 272 pp.

Tschirky, F., Brabec, B. and Kern, M., 2000. Avalanche Rescue Systems in Switzerland: Experience and Limitations. In: K. Birkeland (Editor), International Snow Science Workshop ISSW. Montana State University, Big Sky, Montana, USA, pp. 369-376.

Winterberichte, 1936/37 - 1996/97. Schnee und Lawinen in den Schweizer Alpen. Winterberichte des Eidg. Institutes für Schnee- und Lawinenforschung, Davos.

Yates, J.F., 1990. Judgment and decision making. Prentice Hall, Englewood Cliffs, NJ.

FAILURE ANALYSIS AND THE SAFETY-CASE LIFECYCLE

William S. Greenwell, Elisabeth A. Strunk, and John C. Knight
Department of Computer Science, University of Virginia

Abstract: The failure of a safety-critical system, though undesirable, is often a source of valuable lessons that can help prevent future failures. Current analysis practices do not always yield as much knowledge as they might about possible flaws in the system safety argument. In this paper, we introduce the lifecycle for safety cases. We use it to develop a framework to guide the analysis process and the development of lessons and recommendations. We illustrate the ideas with an example using the failure history of an air-traffic-control safety system.

Key words: failure analysis, safety cases, assurance

1. INTRODUCTION

Safety-critical systems are engineered to prevent failure, but, despite this, accidents and incidents sometimes occur. Developers create safety arguments for each safety-critical system they produce, even though the arguments might be informal, flawed, or undocumented. A safety-related failure of the system implies that there is a flaw in the safety argument, and continuing to operate the system without reassessing the safety argument—even if the immediate cause of the problem has been identified and corrected—is potentially dangerous.

In this paper, we use the safety case as a framework to embody a safety argument, showing how that argument can guide failure analysis and how failure analysis can be used to update the argument. Although safety-case maintenance has been studied, current theory does not combine safety-case maintenance with current work in accident investigation. We introduce a

comprehensive, post-deployment lifecycle for the safety case in which detailed and explicit feedback paths help maximize the benefits realized from any failure. The lifecycle updates the safety case throughout the system's life so that the safety case continues to provide a convincing argument that the system is safe, even after a failure.

The paper is organized as follows. Section 2 describes the safety case, introducing it as the focus not only of system assurance but also of failure treatment. Section 3 details the specifics of our lifecycle framework, and Section 4 gives an example of how it might be applied. Section 5 then concludes the work.

2. SAFETY CASES

In safety-critical systems, the primary loss to be addressed is undesired system behavior that will lead to death or injury to persons or damage to property. Such loss could be extremely serious; in most cases, the system developers are regulated by an external body to ensure that the potential for loss has been adequately assessed and addressed. Assuring system safety, however, is a formidable task. A system cannot be proven to be safe unless it has been operated over all possible inputs to ensure that it can never reach a hazardous state. Any other method of assurance, however well-formed, leaves open the possibility that the assured system properties are insufficient or invalid. For most systems, however, exhaustive testing is impractical, and for others it is impossible.

To avoid this problem, developers rely on other assurance techniques that produce arguments, but not proofs, that the probability of a system's failing in a way that could potentially cause a violation of safety properties is below a maximum acceptable threshold. Several widely-used industry standards for arguing assurance of safety rely on a process-based approach, such as RTCA DO-178B (Weaver, 2003; RTCA, 1992). These approaches assume that following a prescribed set of development processes will result in the production of a safe system. There is, in fact, little evidence to support this assumption (McDermid, 2001), and in process-based safety arguments, there is no way to comprehensively analyze the argument if a system failure occurs. Lessons could only be applied to the process, and there might not be an obvious causal relationship from a particular part of the process to a particular system failure.

Hamilton and Rees(1999) and Weaver (2003) argue that, instead of prescribing a particular process, standards should instead specify what types of evidence developers can generate to show that their systems meet the necessary safety requirements. An example of this type of assurance

structure is the safety case, which argues that a system is safe to use in its intended environment. More specifically, it argues that the risks associated with the operation of the system have been reduced to an acceptable level.

To look at the various elements that make up a safety case, we summarize the *Goal Structuring Notation* (GSN), a graphical notation developed at the University of York for depicting safety arguments (Kelly, 1998). The elements of an argument modeled in GSN are shown in Fig. 1.

In the rest of this paper, we use the elements of GSN as a model for the different pieces of information a safety case should contain because GSN was created precisely for the purpose of assisting engineers in making structured safety arguments.

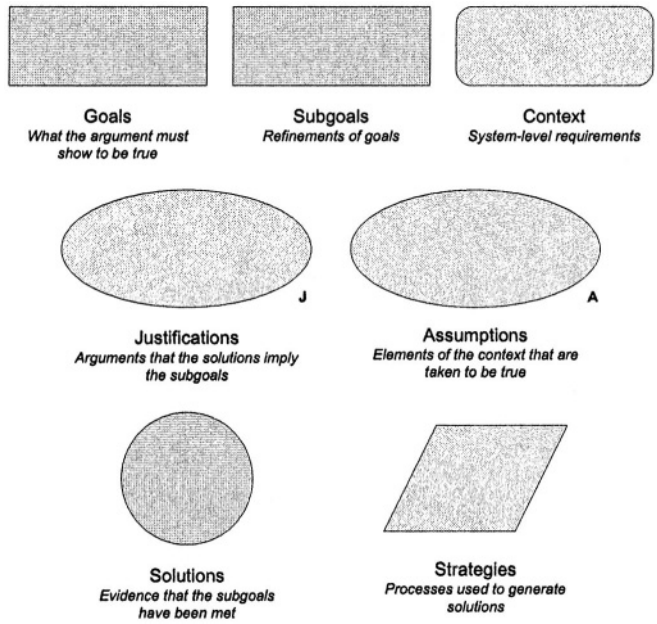

Figure 1. Basic Elements of GSN (Kelly, 1998)

3. THE ENHANCED SAFETY-CASE LIFECYCLE

An important feedback path exists between system development and failure analysis. Faults often manifest themselves after deployment, and the role of analysis after a failure is to uncover these faults. Typically, once a failure—either an incident or an accident—has been analyzed, the lessons and recommendations from the analysis are fed back into the development lifecycle to help prevent future failures.

Ideally, this feedback mechanism would ensure that all the benefits of failure analysis would be realized. However, the complexity of modern safety-critical systems makes the systems very difficult to analyze. Coupled with the informality of the failure analysis process, this complexity reduces the level of confidence that the lessons obtained from an analysis are comprehensive and correct. In addition, differences in designs and in development practices can limit the scope of lessons and recommendations significantly. Opportunities to discover and correct faults in systems and flawed development practices are sometimes missed; we discuss an example of this problem in Section 4.

Safety cases employing evidence-based assurance offer a solution to this problem by providing a rigorous basis for the feedback path between system development and failure analysis. The core safety-case lifecycle has been introduced by Kelly and McDermid; it is the process through which updates are made to the safety case for a given system when a challenge arises (Bishop, 1998; McDermid, 1999; Kelly, 1998). One of the challenges they note is that of a mishap (an accident or incident). We introduce the enhanced safety-case lifecycle, which uses the safety case to guide failure analysis after a mishap and to update the safety case based on the results of the analysis.

A failure is evidence that a system's original safety argument was flawed; the system was in fact less safe than expected. Thus, for any particular failure, we define *two* safety cases: the pre-failure safety case and the post-failure safety case. The former is the original safety case that was developed for the system before the failure. The post-failure safety case is the pre-failure safety case amended to reflect the results of the failure analysis. The post-failure safety case is essentially the result of correcting the original, flawed safety argument.

The fact that the original safety case was flawed and had to be updated as a result of the analysis of a failure leads to the enhanced safety-case lifecycle (illustrated in Fig. 2). The safety case is subject to revision over time based on experience, and careful and systematic determination of the essential changes provides the basis for both the analysis process and the determination of lessons and recommendations.

In order to elaborate the framework and add rigor to the feedback path, it is necessary to note that all of the basic elements of a safety case—including assumptions and contextual information—are first-class objects within the rigorous structure of the safety case. The failure analysis process that we have developed is based on examining these objects in context in the safety case and determining which are somehow defective following an accident.

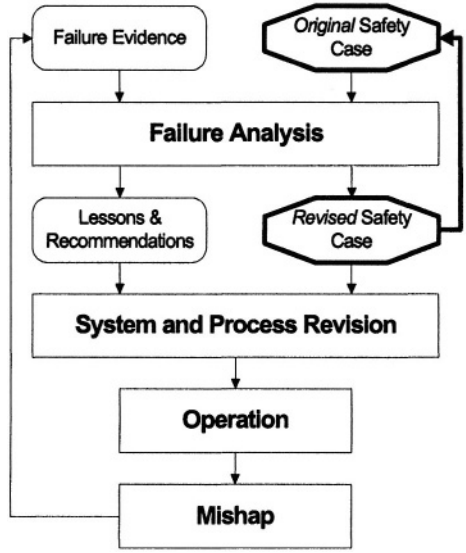

Figure 2. The Enhanced Safety-Case Lifecycle

Viewing each element as a first-class object ensures that the analysis is systematic and covers all aspects of the argument. This contrasts with the traditional approach where assumptions, for example, are usually treated in an ad hoc manner. Treating all of the elements of the safety case in a systematic way is facilitated by basing the analysis on a rigorous safety case.

The essence of our overall approach then is to begin with the original safety case for the system and organize the failure analysis around the process of correcting the safety case. Lessons show the elements of the pre-failure safety case that cannot be used in the post-failure safety case, and recommendations suggest ways to address the lessons.

3.1 Safety Goals of Failure Analysis

Investigating failures presents opportunities to learn lessons and prevent future failures. In many cases, however, important lessons are not recognized or essential corrective actions are not taken—such as in the case of the Korean Air flight 801 accident (NTSB, 2001). This stems in part from the lack of a coherent and complete approach to the treatment of failure. Researchers have developed a variety of approaches to the determination of causal factors (Johnson, 2003), but there is no rigorous framework within which lessons are developed, corrective actions defined, corrective actions implemented, and so on.

Accident investigations generally produce two major classes of results: *findings,* an investigation's analysis of the factors contributing to the accident; and *recommendations,* the investigation's suggestions as to how similar accidents could be prevented. The term *lesson* is often applied to findings that expose a flaw in the system or its development process, and recommendations tend to be directed towards fixing the flaws exposed by lessons, whether those lessons are stated explicitly or inferred from the findings. While the meanings of the terms are fairly clear in this informal context, we can define these terms more rigorously using the safety case as the artifact to which they apply.

3.2 Lessons Learned About the Safety Case

In the enhanced safety-case lifecycle, the first step in constructing the post-failure safety case is to find the flaws in the pre-failure safety case, i.e., determine the lessons of the incident. The safety case structure can aid the failure analysis process for determining lessons. Two specific techniques for doing this are *backtracking* and *dependency analysis,* such as in Kelly's process of using GSN to support safety-case maintenance (Kelly, 1998). Investigators would begin a failure analysis by assuming that the top-level goal of the safety argument was not satisfied and then backtracking through the argument to isolate faulty assumptions and evidence. Upon identifying a flaw in the safety argument, investigators could then use dependency analysis to determine which parts of the argument are affected by the flaw. Dependency analysis is a powerful tool for assessing the severity of a flaw, particularly if the flaw related to an assumption or piece of evidence used in safety arguments for other systems.

Once investigators have analyzed a failure and identified the flaws in the safety argument that contributed to it, they can prepare the lessons resulting from the analysis. These lessons state the elements of the safety argument that are faulty and in need of revision. For example, a lesson that might be learned from the failure of an advisory system could read:

> *"The assumption that a visual alert alone is sufficient to notify the controller of an altitude violation is invalid because the controller might not be at the workstation when the alert is raised."*

This lesson might derive in the enhanced safety-case lifecycle from a safety case element that relied on the assumption:

> *"Any visual alert would be seen by the controller."*

3.3 Recommendations Derived from the Failure Analysis

Recommendations in the enhanced safety-case lifecycle are suggestions made by an investigating team that are intended to help system engineers in the task of creating a valid post-failure safety case. Recommendations can take two forms. The first is that of a revised piece of the safety argument. System engineers would be expected to construct a corresponding revised system design that would allow the use of the argument fragment in the system's updated safety case. For the example lesson given above, a corresponding recommendation might state:

> *"It may be assumed that visual and aural alerts are sufficient to notify the controller of an altitude violation, because, although controllers might not be at their workstations when an alert is raised, at least one controller will be in the tower at any given time."*

In order to use this assumption in a safety argument, a developer would now have to show that the system generates both visual and aural alerts to conclude that the controller will be notified of an altitude violation, which might require changes to the system's design.

The second form a recommendation might take is that of a possible system change, always accompanied by a corresponding postulated change to the pre-failure safety case. If the system engineer chooses to implement recommended system changes, he can use the postulated change to guide the development of the actual post-failure safety case.

Recommendations are weaker statements than lessons. By invalidating an assumption, the effect of a lesson on the system development / failure analysis feedback loop is to forbid the use of that assumption in current and future safety arguments. On the other hand, a recommendation merely offers a potential system change and / or alternative safety argument to use in the updated safety case. Compliance with a recommendation does not automatically imply compliance with a safety goal. While the recommendations can include a potential safety argument, the safety case is a model of system operation and cannot be evaluated apart from the actual system design.

4. EXAMPLE: THE MINIMUM SAFE ALTITUDE WARNING (MSAW) SYSTEM

To illustrate the use of the enhanced safety-case lifecycle, we present a hypothetical example based on the Federal Aviation Administration's (FAA)

Minimum Safe Altitude Warning (MSAW) system. We selected the MSAW system for this example because it was conceived, implemented, and has since been revised through an informal interaction between the FAA and the National Transportation Safety Board (NTSB), a process intended to serve a purpose related to that of the enhanced safety-case lifecycle. We step through the development and revision history of MSAW and show how it could be accomplished in a more systematic, rigorous manner.

The MSAW system did not have an explicit safety case defined. By putting it into service, however, the FAA has made an implicit statement that it enhances the safety of the national airspace system. We have attempted to assemble the FAA's implicit safety argument from available system descriptions, accident reports, and communications with FAA officials.

4.1 System Description

MSAW is a software system designed to alert air traffic controllers to aircraft flying below a predetermined minimum safe altitude. The system receives altitude data for tracked aircraft from radar returns and compares the data against a terrain database containing elevations for the surrounding area. If an aircraft's reported altitude is below, or is predicted to descend below, the minimum safe altitude for the region in which it is operating, the system issues an alert to the controller. Upon receiving the alert, the controller is responsible for contacting the aircraft and notifying the flight crew of the hazard.

4.2 MSAW Chronology

Prior to the development of MSAW, the implicit safety argument that controllers would detect and notify low-flying aircraft relied on the assumption that they would be able to manually spot such aircraft on their radar displays by examining the altitudes reported in each aircraft's data block. The radar displays did not provide terrain information, so controllers would also have to know the minimum safe altitude for the region in which each aircraft was flying. In response to a December 1972 accident in which a commercial aircraft crashed near Miami, Florida, the NTSB issued a safety recommendation asking the FAA to "review the ARTS III [air traffic management] program for the possible development of procedures to aid flight crews when marked deviations in altitude are noticed by an air traffic controller" (NTSB, 2001). The FAA responded to this recommendation by implementing the MSAW system in 1977. Despite the deployment of MSAW, accidents that it was designed to prevent have persisted, and the

system has been changed numerous times to address them. These accidents are highlighted below.

On September 8, 1989, a commercial aircraft struck four transmission lines while executing an approach to Kansas City International Airport in Missouri. The NTSB found that, although the airport's tower was equipped with MSAW, the system failed to raise an alert because the altitude violation occurred in a region that had been excluded from MSAW processing due to a configuration error. The Board recommended that the FAA provide guidelines to its facilities for configuring the MSAW system to prevent such errors from recurring. The FAA implemented this recommendation in 1993.

On June 18, 1994, a Learjet crashed just short of the runway at Dulles International Airport. Upon investigating the MSAW system at Dulles, the NTSB again found an instance in which the system failed to generate any alerts because it had been configured improperly and recommended that the FAA conduct a "complete national review of all environments using MSAW systems" (NTSB, 2001). The FAA completed this review in 1996.

On January 29, 1995, during the period of the FAA's review, a small aircraft crashed during a missed approach to a regional airport in Georgia. The NTSB found that, although the airport tower received four MSAW warnings concerning the aircraft, the controller did not notice the warnings because he was attending to other duties. The MSAW system installed at the tower was configured for visual alerting only, and the controller was not monitoring the screen on which the warnings were being displayed. The NTSB concluded that the controller would have noticed the warnings if they had been issued aurally as well as visually, and asked the FAA to "require the installation of aural [MSAW] equipment [in facilities] that would otherwise receive only a visual alert" (NTSB, 2001).

On October 2, 1996, a small aircraft crashed while on approach to a regional airport in Maryland. The controllers at the regional ATC facility stated that they did not see or hear any MSAW alerts concerning the aircraft. Upon investigating the facility, the NTSB discovered that the MSAW aural alarm speaker "was covered with heavy paper held in place with what appeared to be masking tape," apparently because the system had frequently been generating nuisance warnings (NTSB, 2001). The NTSB recommended that the FAA immediately inspect its air traffic facilities for muted MSAW speakers, train controllers on the nature of MSAW and how to respond to MSAW alerts, modify the MSAW system to enhance the conspicuity of jeopardized aircraft, and, in its upcoming STARS ATM system, include a MSAW aural alert speaker at each radar display. The FAA has complied with some of these recommendations; others are still ongoing.

On August 6, 1997, Korean Air flight 801, a Boeing 747 carrying 254 people, crashed on approach to Guam International Airport, killing 228 and

injuring 26. During its investigation, the NTSB learned that the FAA had intentionally inhibited the Guam MSAW system in order to eliminate nuisance alarms. If the system had been operational, it would have issued an alert concerning Korean Air 801 64 seconds before impact. In response to this accident, the FAA recertified all of its MSAW installations and revised its standards and guidelines for configuring MSAW systems. It also developed a comprehensive program to validate MSAW site configurations as part of its commissioning process for new air traffic facilities.

On January 13, 1998, a Learjet crashed during approach at Houston Intercontinental Airport in Texas. The NTSB's investigation revealed yet another instance in which the MSAW system was configured improperly in spite of the guidelines the FAA produced in response to the Guam accident.

With these various incidents in mind, in the next section we hypothesize a series of revised safety cases that might have been created for this system had the FAA documented its safety argument rigorously. We also argue that, had they followed the enhanced safety-case lifecycle, many of these accidents could have been avoided.

4.3 MSAW System Safety Argument

Even though the MSAW system was not certified using evidence-based assurance, it was developed, augmented, and revised through the analysis of failures, or more specifically, aircraft accidents. Thus, it exemplifies how the enhanced safety-case lifecycle could be put into practice. The MSAW failures involved in the incidents described above can be classified into two major categories: (1) those in which MSAW did not generate an alert because it was configured improperly; and (2) those in which MSAW generated an alert but the alert failed to notify the controllers. This section considers how the latter set of incidents would have affected the MSAW safety argument had the FAA and the NTSB employed the lifecycle we discussed in Section 3. The accident in which the MSAW speakers were muted, although belonging to this category, is outside our scope of consideration.

Before MSAW, the safety argument that ATC would detect low-flying aircraft relied on the assumption that controllers would be able to manually spot such aircraft on their radar displays. This argument is illustrated in GSN in Fig. 3.

Upon investigating the 1972 accident, the NTSB would have learned that the justification J1 was invalid because merely displaying the altitude data was not sufficient to alert the controller of low-flying aircraft and because controllers were not always aware of the terrain elevations for each region of their radar displays. Therefore, the *lesson* from this accident, stated

according to the guidelines from Section 3.2, would be, "The justification that the controller is aware of the regional terrain and will manually identify

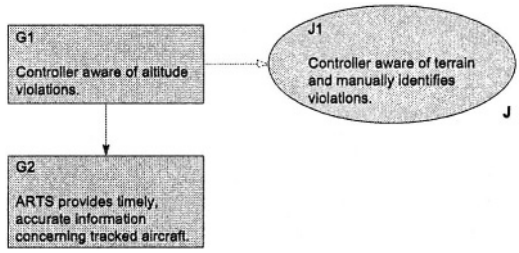

Figure 3. Assumed Original Implicit Argument for Controller Awareness

altitude violations (J1) is invalid." It is clear that without this justification, the original safety argument is invalidated because satisfying subgoal G2 no longer satisfies goal G1. The recommendation from this accident, stated as a revision of the invalidated justification, would be, "Issuing an automated alert whenever an altitude violation occurs would be sufficient to notify the controller of the violation." In order to use this new justification to rebuild the safety argument, the developer (in this case, the FAA) would now have to show that the system provides such an alert. Showing this property would have required the FAA to change the system by implementing MSAW, resulting in the new safety argument depicted in Fig. 4.

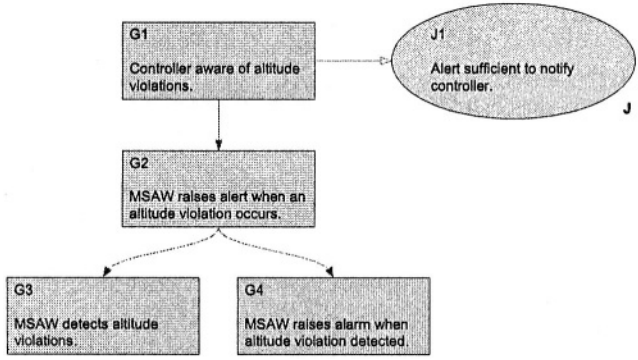

Figure 4. Assumed Original MSAW Controller Awareness Argument

The revised safety argument addresses the lessons and recommendations from the 1972 accident, but it is still flawed. Justification J1, which was the recommendation from the 1972 accident investigation, is vague because it does not precisely define what an alert should be. Consequently, the justification can be used to show that G2 implies G1 irrespective of the type

of alert used. The NTSB discovered this ambiguity in its investigation of the 1995 Georgia accident when it learned that a visual alert alone would not always attract the controller's attention. The lesson from that accident would once again be that J1 is invalid because the meaning of "alert" is ambiguous. The justification should read, "Visual *and aural* alerts are sufficient to notify controllers of altitude violations." Alternatively, contextual information could be included to define an alert as comprising both visual and aural components.

As a result of the lessons from the 1995 investigation, the MSAW safety argument was once again invalidated. In order to rebuild the argument this time, the FAA would now have to show that the system provided visual and aural alerts requiring the installation of speakers at each ATC facility. The final hypothetical argument is shown in Fig. 5, which includes revisions necessitated by the lessons learned from the configuration-related MSAW accidents such as the 1997 Guam accident. Subgoals G5, G6, and G7 are placeholders and would need to be developed further in an actual argument.

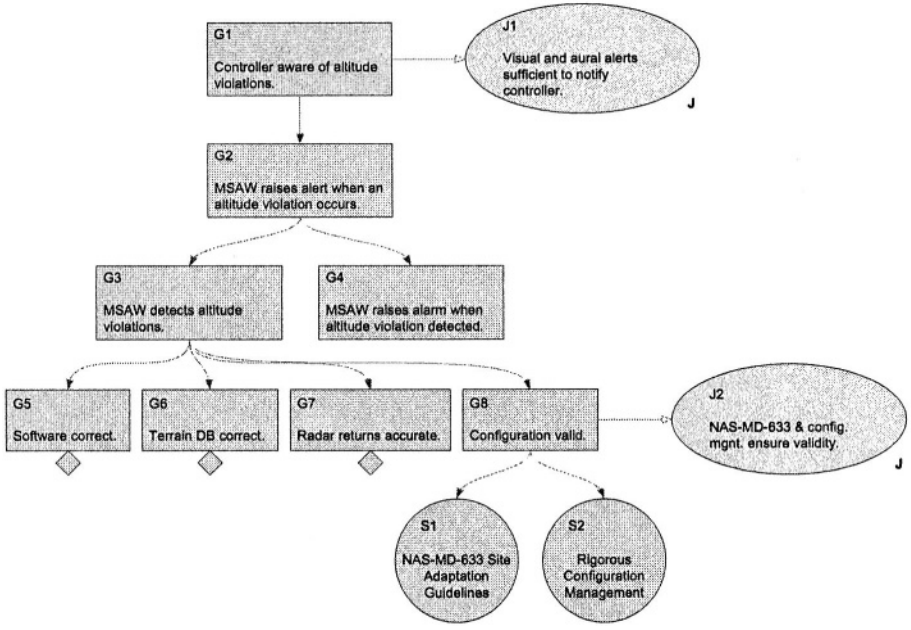

Figure 5. Revised MSAW Controller Awareness Argument

4.4 Observations

Seven accidents occurred between the time the need for a minimum safe altitude warning system was realized and the time at which its safety

argument reached its current state. Of these accidents, only three provided new information that could not have been inferred from prior failures. The 1972 accident established the need for MSAW, the 1989 accident showed that configuration management played a crucial role in the system's ability to detect altitude violations, and the 1995 accident revealed that aural as well as visual alerts were needed to notify controllers of violations. We hypothesize, but we cannot claim, that had the NTSB and FAA used the enhanced safety-case lifecycle we propose, at most these three accidents would have occurred before the safety argument shown in Fig. 5 would have been reached.

Our justification for this statement is that the first-class status of each argument element in the safety case would have required investigators to examine each element associated with the defective part of the pre-failure safety case. By having to document and justify the revised assumption in the post-failure safety case concerning MSAW alerting (which turned out to be flawed), investigators would have been more likely to detect the ambiguity in the meaning of "alert" that contributed to the 1995 accident. The probability of detection might have been raised further if the difficulty of obtaining proper shared meaning of terms like "alert" had been addressed (Hanks, 2003).

One notable difference between the NTSB's safety recommendations and our lessons and recommendations lies in how change is effected. In the former, the call for change is made in the safety recommendation, which usually suggests specific design or procedural changes. These changes might be infeasible for various reasons, and so they may be rejected by the system developer (as actually happened in this case, where the developer was the FAA). In the lifecycle we propose, the call for change is made in the lessons, which are findings of fact evidenced by the failure itself. They are separated from the recommendations, which suggest possible means, but not the only means, of revising the safety argument in light of the new lessons. The validity of the lessons is independent of that of the recommendations, and so the call for change remains intact even if a developer disagrees with the recommendations that accompany it.

5. CONCLUSION

Developers of safety-critical systems and those who investigate the failures of such systems share a common goal of ensuring that the system does, in operation, meet the goals set out in its safety case. This assurance process is most efficient and effective when the relationship between a system's operation and the case for its safety is clearly stated and exploited.

We have developed and illustrated a method for using the safety case to assist in failure analysis. This method greatly improves our assurance of system safety because it augments traditional safety-case development and maintenance activities to include explicit feedback through analysis of the system's failures in operation.

ACKNOWLEDGEMENTS

This work was funded in part by NASA Langley Research Center under grants numbered NAG-1-2290 and NAG-1-02103.

REFERENCES

Bishop, P.G., and R. E. Bloomfield. "A Methodology for Safety Case Development." Safety-critical Systems Symposium, Birmingham, UK, Feb 1998.

Hamilton, V. and Rees, C. "Safety Integrity Levels: An Industrial Viewpoint." *Towards System Safety: Proc. Seventh Safety Critical Systems Symposium,* Huntington, U.K. T. Anderson and F. Redmill (Eds.). Springer-Verlag. 1999.

Hanks, Kimberly S., and John C. Knight. "Improving Communication of Critical Domain Knowledge in High-Consequence Software Development: An Empirical Study." 21st International System Safety Conference, Ottawa, Canada, August 2003.

Johnson, C.W. *Failure in Safety-Critical Systems: A Handbook of Accident and Incident Reporting.* Glasgow: University of Glasgow Press. 2003.

Kelly, T.P. "Arguing Safety—A Systematic Approach to Managing Safety Cases," Ph.D. diss. University of York. 1998.

McDermid, J.A. "Software Safety: Where's the Evidence?" *Proc. 6th Australian Workshop on Industrial Experience with Safety Critical Systems and Software (SCS '01),* Brisbane, Australia. P. Lindsay (Ed.). Conference in Research and Practice in Information Technology, Vol. 3. 2001.

McDermid, J.A. and Kelly, Tim. "A Systematic Approach to Safety Case Maintenance." *Proc. International Conference on Computer Safety, Reliability, and Security (SAFECOMP '99),* Toulouse, France. 1999.

National Transportation Safety Board. *Controlled Flight Into Terrain, Korean Air Flight 801, Boeing 747-300, HL7468, Nimitz Hill, Guam, August 6, 1997.* Aircraft Accident Report NTSB/AAR-00/01. Washington, DC, 2000.

RTCA. "Software Considerations in Airborne Systems and Equipment Certification," document RTCA/DO-178B. Washington, DC: RTCA, December 1992.

Weaver, R.A. "The Safety of Software: Constructing and Assuring Arguments," Ph.D. diss. University of York. 2003.

TOWARD A HUMAN-CENTERED UML FOR RISK ANALYSIS

Application to a medical robot

Jérémie Guiochet[1], Gilles Motet[2], Claude Baron[2] and Guy Boy[3]

[1]Grimm-Isycom/Lesia, Université de Toulouse II, 5 al. A. Machado, 31100 Toulouse, France; [2]LESIA, INSA DGEI, 135 av. de Rangueil, 31077 Toulouse, France; [3]EURISCO International, 4 Av. E. Belin, 31400 Toulouse, France

Abstract: Safety is now a major concern in many complex systems such as medical robots. A way to control the complexity of such systems is to manage risk. The first and important step of this activity is risk analysis. During risk analysis, two main studies concerning human factors must be integrated: task analysis and human error analysis. This multidisciplinary analysis often leads to a work sharing between several stakeholders who use their own languages and techniques. This often produces consistency errors and understanding difficulties between them. Hence, this paper proposes to treat the risk analysis on the common expression language UML (Unified Modeling Language) and to handle human factors concepts for task analysis and human error analysis based on the features of this language. The approach is applied to the development of a medical robot for tele-echography.

Keywords: safety; risk analysis; system modeling; UML; task analysis; human error analysis; medical robot.

1. MOTIVATIONS

Today systems being more complex, and more responsibilities being transfered to them [1], safety requirement is becoming critical. Safety, previously defined as an absolute property [2], is also now expressed in a relative and probabilistic way as the property of a system to be "free from unacceptable risk" [3]. Therefore it is necessary to reduce the risk to an acceptable level with a complete risk management process [4], including

activities presented on the left part of the figure 1. This approach has been used

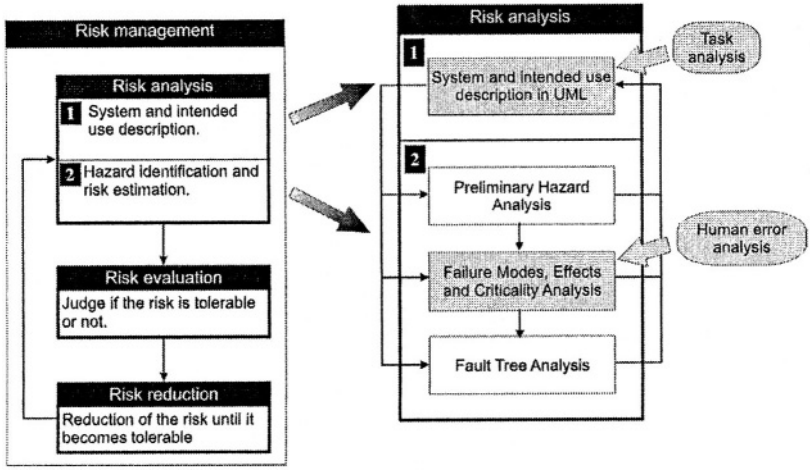

Figure 1. Human factors and UML based risk analysis in the risk management activity

into different domains. For example, some of its concepts can be found in the medical standards [5]. Inside the general risk management activity, our study focuses on the first step: the risk analysis. This step aims at identifying hazards and estimating their associated risk (probability and severity). During this phase, various techniques can be used to handle functional and technological issues such as Preliminary Hazard Analysis (PHA), Failure Modes, Effects and Criticality Analysis (FMECA) and Fault Tree Analysis (FTA) techniques [6,7] as presented on the right part of figure 1.

The interaction between human and technological systems plays a major role in safety. Nevertheless, the integration of human factors in the risk management standards is still in work [8,9]. Based on this research, we focus on two main activities of human factors studies which are particularly important during risk analysis: “task analysis” for which the system and its intended use are described and “human error analysis” to identify new hazards and estimate their risks. These two phases are presented on the right part of figure 1. The second phase is implemented using FMECA [10].

Both activities are based on a system model. Ideally, the system definition is formally modeled. In practice, the use of formal methods in industrial development is still rare. A significant barrier is that many formal languages and analysis techniques are unfamiliar and difficult to apply for engineers. Moreover, several modeling tools have to be used to treat particular and partial aspects of the system. Designers must also communicate between specialists of different domains who usually have

their own language. To handle this issue, we considered UML (Unified Modeling Language), which is now a standard in system and software engineering, even if this language presents several drawbacks (for instance, it has no formal semantics).

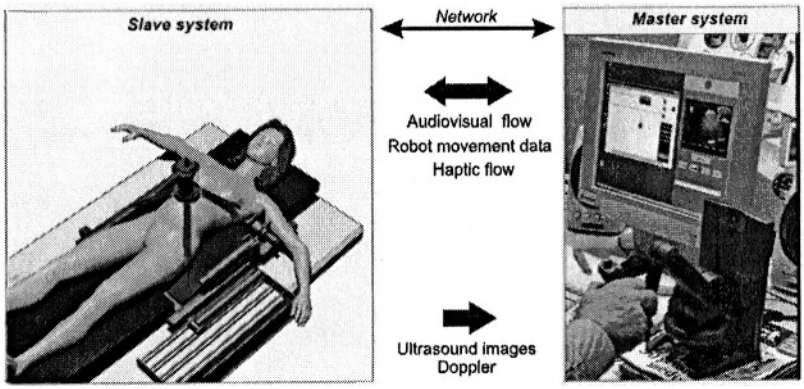

Figure 2. TER system overview

This paper presents how task analysis and human error analysis can be integrated in risk analysis and how UML can be useful to perform them. We will present some UML features to graphically specify tasks and to analyze them. Thus, section 2 exposes task analysis during system definition in UML and section 3 proposes an approach based on FMECA and UML message error models applied to human error analysis.

Each section is illustrated on a system for Robotic Tele-Echography (TER) [11]. TER is a tele-robotic system designed and developed by a French consortium composed of universities, hospitals and industrial companies. The slave robot is tele-operated by an expert clinician who remotely performs the ultrasound scan examination. A virtual probe is mounted on the master interface device. The real probe is placed on the slave robot end-effector. The slave robot is actuated with artificial muscles (pneumatic actuators). An overview of TER is provided on figure 2. We will focus on the computer control system of the slave site, whose safety is critical.

1. TASK ANALYSIS DURING SYSTEM DEFINITION WITH UML

Task analysis aims at identifying the details of specified tasks, including the knowledge, skills, attitudes, and personal characteristics required for successful task performance. During system analysis, this activity is linked

to task allocation which aims at determining the distribution of work between human actors and machines. For instance, it is particularly important to define non ambiguous and consistent tasks for humans who are using the robot.

1.1 Related work

These activities are usually performed with different algorithms ([12,pp.231-236], [13] and [14]). Although there are a variety of techniques [15], the integration of task analysis in system modeling is still under development [16]. In this regard, many workshops aim at integrating human factors in system modeling [17], and more particularly in object oriented modeling [18,19]. Many studies compare *use cases* (based on Cockburn [20] definition which is closed to Jacobson's one [21]) and task analysis for interactive systems [22,23]. This was also applied for medical robots [24,25]. In those studies, use cases are usually derived from existing task analysis, and often led to the *business modeling* [26] like in [27]. Other authors study how to correlate task analysis
and object oriented concepts, in order to model tasks themselves [28,29,30] for further human machine interface design.

Most of those theories are developed to design user interfaces. Our purpose is to provide a method to prevent hazards due to a bad task definition and allocation but also to provide models for human error analysis. This led us to analyze and model tasks with interaction diagrams, even if this "scenario-based" approach is sometimes opposed to "task-based" analysis as discussed in [31].

1.2 Business modeling

We first model the business without the technological system (ultrasound scan examination), with UML use cases and interaction diagrams *(collaboration* and *sequence diagrams).* During this step, "business modeling increases the understanding of the business and facilitates communication about the business" [26], particularly between engineers and doctors. For the considered example, the use case diagram in figure 3 models the common ultrasound scan examination without the robot system. Based on this diagram, the TER system is later integrated in the requirement modeling in the next diagrams.

Structuring the business with use cases helps the designers for the task allocation. For each use case, a textual description specifies more precisely the possible scenarios and their conditions of execution. Even if the use case *Perform Ultrasound Scan* seems to be the most important for the design,

three other use cases have to be analyzed which can be later critical for the safety. Indeed, during the ultrasound scan examination the specialist can simultaneously manage the probe (change the settings), interact with the patient (communicate, prepare the surface to scan, etc.) and even diagnose. Hence, the future system should allow to perform all those use cases safely.

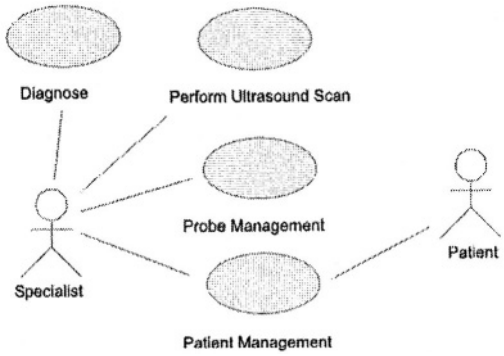

Figure 3. Use case diagram: global view of ultrasound scan examination

This notation completed with textual documents (or even task analysis studies) improves the communication between medical specialists and system designers. Based on this diagram, it is easier to allocate tasks between humans and systems. From the system analysis viewpoint, this documentation leads to design choices such as the system architecture presented on figure 2 (master and slave sites, bidirectional communication, etc.). Considering that the future system will integrate a robot, the other business is the use of a robot. This led to identify two generic use cases which are *Perform a task* and *Robot management,* but also two actors: the *Operator* and the *Robot* itself.

1.3 From business modeling to robotic system modeling

In this phase, we integrate the use cases of a robotic system into the use case diagram of the ultra sound scan examination (figure 3). This led to modify specifications of previous use cases. New actors are then identified. An *actor* characterizes an outside user or related set of users who interact with the system [4], It is possible for an *actor* to be a human user (like the *Specialist* in the previous section) or an external system. This is really useful in socio-technical systems, and particularly in the TER project. We choose to represent two external systems as actors: the *Master Site* and the slave *Robot.* The *Master Site* replaces the actor *Specialist* (see figure 3) who is in

charge of performing the examination. It is important to observe that the use case *Diagnose* has also disappeared, being transfered to the master site.

The use case diagram of figure 4 shows an allocation of work between actors. On this diagram, the boundaries of the computer control system are defined. It has been determined "which of the requirements are system requirements, which are requirements for the operational processes associated with the system and which requirements should be outside the scope of the system" [33]. This means that we have decided for each use case if it belongs or not to the system. For instance, the use case *Probe Management* has been removed from this use case diagram because it does not belong to or has any interaction with the computer control system

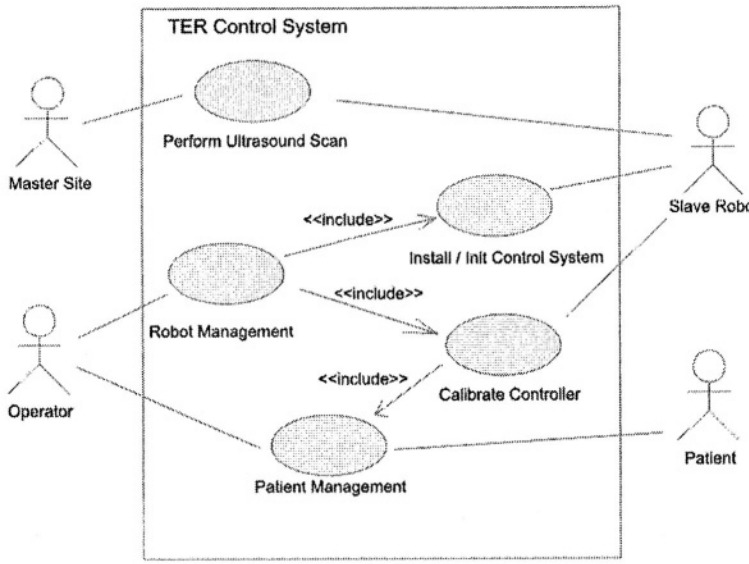

Figure 4. Use case diagram with Control System boundaries

1.4 Tasks description

We chose to specify tasks and subtasks with the UML concept of *message.* On sequence diagram figure 5, the main scenario of the use case *Install/Init Control System* is presented. This diagram can also be refined. For instance the *Operator* has to *Prepare Patient,* which can be detailed in: position the patient, put ultrasound scan gel on patient's body, give information to the patient, monitor the patient, etc. This notation of tasks is also useful to specify a sequence order, which can be essential for safety. By definition, sequence diagrams just specify possible scenarios (descriptive models). Nevertheless we use those diagrams as prescriptive models to

establish a safe order of messages, because they are easily readable by non experts of UML modeling.

These models can directly be used for different safety-dependent tasks: writing of a user-guide (using the sequence diagrams), specifying and designing the Human-Machine Interface (HMI) and furnishing models for the specification of the system. It is important to note that in such robot systems, HMI includes the robot-human interface (control panels, teach pendant, etc.), but also the robot itself (in the TER project the slave robot is always in contact with the patient's body).

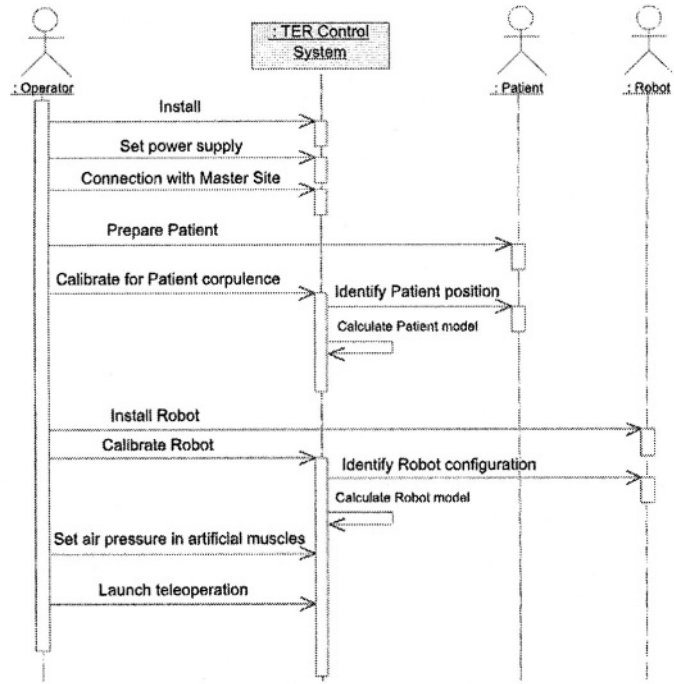

Figure 5. Sequence diagram of installation of the whole system

2. UML BASED FMECA FOR HUMAN ERROR ANALYSIS

As a potential source of harm (a hazard), human error has to be analyzed during the step of hazard identification and risk estimation. Although there is a variety of techniques (the most relevant and complete technique is certainly the Technique for Human Error Rate Prediction [34,35]) and tools [36], the complexity of human error classification and cognitive theory [37] usually leads engineers to the use of design checklists and guidelines [2] for

the design of human computer interfaces. Nevertheless, as noted in [38], guidelines are not sufficient for innovative projects as medical robots.

2.1 Message failure mode analysis

The notion of failure mode is close to the notion of error; both concepts will be indifferently used in this section. In order to perform human error analysis, we have based our approach on several points. First, to be consistent with the previous section, we focus on a task-based analysis, quickly usable for non-specialist of human error. As proposed in [39], we do not have ethnographic studies and cognitive task analysis to perform the analysis. Thus, we only based our analysis on a set of models of scenarios, interface proposals, and human errors models.

Second, in order to reduce the number of analysis and modeling techniques, we propose to perform the human error identification and analysis with the well known analytical method FMECA. Among analytical methods allowing fault forecasting, FMECA [10] is certainly the most used during functional analysis. In an object oriented model, actors are represented as objects sending messages to the system. Hence, FMECA has to be conducted based on object concepts. This has been applied to object oriented elements such as components software [40], object methods [41], or use cases [42]. In those cases, the authors perform analysis focusing on functional aspects of object oriented models. On the contrary, the main idea of our approach is to propose an object oriented FMECA as we have previously done [43], and to apply it to objects such as actors.

The FMECA technique consists at first in identifying errors. These errors are often specific to the application. However, to realize a more systematic error identification step, one can sometimes use some generic error models. Those error models are related to generic elements of the system. In our approach we chose to focus on a central element of the UML dynamic diagram: the *message.* The concept of *Action* is also an important feature of UML to describe behaviors. But we did not handle this feature because its semantics changed a lot from version UML specification 1.4 [44] to 1.5 [45] and now to 2.0 [46].

2.2 Message error models

Most of language specifications contains *operational semantics* as well as *verification semantics.* The operational semantics is used to specify system functional aspects and to describe how the system will deliver the service. Most of UML diagrams belongs to operational semantics. The verification semantics defines properties associated with the correct use of

features of the language. Some elements in the UML specification belong to the verification semantics. For instance, the use of constraints, graphically represented with curly brackets, allows to specify a restriction on a modeling element.

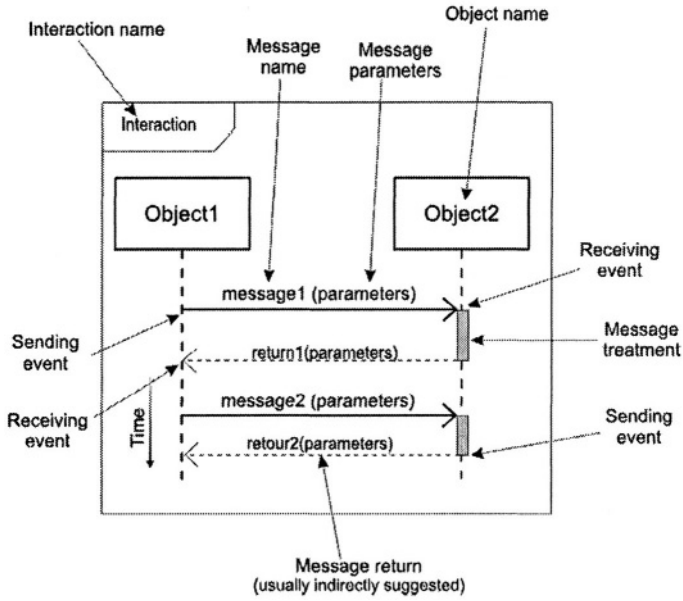

Figure 6. Elements of an interaction realized by the exchange of two messages

There are also in UML the *Well-Formedness Rules,* which define a set of constraints expressed with the OCL language [44]. These constraint violations specify generic errors. However, most of verification properties are not explicitly provided in the UML standard. They are often implicitly integrated into the operational semantics, and thus they have to be deduced studying each feature of UML. As previously mentioned, our study focused on the *Message* feature.

A message can be a signal creation, an operation call, a creation or destruction of an instance. The graphical representation by a sequence diagram is illustrated on figure 6. The different elements of a message are: the interaction it belongs, the next and previous messages in the interaction, the objects that send and receive the message, the sending and receiving events, the parameters (number, type and value), the implicit response (defined by its arguments, sending and receiving events), and the period of the message treatment.

Based on those elements we identify eleven error models (see [43] for details):

E.1. Sending of a message not belonging to the planned interaction.

E.2. Execution of one or several messages in a wrong order.
E.3. Omission of a message among an interaction.
E.4. Lack of an instance to receive the message.
E.5. Sending or receiving of a message outside its specified time limits (too soon or too late).
E.6. The arguments type is different from the type of parameters expected by the receiver.
E.7. The number of message arguments is different from the number of parameters expected by the receiver.
E.8. The value of message arguments is different from the value of parameters expected by the receiver.
E.9. The values returned by a response to a message do not fit with the expected values (for example: constant, random, out of limits, etc.).
E.10. Treatment of a message out of the specified time limits.
E.11. Lack of link between sender and receiver objects.

2.3 Proposition of a generic FMECA array for a system analysis

The error models been specified, their effects on system harm risk have to be studied. To handle this activity, we tuned the FMECA array [10] (originally devoted to functional analysis). This section proposes to introduce the following elements into the FMECA array for a message failure mode analysis (see figure 7): the interaction or the message name, the failure modes or the errors identified thanks to the previous error models, the causes of those failure modes, the effects at a local, higher or system level, the data to estimate the risk *(severity* is the harm seriousness, and failure mode occurrence is noted as *probability),* the on-line means to detect failure modes and their effects, the possible means for risk prevention and protection and other pieces of information.

Note that the goal in these arrays is not to proceed to a deep analysis of each of the mentioned points; in particular, the aim is not to consider the causes of the causes but to synthesize the main data in order to obtain a system analysis.

The *Potential solutions* of the array deal with the possible means to reduce the risk. It is important to notice that these means are not directly implemented but this highlights that a preliminary risk evaluation must be done. Risk is here calculated from a qualitative estimation of the probability of occurrence of a failure mode and of the severity of the induced harm. We chose to represent the prevention and protection means in order to reduce the probability or the severity of the considered harm.

This FMECA was essentially useful to focus on critical and weak design points from the safety point of view. Moreover, as FMECA directly depends on the model level of details, its use depends on the development process step. In our approach, we recommend to concentrate on the first steps, when safety requirements, architecture choices and major hazards are identified.

Interaction/ Message	Failure mode (error)	Effects a. Same level b. Upper level c. System level	Risk			Possible detection means (online): a. Failure mode b. Effects	Potential solutions: a. Prevention b. Protection c. Other actions d. Remarks
			Severity	Probability	Risk		
Install/Init Control System:: **Set air pressure in artificial muscles**	Omission (E.3)	a. No power supply in artificial muscles b. No movements c. Patient waiting (stress)	4	P	I	b. Pressure sensor	a. Detailed user manual, formation, detailed actions on a screen b. Make a pumping test before launch teleoperation
	Wrong order (E.2) : before *Set power supply*	No initialization of Control System outputs. When power on: a. Spike of an output b.Uncontrolled movement c.Harmful movement for operator (patient not installed)	2	P	H		a. Detailed user manual, formation, detailed actions on a screen b. Intelock system (to be defined)
	Pressure too high (E.8)	a. Reach the limit of intensity/pressure converters (to be determined) b. Partial or complete destruction c. Uncontrolled and harmful movements for patient	1	O	H	a. The operator check the pressure on a manometer	a. Indications on the manometer (close to the button) b. Pressure regulators before artificial muscles

Figure 7. Example of a table of FMECA for the message "Set air pressure in artificial muscles"

2.4 Application to the analysis of messages sent by actors

This section presents an example of use of error models previously identified (section 3.2) in order to demonstrate the tractability of the analysis proposed in section 3.3 for human error analysis. This approach has been successfully applied to medical robot system TER as presented in [6].

2.4.1 Types of errors

Merely all the error models previously identified can be applied for human error analysis. Common errors are the occurrence of an action of the actor not belonging to the planned interaction (error E.1), the execution of actions in a wrong order (E.2), and the omission of an action during an interaction (E.3). It is also possible to note human errors such as E.6, E.7 and E.8 consisting in furnishing bad data to the system. For instance, a user can type a letter whereas the system is waiting for a number (E.6), or he/she can tune a pressure valve too high for the system (E.8). The error E.4 is rather rare in human error analysis because it implies that the object for the interface is absent. The error E.5 depends on the time constraints a system

can have, and is based on non functional requirements as for the error E.10. The error model E.9 which concerns response of a message (return values) is really useful for software or electronic components analysis. In case of the human component this is equal to E.6, E.7 and E.8, for message coming from the system.

2.4.2 Failure mode analysis

The modeling of all exchanged messages with UML sequence diagrams allows to perform an analysis very soon in the development. It can lead to formulate safety requirements from the start, without detailing design choices. Types of error are integrated in tables of an FMECA analysis in the column "Failure modes". For instance, we consider the message *Set air pressure in artificial muscles* from figure 5. As shown in figure 7, we identify three failure modes (the number has been reduced to present this example) from error models. In order to determine other columns data, we have to study all the UML models such state diagrams and class diagrams. Those diagrams are not presented here but can be found in [6,25]. For instance, those diagrams are used to determine effects of the failure modes on actors (column "Effects"). We proposed to use a scale for harm severity with five levels: negligible (5), minor (4), major (3), sever (2), catastrophic (1). Then, during a FMECA, it is easier to estimate the probability of the failure mode leading to the harm rather than the probability of the harm itself. Considering that a quantitative evaluation of the probability of occurrence of a human error is impossible to perform, we only do a qualitative estimation with different levels of probability of occurrence: frequent, probable, occasional, rare, and impossible. This point has to be developed, and relied to our type of errors. We have determined types of human errors that can appear in a human-machine interaction, but the causes are not integrated. In this table it is possible to highlight some important data missing for the analysis (like the maximum limit of air pressure in the converters).

3. CONCLUSIONS AND PERSPECTIVES

The risk analysis approach proposed in this paper is motivated by the growing system complexity and safety requirements. At this stage, the human-centered approach of UML is twofold: a scenario-based task analysis and a message-based human error analysis.

We have shown that a scenario-based analysis performed throughout use case modeling helps designers in describing tasks. UML diagrams are

initiated by UML specialists and further proposed to the other actors of the development process. This approach leads to a more consistent task allocation and to produce models that are useful in subsequent development steps.

Eleven error models have been presented. They are related to the concept of message in UML. In this paper, the object-oriented approach is linked to the FMECA functional risk analysis technique. Error models are integrated into FMECA. The resulting approach enables the various actors of the development process to use the same models.

This approach was applied successfully to the development of a first prototype of a medical robot for tele-echography. Others studies will be performed in different fields to complete and validate this work. The next technical step would be the development of tools to automatically integrate FMECA to UML design diagrams. We also need to go further in human error modeling to provide diagrams to understand how our types of error can be generated. Finally, a complete error model associated with the UML features is under development.

REFERENCES

[1] G. Motet and J.C. Geffroy. Dependable computing: an overview. *Theoretical Computer Sciences,* 290(2): 1115-1126, 2003.

[2] N.G. Leveson. *Safeware - System safety and computers.* Addison-Wesley, 1995.

[3] ISO/IEC Guide 51. Safety aspects - Guidelines for their inclusion in standards. International Organization for Standardization, 1999.

[4] ISO/IEC Guide 73. Risk management - vocabulary - guidelines for use in standards. International Organization for Standardization, 2002.

[5] ISO 14971. Medical devices - Application of risk management to medical devices. International Organization for Standardization, 2000.

[6] J. Guiochet. *Safety management of service robot systems - UML approach based on system risk analysis (in french).* PhD thesis, Institut National des Sciences Appliquées de Toulouse, France, 2003.

[7] J. Guiochet and A. Vilchis. Safety analysis of a medical robot for tele-echography. In *Proc. of the 2nd IARP IEEE/RAS joint workshop on Technical Challenge for Dependable Robots in Human Environments, Toulouse, France,* pages 217-227, October 2002.

[8] Food and Drug Administration. Medical device use-safety: incorporating human factors engineering into risk management. Technical report, U.S. Departement of Health and Human Service, 2000.

[9] HSE. Proposed framework for addressing human factors in IEC 61508. Technical Report 373/2001, Health and Safety Executive, UK, 2001. http://www.hse.gov.uk.

[10] MIL-STD-1629A. Procedures for performing a Failure Mode, Effects and Criticality Analysis. Military Standard, 1980.

[11] A. Vilchis, P. Cinquin, J. Troccaz, A. Guerraz, B. Hennion, F. Pellissier, P. Thorel, F. Courreges, A. Gourdon, G. Poisson, P. Vieyres, P. Caron, O. Mérigeaux, L. Urbain, C. Daimo, S. Lavallée, P. Arbeille, M. Althuser, J-M. Ayoubi, B. Tondu, and S. Ippolito.

TER: a system for Robotic Tele-Echography. In *4th Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI'01),* volume 2280 of *Lecture Notes in Computer Science,* pages 326-334. Springer, 2001.

[12] J-C. Laprie, J. Arlat, J-P. Blanquart, A. Costes, Y. Crouzet, Y. Deswarte, J-C. Fabre, H. Guillermain, M. Kaâniche, K. Kanoun, C. Mazet, D. Powell, C. Rabéjac, and P. Thévenod. *Dependability handbook (in french).* Cépaduès - Éditions, Toulouse, France, 1995.

[13] D. Beevis, R. Bost, B. Döring, E. Nordø, F. Oberman, J-P. Papin, H. Schuffel, and D. Streets. Analysis techniques for man-machine systems design. Technical Report AC/243(Panel 8)TR/7, NATO, Canada, 1994.

[14] M. Mersiol, C. Mazet, H. Guillerman, and H. Waeselynck. Human dependability in complex system: an issue of task consistency and task allocation. *International Conference on Probabilistic Safety Assessment and Management (PSAM'4),* 4:2693-2698, September 1998.

[15] F. Paternó. *Model based design and evaluation of interactive applications.* Springer Verlag, 2000.

[16] N.J. Nunes. *Object Modeling for User-Centered Development and User Interface Design: The Wisdom Approach.* PhD thesis, Universidade Da Madeira, Madeira, Portugal, April 2001,

[17] CHI97. Conference on Human Factors in Computing Systems, Atlanta, USA. ACM, 1997. http://www.acm.org/sigchi/chi97.

[18] A. Seffah and C. Hayne. Integrating human factors into use cases and object-oriented methods. In *Proc. Workshop on Integrating Human Factors into Use Cases and OO Methods (WISDOM'99) in the 13th European Conference for Object-Oriented Programming (ECOOP'99),* volume 1743 of *Lecture Notes in Computer Science,* pages 240-254. Springer-Verlag, 1999.

[19] TUPIS2000. Towards a UML Profile for Interactive Systems Development Workshop in the <<UML2000>> International Conference, York, UK. Online, 2000.

[20] A. Cockburn. Structuring uses cases with goals. *Journal of Object Oriented Programming,* 8(6/7), 2000.

[21] I. Jacobson. *Object-oriented software engineering: a use case driven approach.* Addison-Wesley, 1992.

[22] P. Forbrig and A. Dittmar. Relations between uses cases and task analysis. In *Proc. of Workshop on Integrating Human Factors into Use Cases and OO Methods in ECOOP'99,* 1999.

[23] C. Hayne, A. Seffah, and D. Engelberg. Comparing uses cases and task analysis: a concrete example. In *Proc. of Workshop on Integrating Human Factors into Use Cases and OO Methods in ECOOP'99,* 1999.

[24] M. Lee and H.A. Abdullah. Applying UML to task analysis of the user interface for rehabilitation robotic system. In *Proc. of 8th International Conference on Rehabilitation Robotics, Daelon, Korea,* 2003.

[25] J. Guiochet, B. Tondu, and C. Baron. Integration of UML in human factors analysis for safety of a medical robot for tele-echography. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS03),* pages 3212-3218. IEEE Publisher, October 2003.

[26] H.E. Eriksson and M. Penker. *Business modeling with UML: business patterns at work.* John Wiley and Sons, Inc., 2000.

[27] Y. Barnard and I. Blok. Gathering user needs for knowledge management applications for engineers in advanced manufacturing industries. In S. Bagnara, editor, *Proc. of the 8th*

International Conference on Human Aspects of Advanced Manufacturing: Agility & Hybrid Automation, pages 339-346, 2003.

[28] S.Wang. Object-oriented task analysis. *Information and Management,* 29:331-341, 1995.

[29] P. Markopoulos. Modelling user tasks with the Unified Modelling Language. In *Proc. of Workshop TUPIS2000 in <<UML2000>> International Conference,* York, UK, 2000.

[30] M. Abed D. Tabary. A software environment task object-oriented design (etood). *The Journal of Systems and Software,* 60:129-140, 2002.

[31] F. Paternó. Commentary on 'scenarios and task analysis' by Dan Diaper. *Interacting with computers,* Elsevier, 14:407-409, 2002.

[32] G. Booch, J. Rumbaugh, and I. Jacobson. *Unified Modeling Language Users Guide.* Addison Wesley Longman, 1999.

[33] I. Sommervile and P. Sawyer. *Requirements engineering : a good practice guide.* John Wiley and Sons, Inc., 1997.

[34] G. Hannaman and A. Spurgin. Systematic human action reliability procedure (SHARP). Project 2170-3, Interim report EPRI NP-3583, NUS Corporation, San Diego, CA, US, 1984.

[35] A. Swain and H. Guttmann. Handbook on human reliability analysis with emphasis on nuclear power plant application. NUREG/CR-1278 SAND 80-0200 RX, Nuclear Regulatory Commission, Washington, US, 1983.

[36] C. Kelly, P. Enterkin, and P. Goillau. Human factors integration in future ATM systems - methods and tools. Technical Report HRS/HSP-003-REP-03, Eurocontrol, European Organisation for the Safety of Air Navigation, 2000.

[37] J. Reason. *Human Error.* Cambridge University Press, 1990.

[38] P. Wright, B. Fields, and M. Harrison. Deriving human-error tolerance requirements from tasks. *IEEE International Conference on Requirements Engineering (ICRE'94),* 1:462-467, 1994.

[39] P.C. Cacciabue. Human error risk management for engineering systems: a methodology for design, safety assessment, accident investigation and training. *Reliability Engineering and System Safety,* 83(2):229-240, 2004.

[40] S. Yacoub, H. Ammar, and T. Robinson. A methodology for architectural-level risk analysis. In *11th International Symposium on Software Reliability Engineering (ISSRE'2000), San Jose, CA,* pages 210-221, October 2000.

[41] F. Bitsch. Requirements on methods and techniques in perspective to approval process for railway systems. In *Second International Workshop on Integration of Specification Techniques for Applications in Engineering (INT 2002), Grenoble, France,* April 2002.

[42] P. Johannessen, C. Grante, A. Alminger, U. Eklund, and J. Torin. Hazard analysis in object oriented design of dependable systems. In *2001 International Conference on Dependable Systems and Networks, Göteborg, Sweden,* pages 507-512, July 2001.

[43] J. Guiochet and C. Baron. UML based FMECA in risk analysis. In *Proc. of the European Simulation and Modelling Conference ESMc2003, Naples, Italy,* October 2003.

[44] OMG. Unified Modeling Language Specification v1.4. Technical report, Object Management Group, September 2001.

[45] OMG. Unified Modeling Language Specification v1.5. Technical Report formal/03-03-01, Object Management Group, March 2003.

[46] OMG. 2nd revised submission to OMG RFP ad/00-09-02 - Unified Modeling Language : Superstructure - version 2.0. Technical Report ad/2003-01-02, Object Management Group, January 2003

This page intentionally left blank

HANDLING HUMAN FACTORS IN INTEGRATED SYSTEMS ENGINEERING

Coping with Context-Adaptive Behavior

Michael Cebulla

Technische Universität Berlin, Fakultät IV, Institut für Softwaretechnik und theoretische Informatik, mce@cs.tu-berlin.de.

Abstract: In this paper we focus on architectural concepts for complex sociotechnical systems and advanced pervasive applications which have to be highly context aware. First we claim that there is a great need for model-based reasoning about systemic properties concerning questions of system design and the definition of long-term management policies. After this we take our starting point from formal methods, requirements engineering, and software architecture. We provide special extensions for these methods which are well-suited for the special challenges of sociotechnical systems: adaptive behavior and the behavioral relevance of cognitive parameters. We maintain the visual style of modeling concepts as known from software architecture and provide elements of an easy to use notation for reasoning about the features of specific situations. Finally we provide concepts to deal with uncertain system behavior and human error.

Key words: Methodologies, Context Models, Human Error.

1. INTRODUCTION

Modern system engineering is increasingly confronted with a new quality of contextual embedding. For the specification of systemic behavior a growing number of environmental parameters have to be taken into account. We conceive these parameters as systemic aspects and use ontologies to achieve a modular way to manage the related knowledge. In this paper we focus on human factors as a major contextual parameter.

Sociotechnical systems as well as advanced applications like pervasive services have to reside in complex contexts and frequently have to deal with

unforeseen situations which may lead to error situations. Hence their behavior is deeply interwoven with numerous parameters determined by the external environment. One important special case of contextual dependency is represented by the increased significance of human machine interaction. Advanced pervasive applications with multimodal interfaces will have to be able to maintain complex hypotheses about users and their situation in order to be of any practical value. This new type of application has to provide a behavior which is far more flexible than traditional computer systems. Apart from being *context aware* in some general sense they have to be aware of the users identity, their capabilities, their goals, and their plans.

Consequently, from the perspective of integrated systems engineering context dependent human machine interaction has to be conceived as a systemic aspect which has gained increased relevance. In addition, in the light of a more intensive aspectual interweaving concepts for human error modeling have to be compatible with concepts for the description of other aspects. On the basis of this compatibility *complex interactions* (Leveson, 1995; Perrow, 1984) between different types of systemic agents can be analyzed. These interactions are a prominent cause of system failures and losses.

In this paper we focus on the integration of two important aspects of system modeling. By providing *ontological concepts* for the aspects of workflow and human factors we gain the advantages of modularity, compositionality, and conceptual reusability. On this platform we can integrate concepts related to the description of cognitive states into workflow description.

We claim that our conceptual frameset allows for an integrated view on system behavior with specific consideration of human factors. Since our concepts are abstract and located on the level of conceptual modeling they are equally well suited for the description of several types of hybrid systems where teams of humans cooperate with ensembles of devices. In addition, we claim that our approach based on conceptual modeling is well-suited to specify insights about interactive behavior in sociotechnical systems and transfer them to the design of context aware systems in general.

2. CONCEPTUAL MODELING

Originally, we developed our concepts for the analysis and modeling of complex sociotechnical systems. Due to the complexity of these systems their analysis and understanding is very difficult. But on the other hand for various reasons (safety, efficiency, organizational learning, change management) there is a strong need for a greater transparency of the related

processes. This is shown vividly by the great number of disasters, losses and catastrophes during the recent decades.

In our approach we develop a visual notation for the modeling of complex sociotechnical systems. Starting from formal methods and the experiences of *requirements engineering* (Partsch, 1998; Pepper, Wirsing, 1995) we provide concepts which are well suited for the description of specific sociotechnical features. Especially the aspects of variability and adaptation but also those of human cognition and organizational relations are traditionally hard to grasp by formal notations. In addition, uncertain system behavior and human error are topics of great relevance.

We claim that a visual modeling notation significantly increases system transparency, support interdisciplinary system analysis, and is well-suited to support measurements of further education. On the long run we plan to provide automated tools for analysis, simulation and the support of system management. For the semantic foundation of our notation (which is beyond the scope of this paper) we use description logic (Baader et al., 2003).

In addition, we claim that our approach of conceptual modeling provides a platform for a better understanding of the impact of human factors on safety-critical features of complex systems in general. The resulting conceptual models of social interaction, organizational dependencies, and human error are equally well-suited as foundations for the design of pervasive and adaptive systems. Since these advanced services have to adapt smoothly to changing contexts they have to possess similar adaptive capabilities as traditional sociotechnical systems. We think that conceptual models of adaptive and context-aware behaviors are well-suited as transfer media for these capabilities. Generally, these conceptual models are major contributions for a better understanding of complex systems and an enhancement of their safety-related properties.

We describe the motivation for our work in sections 2 and 3. In section 4 we identify some specific sociotechnical challenges for model-based methods. While we explain the foundations of our method in sections 5 and 6, we apply our concepts on specific sociotechnical features in sections 7 to 9.

3. COMPLEXITY AND SAFETY

Complex sociotechnical systems have evolved to control high risk technologies by teams of highly qualified specialists. Sociotechnical systems can be defined as *complex* safety-critical systems where *teams* of human operators cooperate with *ensembles* of technical units and devices. Usually, the resulting processes are significantly more complex than in traditional

systems consisting solely of technological components. Generally, their behavior is more dependent on contextual changes. Thus, a model for these kinds of systems has to take into account not only technology but also contextual parameters as in our case human factors. Examples for this kind of systems are not only atomic power plants, medical operation theaters and air traffic control, but also safety-critical pervasive applications, which we expect to evolve in the near future.

This new classes of systemic complexity, contextual determination and its related risks have established new requirements for system design and system safety. This is documented by the sad history of catastrophes from Three Miles Island (1979) to Überlingen (2002). The analysis of such complex systems has proven too multi-faceted for the traditional single-disciplinary approach.

A model-based interdisciplinary system analysis is a promising strategy against what Leveson calls intellectual unmanageability of high risk systems (Leveson, 1995). The increasing complexity and tight coupling in contemporary high risk systems make a safe and efficient management difficult if not impossible. The main source of failure in complex systems is not human error or an erroneous component, but the *complex interactions* between components which is not understood to a sufficient degree. To increase the level of understanding we choose a model-based approach which is open for results of interdisciplinary research (i.e. from disciplines like semiotics, psychology or sociology).

4. CHALLENGES OF SOCIOTECHNICAL SYSTEMS

Of course, the methods of system modeling are not new. Especially in *software engineering* concepts and methods were synthesized to handle the challenges of complexity in development processes. Consequently, in our work we heavily rely on the results of system architecture, formal specification and systems engineering (Sage, 1999). The concepts and methods from requirements engineering provide the basic means to manage informal and semiformal knowledge about the target system.

We observed some specific features of sociotechnical system which can be conceived as challenges for traditional modeling concepts. We claim that the traditional modeling concepts and formal methods as known from software engineering have to be adapted and extended for the specific properties of sociotechnical systems. Speaking generally, just the merits of concepts for formal specification as exactness and well-definedness sometimes prove as shortcomings in the context of sociotechnical systems. It

is a generic feature of these systems that they have to deal with vague data, uncertainty and incomplete specifications. Modeling concepts have to adapt to this specific vagueness which can be conceived as important system quality. Paradoxically speaking, too much exactness would lead to less adequate or even wrong specifications.

4.1 Uncertainty

Sociotechnical systems tend to reduce the load of information processing by using vague concepts. So human experts normally don't use exact mathematical expressions (like for example partial differential equations) but vague expressions from natural languages. We claim that the resulting vagueness is an important precondition for the systems robustness and safety since vague specifications are compatible with changing contexts.

In addition, human actors frequently have to deal with incomplete specifications. In many situation relevant information is not accessible for them. Due to situational time pressure they have to make uncertain decisions based on incomplete information (Cebulla, 2003). In our approach we use fuzzy sets and fuzzy logic to specify uncertain information and vague relations (Klir, 1995).

4.2 Adaptive Behavior

One important feature of sociotechnical systems is their *structural dynamism.* The internal structure of systems like the medical operation theatre can be rapidly changing from one phase of the process to another according to environmental changes. For the description of this structural dynamism powerful concepts from *dynamic architectures* (Pepper, 2003) are necessary. We handle this problem by introducing *transformation rules.*

By specifying transformation rules we define the possibilities of configurations to evolve under changing context conditions. Hence transformation rules describe the adaptive behavior of complex sociotechnical systems in specifying the way these systems react to environmental changes by structural mutation (for an example cf. section 7).

Sociotechnical systems like the medical operation theatre have distinctive qualities regarding their adaptive capabilities. Thus, their ability to recover in the face of adverse environmental conditions or unexpected events is clearly larger compared to the behavior of traditional component-based systems. Regarding this adaptivity a deeper understanding of sociotechnical processes can contribute to the robustness and flexibility of software-based systems.

4.3 Subjective Evidence

Domain experts tend to disagree about the facts in their field. Sometimes they are not completely sure. We use fuzzy concepts for the representation of different degrees of subjective evidence and relevance. This enables us to distinguish for example between different degrees of adaptation to a given context (cf. Section 6).

5. BASICS OF ARCHITECTURAL DESCRIPTION

For the description of a system's structure we use the concepts known from software architecture (Shaw 1996, Pepper 2002). Doing this we apply a well-known method to an uncommon area. By this transfer we reuse the competence that was gained on the field of software engineering for the structural description of sociotechnical systems. By adopting these concepts we establish an ontology for the conceptual modeling of complex systems.

We conceive the structure of a system as a configuration of *agents* which are interacting using *connectors.* By providing different types of agents and connectors with different signatures (or interfaces) we are able to define semantically meaningful types of configurations (usually called architectural styles) by composition. Both, agents and connectors publish *interfaces* (specified by algebraic signatures) which may be connected by *links* (cf Figure 1).

First we provide a basic vocabulary of agent and connector types which we found suitable for the description of a given domain (cf. Figure 3 for an example). We notate agents as soft-boxes and connectors as ovals. Interfaces are represented by little squares.

As an example we choose the setting of a medical operation theatre. While this special system is characterized by a great variability we have to content ourselves with describing exemplary scenarios taken from anesthesia.

We describe interfaces using a tabular specification: each variable and action name is listed with its name and its type (cf. Figure 2). Thus, for example the anesthesist can use his voice to give some commands *(give, take)* or use his visual sense to observe the color of the patient's face. In the given situation he is also able to perform some actions (intubating the patient or configure the monitoring device).

Every interface is described by a set of input and output variables and a set of action names which specify his capabilities to interact with his environment by physical channels. This visual specification technique is fully compatible with the more formal methods of algebraic specification.

We use features (also called port variables) to describe the physical qualities and services of agents. The left hand of an actor or the displays of a technical device are examples for this kind of variable. The domain of these variables frequently is defined by enumeration types which may have fuzzy semantics (Klir, 1995).

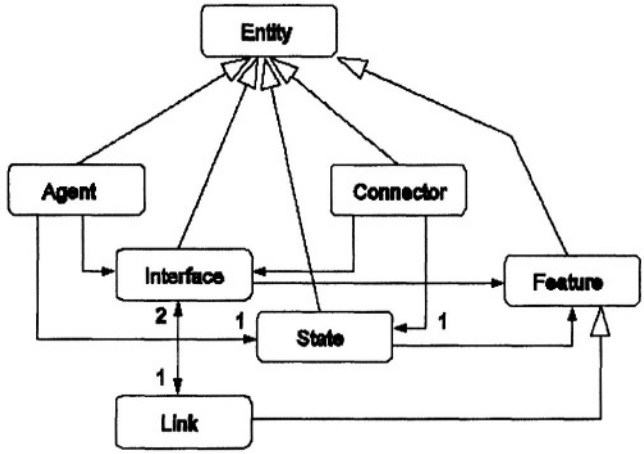

Figure 1. Metamodel for Systems Analysis

As we show in Figure 1 we conceive an ontology as a collection of entities. Each of the syntactic elements presented in this section is an extension of *entity*. In addition we make some statements about the way these concepts are combined. Agents as well as connectors may possess a set of interfaces, which in turn contain a set of features. These aggregation relations possess the cardinality 0...n which we conceive as default and which is ignored in the figure. Agents and Connectors may contain a state which in turn contains a set of features. Links as special cases of features connect exactly two interfaces.

We conceive this framework as a starting point for the definition of domain-specific ontologies which consists of typed specializations of these abstract concepts (i.e. HumanActor, Anesthesist, or Communication).

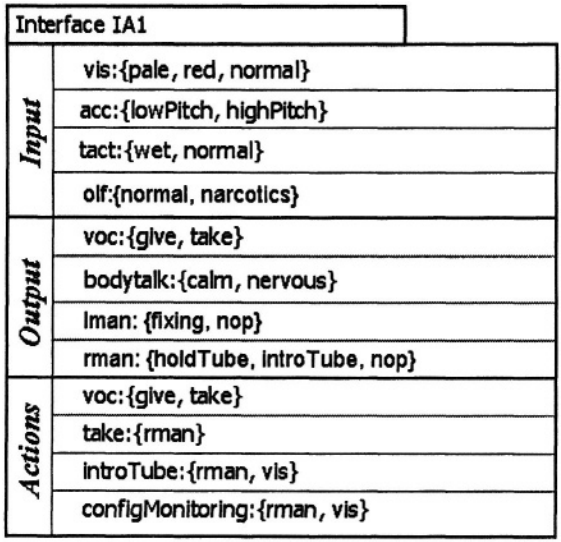

Figure 2. Tabular Specification of an Interface

6. MENTAL MODELS: A COGNITIVE APPROACH

We provide a *cognitive perspective* which takes into account an agent's subjective motivations and their influence on the global system's behavior as major contextual parameters. We claim that this allows for a better understanding of the contextual determination of systemic behavior. Especially the human factors and organizational relationships in sociotechnical processes can be modeled by these concepts. For this sake we adopt the concepts related *mental models* as known from the BDI-style frequently used in agent oriented modeling (Singh, 1994).

An agent's mental model consists of:

- *The agent's intentions:* usually a decision aims at a certain goal. That means that the agent tries to achieve a certain system state by selecting between alternative behavioral options.
- *The agent's beliefs (also: expectations):* a decision is highly influenced by the agent's belief concerning the system's actual state and its further behavior.
- *The agent's desires:* usually the agent has a subjective preference for a certain behavioral option which may or may not interfere with the real situation.

We represent this internal information using fuzzy sets. This gives us the possibility to represent the *subjective relevance* of a proposition by the membership relation μ (Sperber, Wilson, 1986).

We notate fuzzy sets by using bold font. Following a widespread convention we occasionally use the name of the set as a shortcut for its membership function. The function of the three fuzzy sets **B, D** and **I** consists in the mental representation of the relevant contextual features.

- Fuzzy set **I** represents the agent's intentions.
- Fuzzy set **B** represents the agent's expectations concerning the system's actual state.
- Fuzzy set **D** represents the agent's desires (preferences) concerning the behavioral options. These preferences are usually context independent.

In Figure 3 we use these concepts for the definition of an actors's *adaptation.*

We claim that in order to be well adapted the actors's internal representations have to be very close to the relevant features of the given context. Especially the actor has to have an intensive internal representation of the global system goals (first condition) and has to be well informed regarding the relevant features of a given context (second condition).

Alternatively she may have a strong inclination for the right behavioral option (third condition). In every case the relevant propositions have to be members of the nurse's mental model to a high degree. We say that the nurse is well adapted to her context if the first and second conditions or the third condition are given.

As in Figure 3 we use fuzzy rules to approximate the relation between contextual features and an actor's mental representations. In general, we claim that an actor's contextual adaptation is good if he has in intensive representation of the relevant features. We use AND/OR-tables to reason about different configurations and their consequences for the quality of adaptation. There we use the letters T and F to mark the conditions which are true resp. false. Conditions that are not relevant in a given configuration we mark with a star. To qualify key features of a configuration we use the symbolic constants *high* (*h*), *medium* (*m*) and *low* (*l*).

For organizational purposes there are some interesting features of contexts. The cardinality of set *A* describes the free room of an agent's decision. If it is small this has inhibiting effects on his motivation. If it is too great the agent may be overstrained. Moreover, another important feature of a given context consists in the possibility of differentiation of behavioral alternatives.

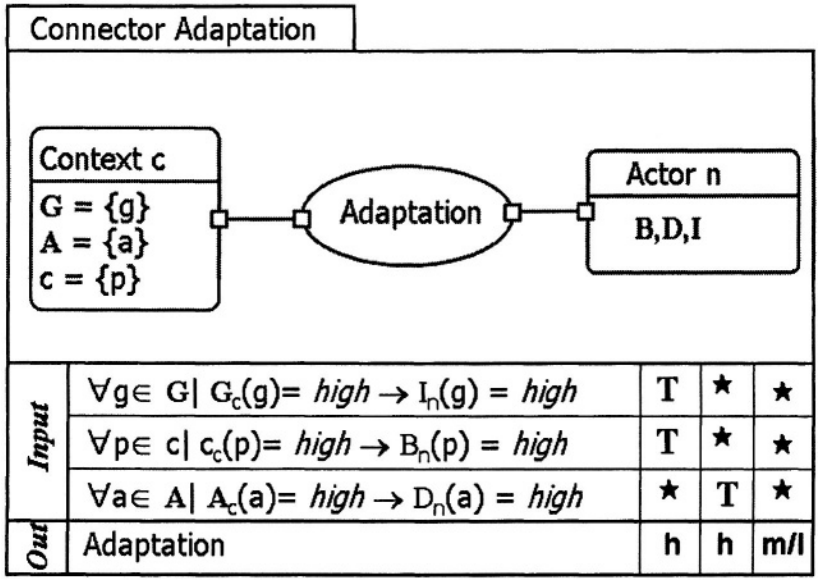

Figure 3. Adaptation

7. CONTEXTUAL DEPENDENCIES

For the processing of systemic tasks agents and devices constitute configurations by establishing interaction networks using connectors. In order to model the adaptive capabilities of sociotechnical systems as well as the contextual dependencies of configurations we distinguish between *abstract models* and *situational models* (adopting analogous concepts from Soley, 2000). We provide concepts for the definition of context-specific transformation rules which transfer abstract specification into situational descriptions according to contextual conditions.

Hence, in Figure 4 we show a transformation rule which defines the adaptive capabilities of a given configuration with respect to certain adversal context conditions. In our example, at the beginning of a given task a necessary precondition is not given. The laryngoscope (a special instrument which is used during intubation) is not owned by the anesthesist.

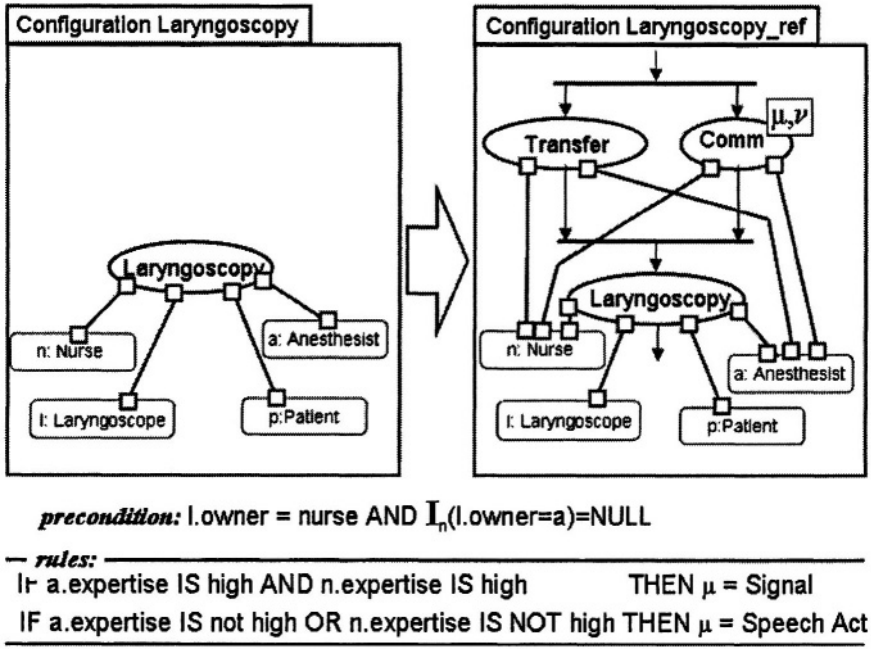

Figure 4. Transformation Rule for Adaptive Behavior

In order to compensate this adversal condition we have to change the task structure by the application of our transformation rule. We generate two coordinative task which establish the desired state of affairs. The way in which this is done highly depends on the capabilities and internal representations of the agents involved. Thus we have to generate a communicative task because in our example the nurse isn't aware of the necessity to give the laryngoscope to the anesthesist (as stated in the precondition).

Intuitively, in Figure 4 we describe the case that the nurse possesses the device in the wrong moment and is not aware of necessity to give it to the anesthesist. We claim that this situation is characterized by strong adverse conditions which the system has to compensate by context-adaptive behavior. We describe one possible behavioral alternative in our rule.

Using this kind of rule-based specification allows us to catch the context-sensitivity of task processing. Which type of procedure is used by the actors in a specific situation is highly dependent on the parameters of the context. One example for such parameters is the *expertise* of the actors: highly trained experts tend to coordinate their activities using signals, while novices normally use convention-based speech acts. This rule is very similar to configuration rules in feature modeling (Czarnecki, Eisenecker 2000).

8. SOCIOTECHNICAL CONNECTORS: COMMUNICATION

We conceive connectors as specifications about the behavior of agents. We use temporal logics to define the required behavioral properties. In our example (Figure 4) we use enabling relations inspired by event structures (Winskel, 1988) which allows us to specify partial orders over events.

An important relation between agents is *Communication.* For the definition of this relationship we have to use our operators from epistemic logic to specify formulas about the mental models of agents.

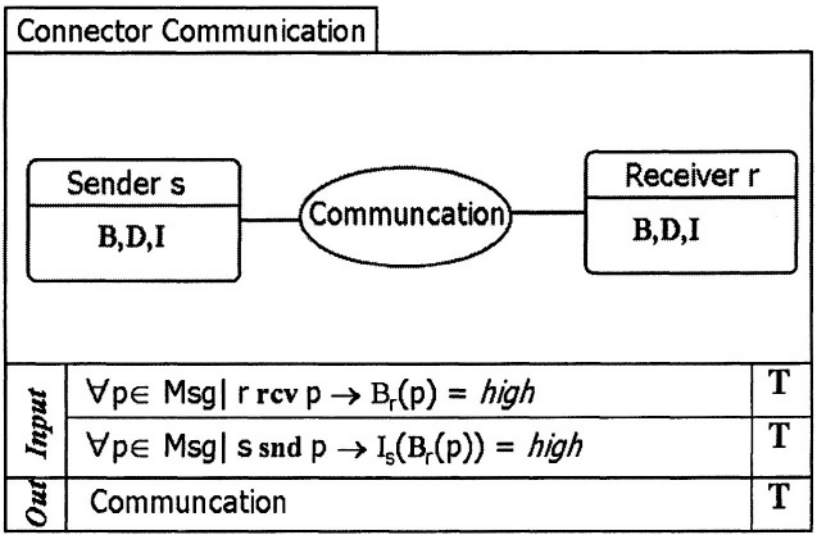

Figure 5. Connector *Communication*

Our first input condition in Figure 5 specifies what is known as the *perlocution condition* from speech act theory: a result of a communication process is that the message's recipient believes in the content of the message because he received this message. Using our terminology we claim that the recipient of message beliefs its content when the proposition is member in his set of beliefs **B** to a high degree.

The second condition is called *illocution condition:* for a communication act it is a necessary condition that it belongs to the actor's goals to induce the receiver's conviction that the proposition *p* holds.

This formalization of communication processes is an exemplary result of interdisciplinary cooperation. We introduced semiotic concepts in our modeling framework which are presented in (Cebulla, 1995).

9. UNCERTAIN BEHAVIOR AND HUMAN ERROR

Unlike deterministic behavior of technical devices human decision making is a source of behavioral uncertainty. The outcome of an actor's decision is highly unpredictable and depends on numerous parameters which are not directly observable (as for example mental models).

Since we have a special interest in the safety analysis of sociotechnical systems we use our modeling framework for the analysis of human error. We reason about errors using fuzzy fault trees. Using fault trees we can reason about different types of human errors using propositions about mental models. As shown in Figure 9 we distinguish between three cases of human error which are grouped in two sections following (Reason, 1990):

- Execution-based errors *(slips)* occur when the subjective preferences of an actor are not adapted to the context (goals and beliefs may be fitting).
- Knowledge-based errors *(mistakes)* may occur when the beliefs of an actor are not adapted to the given context. For example a nurse may have the right goals but may not act right because she doesn't see the need to act. Another error-case occurs when an actor's goals do not fit well into the given context.

We claim that the integration of cognitive parameters like mental models into our modeling formalism provides a foundation for the reformulation of interdisciplinary results from semiotics, sociology or psychology. Note that we want to provide a conceptual framework which allows us to consider these aspects into the integrated modeling of complex systems. As a specific benefit of this integration we conceive the modeling of complex interactions as for example the situational interplay of human factors and physical properties of technical devices. A modeling framework which allows for this integration is a necessary precondition for the systematic analysis of this systemic couplings and side-effects.

Thus, we suggest that the use of integrated fuzzy fault trees is an essential method for an interdisciplinary scenario based system analysis. We claim that the impact of misplaced cognitive representations on the overall systemic behavior can be analyzed using these concepts.

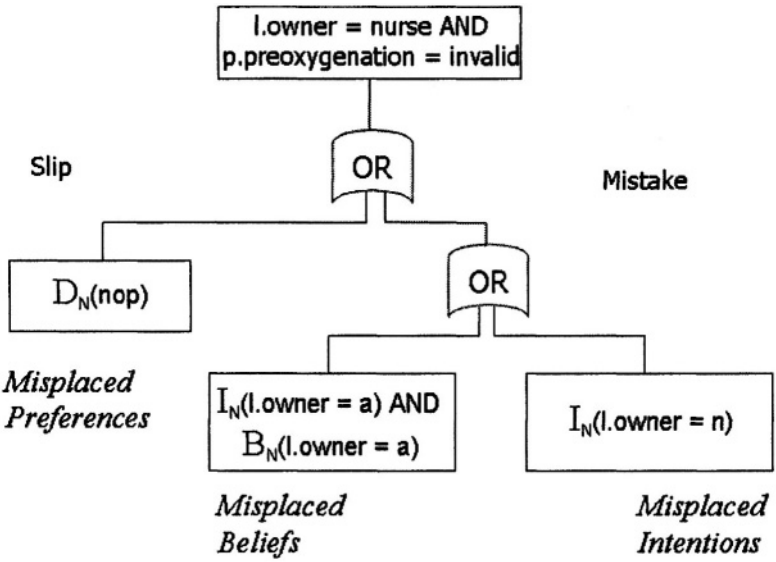

Figure 6. Fuzzy Fault Tree for Human Error

10. CONCLUSION

In this paper we provided and demonstrated elements of a visual notation for the integrated analysis of complex sociotechnical systems as well as for advanced context aware applications. We started with structural description, dynamic architectures, and rules for reasoning about adaptive system properties. For this sake we extended common architectural concepts by fuzzy methods and a transformational approach. We then took a *cognitive perspective* by introducing the concepts of mental models. At this point we used results from psychology, sociology and semiotics.

On this platform we provided modeling concepts for specific human and organizational relationships. We conceived for example communication as coordination mechanism which makes it possible for agents to balance their intentions and beliefs concerning the further processing of the system's task.

Using this model we can reason about the system's safety-critical features. As we demonstrated each mechanism acts as an context adaptive mechanism in the face of adverse environmental condititons concerning the system's safety. We also call these mechanisms *safety barriers.* Finally, we sketched our way to deal with uncertain behavior and human error.

ACKNOWLEDGEMENTS

This work was funded by Technical University of Berlin in the context of an interdisciplinary research focus *Cooperation and Safety for Complex Sociotechnical Systems (KOSIS).* I am very grateful for the cooperation with my colleagues from Ergonomics, Semiotics, Sociology and Psychology. I am also indebted to the anonymous reviewers for their constructive criticism.

REFERENCES

Baader, F., Calvanese, D., McGuiness, D., Nardi, D., Patel-Schneider, P. *The Description Logic Handbook. Theory, Implementation and Applications.* Cambridge Univ. Pr. 2003.

Cebulla, M. "Pragmatik und KI: Posners semiotische Fundierung der Sprechakttheorie." *S-European Journal for Semiotic Studies.* Vol. 7, No. 1,2,1995.

Cebulla, M. "Reasoning about Variability and Structural Adaptation in Sociotechnical Systems using Dynamic Architectures." *Proc. of SEKE'03,* Redwood City, Ca. 2003.

Czarnecki, K., Eisenecker, U. *Generative Programming. Methods, Tools, and Applications.* Addison Wesley: Reading, Mass., 2000.

Klir, G., Yuan, B., *Fuzzy Sets and Fuzzy Logic. Theory and Applications.* Prentice Hall: Upper Saddle River, N.J. 1995.

Leveson, N., *Safeware. System safety and computers.* Addison Wesley: Reading, Mass. 1995.

Leveson, N., *A new Accident Model for Engineering Safer Systems.* In preparation 2002.

Minzberg, H. *Structures in fives: designing effective organizations.* Prentice Hall: Englewood Cliffs, 1983

Partsch, H., *Requirements Engineering systematisch.* Springer, Berlin u.a., 1998.

Pepper, P., Wirsing, M, "A method for the development of correct software", In: Manfred Broy, Stefan Jähnichen (eds.), *KORSO: Methods, Languages, and Tools for the Construction of Correct Software.* Springer: Berlin u.a., 1995

Pepper, P., Cebulla, M., Didrich, K., Grieskamp, W. "From Program Languages to Software Languages", *The Journal of Systems and Software,* 60,2002.

Pepper, P, Frank, C., Holfelder, W., Jiang, D., Matylis, G. *Dynamic Architectures for a „sometimes somewhere" concept.* Technical Report, TU Berlin, 2003.

Perrow, C. *Normal Accidents. Living with High Risk Technologies.* Basic Books: New York, 1984.

Reason, J. *Human Error.* Cambridge Univ. Press, 1990.

Sage, A.P., Rouse, W.B. (eds.) *Handbook of Systems Engineering and Management.* Wiley: New York. 1999.

Shaw, M, Garlan, D. *Software Architecture. Perspectives on an emerging discipline.* Prentice Hall: Upper Saddle River, N.J., 1996.

Singh, M.P., Multiagent Systems. A Theoretical Framework for Intentions, Know-how, and Communications. Springer: Berlin, 1994.

Soley, R. "Model Driven Architecture, White Paper." ftp://ftp.omg.org/pub/docs/omg/00-11-05.pdf (23.3.2004).

Sperber, D., Wilson, D. *Relevance. Communication and Cognition.* Harvard Univ. Pr.: Cambridge, Mass, Blackwell: Oxford, 1986.

Winskel, G. “An Introduction to Event Structures”, in: *Linear Time, Branching Time, and Partial Order in Logics and Models for Concurrenc.,* Springer: Berlin, 1988.

STUDYING OPERATOR BEHAVIOUR DURING A SIMPLE BUT SAFETY CRITICAL TASK

Hans H. K. Andersen and Gunnar Hauland
Risø National Laboratory, Denmark and DNV, Norway

Abstract: A loss of sufficient Situation Awareness may lead to human errors, possibly resulting in accidents. Situation Awareness is often conceptually described in terms of operators' correct perception and understanding of a situation. It is though becoming increasingly clear, however, that team SA is an important concept as well. An initial idea for a continuous measure of Team Situation Awareness was tried out in a small technical pilot study conducted in the nuclear reactor control room at Risø. In our present study we seek to develop integrative methods combining eye-movement tracking data with other behavioral data. The study described in this paper is a pilot study, which seeks to establish the feasibility of applying these methods of measurement and analysis, not their validity.

Key words: Team Situation Awareness, Collaborative Work, Eye-Movement Tracking

1. INTRODUCTION

During millennia, evolution has granted humans an extremely effective perceptual-motor system serving control of the dynamic interaction with the environment and its inhabitants. In natural environments, all observable information is available all of the time. Thus, the actor can vary the level of abstraction at which the environment will be perceived at will. Listening to the sound of the wind in the trees helps us in understanding the weather even though we are focusing entirely on other activities. Likewise in the periphery of our attention, we are aware of activities of colleagues in our office, for example, by listening to footsteps outside our office door, listening in to

small talk activities, discussions, etc. Also we signal our own availability in opening or closing our office doors.

Behavioral studies of decision-making during the natural flow of work have identified some basic characteristics of team situation awareness. Expert know-how and rules-of-thumb depend on adaptation to a work environment governed by an empirical correlation of convenient cues with successful acts. During normal familiar work, actors are immersed in the context for long periods; they know the flow of activity and the useful action alternatives by heart. They therefore do not base their actions on rational situation analysis, they do not have to base decisions on integration of a defining set of situation attributes (data) before acting in a familiar situation. Instead, they will seek no more information than is necessary for discrimination among the perceived alternatives for action in the particular situation. Consequently, they ask very biased questions to the environment and, consequently, when work situations change reliance on the usual convenient cues will lead to failure

The notion of team situation awareness in cooperative work settings has been emphasized by Rochlin et al. (1987) in their study of normal work practice onboard an aircraft carrier. Their findings point to the importance of the role of fringe-consciousness and an updated, tacit context awareness for a high cooperative reliability. Also within the CSCW community it has been frequently recognized that people's awareness of current work situations is an important characteristic of cooperative work. As such it has been shown that in many cooperative work settings it is possible to maintain situation awareness by the rich interaction, communication, and perceptive modalities of everyday social life. The study of Line Control Rooms on the London Underground (Heath and Luff, 1992) shows how actors maintain fluent reciprocal awareness regarding other actors' activities. In doing so the actors monitor each other's activities by overhearing other actors' radio or telephone conversations. Also they attract attention to activities, which are less visible to others, for example, when working with timetables and logs, by reading or thinking aloud or even by humming, singing, feigning momentary illness etc. In distributed cooperative work settings it is very difficult to maintain the same kind of situation awareness due to separation in perhaps both time and place.

2. TEAM SITUATION AWARENESS AND TEAMWORK

A loss of situation awareness may lead to human errors, possibly resulting in accidents. Situation awareness is often conceptually described in

terms of operators' correct perception and understanding of a situation. This conception of individual situation awareness has been summarized by saying that an operator should be able to (1) recognize the relevant elements in a situation, (2) understand how these elements are interacting and, on the basis of this understanding, (3) predict the system status into the immediate future (Endsley, 1993; 1995). A diminished degree of awareness at any of the three levels will typically compromise performance.

It is typical of complex technical real time systems that they require, for the sake of safety and efficiency, more than one operator. When two or more operators are controlling a process, the collaborating collection of operators can be referred to as a team, i.e. the concept of team situation awareness should include inter-personal aspects of awareness. So, relative to each individual operator, his or her current model of the task domain will also include how other team members perceive and understand the situation and how they understand his current knowledge. Thus, in a sense team situation awareness is a component of individual situation awareness: operators must to some extent be aware of each other's tasks and of each other's awareness of those tasks. In short, therefore, team situation awareness involves team members' mutual knowledge about their task domain (See Andersen et al., 2001 for details on this point).

SA may be assessed by either subjective measures (involving operators' or expert observers' qualitative evaluation of performance and behaviors) or objective ones directed at subjects' responses and task directed behaviors. In this paper and in the study we refer to - we have concentrated efforts on objective measures- For several reasons we prefer to apply measures that do not involve an interruption of operators' task. That is, we have focused on assessment methods that involve measures that are made continuously across the evolvement of the task scenario. In a study of pilots visual behavior and its relations to team situation awareness we found that the pilots showed a high similarity in their scanning behavior across normal and abnormal situations and a high homogeneity within these situations.

Based on interviews with the operators we have sought to define such set of ideal visual behaviors that could allow us to compare this norm with the observed visual behavior. Thus, we hypothesize that the degree of correspondence (i.e., in terms of percentage) between a pre-defined ideal visual behavior and the observed visual behavior may be able to constitute a measure of situation awareness. As will be explained below, the team aspect will then be added to this measure by including behaviors, which involve the perception of, and interaction with fellow team members.

While it could be possible to define an ideal visual scanning behavior for the completion of a certain task, this need not be the case when it comes verbal and non-verbal coordination activities. As also mentioned above such

activities may not in all cases follow ideal paths, since there might be a need deviate from, for example, procedures just to make things work in a given situation.

While operators do not always access system and task information visually, the situation awareness measure is based on operators' accessing visually presented information (Hauland, 2003). A range of measurable psycho-physiological patterns and variables (including those that originate from the use of eye-movement tracking) are underdetermined when it comes to interpreting what they might indicate about cognitive processes. That is to say, data thus obtained must be disambiguated, categorized and interpreted in the light of additional data about what cognitive processes and actions are likely in the domain under consideration.

The data analyses of such information gathering include units of analysis at this level of meaning, i.e., at the semantic level. At the same time there is a need for having data that potentially represent meaning in order to guide analyses of physiological data. Visual behavior does not by itself reveal or signify what the subject is thinking and intending rather, that semantic analysis is required to achieve this. Operators communicate through the process interface, i.e. they may infer what the other operators are aware of by watching key parameters in the process interface. Such Areas of Interest (AOI) may be substitutes for questions concerning each process parameter. We therefore pursue a method of defining AOI's relevant to team situation awareness and, subsequently, of measuring the line of gaze towards these AOI's by using eye-movement tracking. For the verbal behavior of operators we have selected an aspect that in previous studies have proven to correlate well with performance and task outcomes. Thus, communication may reveal how well the operators understand the developing situation. In studies conducted by Risø and the Danish Maritime Institute (Andersen et al., 1996), a significant correlation was found between crews' communication related to future system states and performance in simulated (ship maneuvering) tasks.

What the above means, in terms of trying to measure operators team situation awareness during system control activities is that it is necessary to have methods capable of combining the analysis of (i) visual and other behavioral data; (ii) subjective (where raw data observers' or subjects' own interpretations), and objective data (where raw data are recordings of directly observable behavior), and (iii) data representing both voluntary/intentional behavior (actions) and psycho-physiological data (automatic, micro-behavioral data that do not directly represent intentions).

In our present study we seek to develop such integrative methods combining eye-movement tracking data with elements of, first the Cognitive Systems Engineering framework developed at Risø National Laboratory (Rasmussen et al, 1994) for analyzing operators' cognitive activities and

second the sociological frameworks for the analysis of everyday social non-verbal communication modalities (Andersen, 1997). While Cognitive Systems Engineering is relatively well known within the area of Human Error Safety and Systems Development the sociological approach to the study of team situation awareness activities is less recognized and may require a little attention here.

In engaging in teamwork actors generally become mutual dependent. They cannot fulfill the tasks on their own, so they have to rely on the contribution of other actors applying their different capacities, competencies, strategies and perspectives. Given their interdependence they need, in some way, to articulate their individual activities in joining their efforts. The term "articulate" in this context comes from the work of Strauss (1985), and Gerson and Star (1986). In this sense articulation means to allocate, co-ordinate, schedule, interrelate, integrate, etc., individual activities according to the dimensions of who, where, when, how, what, etc. The articulation work can be considered a type of second order activities or overhead cost in terms of the use of resources or time. The actors engage in these overhead activities because they would not on an individual basis be able to accomplish a certain task.

Teamwork is constituted by the fact that multiple actors are interdependent in their work. In other words, they are working in the same "field of work", that is, they are transforming and controlling a conglomerate of mutually interacting objects and processes. Thus, all teamwork involves and, indeed, is based upon interaction through changing the state of a common field of work. What one actor A is doing is of import to B and C in doing their work. The other actors C and B may to some extent be able to infer what A is doing from the changing state of the field of work. However, while collaborating via changing the state of the field of work is basic to all teamwork, it is rarely adequate. In fact, articulation of teamwork involves and, indeed, requires a vast variety of social modes of interaction that are combined and meshed dynamically and seamlessly in accordance with the specific requirements of the unfolding work situation and the means of communication available. As we see it there are four main interaction categories or modes of interaction:

Maintaining reciprocal awareness: The team could be involved in synchronous activities, by monitoring colleges' location in a room, and to monitor their activities. Moreover, they could be engaged in explicitly making their own activities publicly visible to teammates by thinking aloud, humming, etc.

Directing attention: Actors attract the attention of team-mates to focus on certain features or emerging problems in the field of work by, for

example, to position certain items in certain ways, by pointing or nodding at particular items.

Assigning tasks: Actors could for example allocate a task by nodding at a work object or by stating a verbal request.

Handing over responsibility of processes in the field of work, for example, by passing on the work object in question, or the interface of a control mechanism.

These social modes of interaction are combined and meshed dynamically and fluently to meet the requirements of a specific situation. The different modes of interaction cannot be ordered in any simple kind of way but is possible to point at a limited number of prominent dimensions of the modes of interaction. Some examples:

Unobtrusive versus obtrusive, that is, some modes of interaction can be disruptive in nature in relation to a colleagues' line of work, while others are very conspicuous and therefore permit colleagues to carry on working.

Embedded versus symbolic, that is, to embed cues in highlighting certain items belonging to the field of work by for example marking them versus using a symbolic representation of the cues, which through its abstract function offers a higher degree of freedom regarding the manipulation of the cues.

Ephemeral versus persistent, that is, shared situational awareness only appears during the course of work and then disappears without leaving any trail to track. It is for example not immediately possible to trace activities like monitoring co-workers activities or to make one's own activities publicly visible.

The study described in this paper is a pilot study, which seeks to establish the feasibility of applying these methods of measurement and analysis, not their validity. The initial ideas for a continuous measure of team situation awareness were tried out in a small technical pilot study conducted in the nuclear reactor control room at Risø (a 12 MW research reactor). The study was carried out during normal operation of the reactor control room focusing on operators' co-ordination of tasks in achieving the desired level of safety and efficiency for running the reactor.

3. THE PILOT STUDY

The data derived from the pilot study described here have been subjected to the integrative analysis described above. Since the data recordings associated with this type of eye-movement tracking are rather complex in themselves we need to describe them briefly.

Risø has invested in new laboratory equipment enabling faster and more flexible analyses of many types of data, visual data in particular. The use of these data recording and analyses systems is highly skill based and the set-up of components in the field is not straightforward. It was therefore decided to explore the usability of major components of the new laboratory equipment in the field with the specific consideration in mind that these types of data may be used for the analysis of team situation awareness.

The two main objectives of this technical pilot study were to (1) gain hands-on experience with the eye tracking and the analysis systems under conditions where this equipment had to be operated in the field during real (non-interruptible) scenarios and (2) to acquire experience about the implementation of the suggested measures to be combined into an assessment of team situation awareness.

Figure 1. A nuclear control room operator wearing the eye-tracking helmet.

The selected target task in the nuclear research reactor (configuration of neutron flux by inserting and removing of neutron absorbing rods) is typically performed at intervals of 48 hours during normal operations. The exchange of rods aims at optimizing the configuration of the core. The task, which requires one operator at the top of the reactor to adjust the rods, and one operator in the control room to monitor instruments, lasts from 2-5 minutes (excluding preparations). The two teammates have to co-ordinate their tasks closely in order to adjust the reactor in a safe way. The removal of the rod has been done very smoothly and not too fast. Failing to do so will cause the reactor to shutdown automatically. For the pilot study data was collected through:

a) Combined head- and eye-movement tracking from the operator in the control room.
b) Video recordings of the operator at the top of the reactor.
c) Video recordings of the 3D model of the instrumentation in the control room (mixed with eye point gaze data).

d) Audio recordings from both operators.
e) Questionnaire.
f) Debriefing with the group of operators.

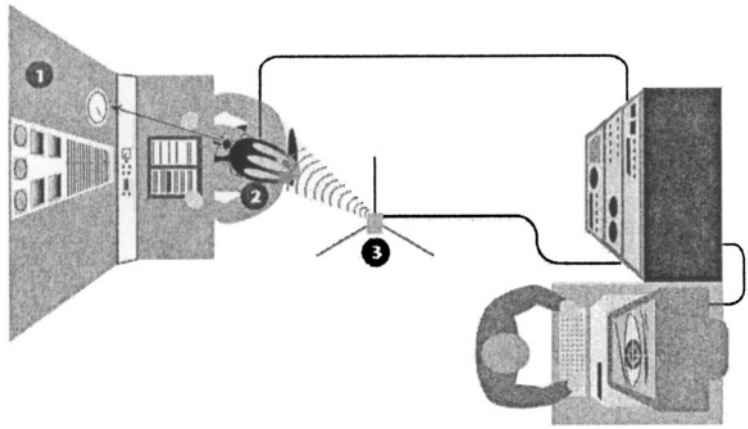

Figure 2. The technical set-up of equipment.

Figure 2 illustrates the overall set-up in the reactor control room with one of the operators and one of the researchers. The second operator is located on the top of the reactor, but is visible to his colleague through the monitor in the control room.

1. The reactor instruments, monitored only by the operator in the control room, display how the neutron flux in the reactor core is changing as a consequence of the team-mate's manual removal of neutron absorbing rods. The operators communicate through the intercom.
2. The operator wears the eye-tracking equipment. An effector placed on the top of the helmet gives the position of the head relative to the environment using a magnetic field. Together these enable continuous measurement of what instruments the operator is looking at. Both operators are wearing wireless microphones.
3. The magnetic tracking system combined with a laser pointer tool is used for building a 3D computer-model of the control room. The magnetic transmitter is the reference point for the effector mounted on the helmet, enabling integrated eye- and head tracking displayed in the model. The advantage is that eye-movement data can be analyzed automatically.

4. RESULTS

Pre-study interviews indicated that the operators used most of their time during the task to monitor the Fine Control Rod meter (FCR), the effect meter, doubling time meter and the monitor (video of top of reactor). The fixation frequencies for the operators showed another picture. The operators looked outside these instruments more than 20 times on average during the operation. They mostly used the FCR for monitoring the task (15 fixations an average). They had 3 fixations on the effect meter, 6 fixations on the doubling time meter; and only one fixation on the monitor (video from top of reactor). The total duration of eye-point of gaze during the operation the operators looked outside the mentioned instruments for more than 40 % of the time on an average, while gazing at the FCR in 42%, the effect meter for 3%, the double time meter 12,5%, and the monitor 2,5% during the task (see *Figure 3*).

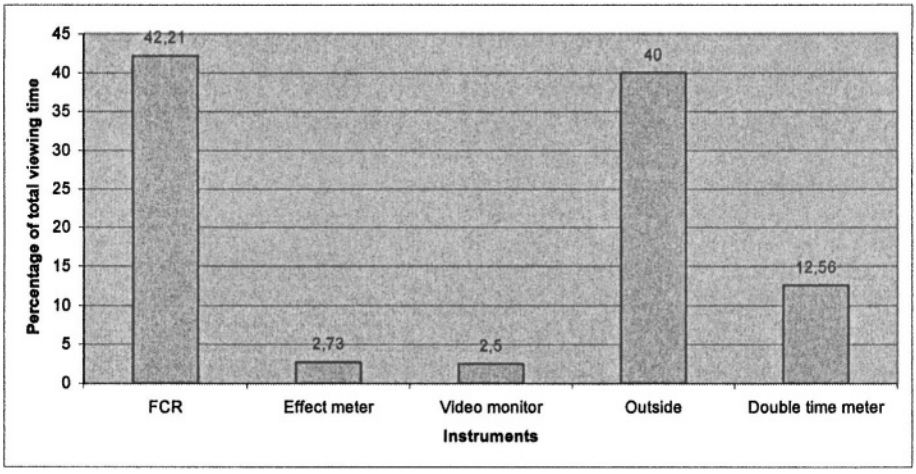

Figure 3. Viewing time on areas of interest.

This means that although the operators in the control room had the possibility to watch the actual removal of the rod they prefer not surprisingly to monitor the task through the different meters. The data shows that they mostly use the monitor to see that the operator is at the top of the reactor, so they can tell him (through an intercom) to start the task.

Figure 4 shows a model of the task. The two involved operators initiate the task in the control room in coordinating who is to do what. The operator on the top (OP) prepares for the removal of the rod, while the operator in the control room (OC) goes through the logbook to check the size of the rod, and date and time for removing the rod. While doing this he looks at the monitor a couple of times to see how OP progresses. When OP has finished

the preparation for removal he positions his body in a certain way to in a non-verbal way to communicate through the video camera to OC that he is ready to pull the rod out. The reason for this is that, he is not able to use the intercom from where he is standing (to speak through the intercom the OP needs to press a button because it is a simplex based devise). The OC sees that OP is ready to pull and issues a "start" command. While OP pulls the rod, OC monitors the instruments. When OP is finished he walks to the intercom and issues a "finish" command. Then OC update the logbook, while OC cleans up after the removal of the rod. The team situation awareness aspects of this task is most clear, we think, in that the OP knows that he is being monitored via the video channel. The OP also knows that the OC knows that when he (the OP) positions his body to communicate readiness for pulling the rod, so this is a signal to issue the start command. In terms of the social modes of interaction (discussed in Section 2) conveying the team situation awareness of the two operators, the operators seek to maintain reciprocal awareness. That is, in their synchronous activities, the operator in the control room monitors his college's location on the top of the reactor, while the operator on the top explicitly makes his own activities publicly visible to his teammate and communicates readiness to carry on his task by placing his body in certain way. If we look at the dimensions of the social modes of interaction the operators' team situation is maintained in unobtrusive way and therefore permit the teammates to carry on working that in the given situation can be considered as optimal strategy. The coordination activities are also ephemeral in the sense that team situational awareness only appears during the course of work, which could be considered as a less optimal strategy, for example, if less experienced operators where to coordinate the task it could be considered to use a checklist to control the coordination activities.

This is of course a very simple task and there might be other ways of interpreting the task. Also we are of course aware that the technology available in the 30-year-old reactor, in certain ways, provokes certain awareness activities. This could also be the case in more advanced systems, but then on a very different level.

In the questionnaires, operators were asked to describe the success of the rod removal. They all agreed that the task could not have been performed better. All tasks were performed according to regulations and there were no shutdowns.

Operators had not worked together on this particular task before (due to summer vacation), but they had all worked together for a long time on other tasks at the reactor. This may have played a role in the communication, since no misunderstandings were produced.

Based on the visual and verbal behavior/questionnaires/recorded instruments, operators had no problems performing the task. The removal of neutron absorbing rods is a relatively simple task with respect to perception and understanding, but it is a safety critical task that has to be carried out manually. The fact that the task has to fine-tuned with activities in the control room demands the operators to maintain a sufficient level of team situation awareness in order not do any mistakes.

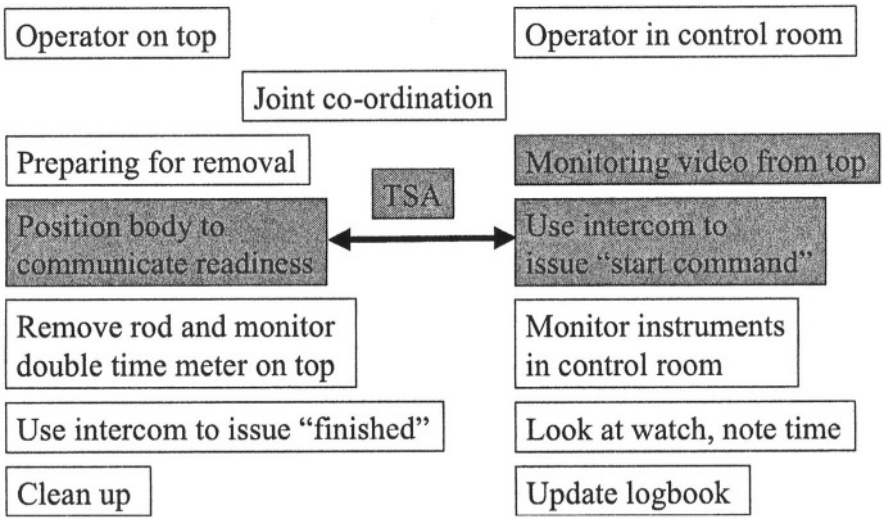

Figure 4. A model of team situation awareness during pulling of the control rod. Gray-scaled areas indicate team situation awareness activities.

As a point of departure for de-briefing session we showed a video where we had included clip that seemed a bit more difficult than the rest. The debriefing-video consisted of a mix between our eye-movement tracking scene video recordings, the recordings of the 3D model and various location recordings. We focused on discussing situation awareness, but also discussed other issues. The main result from the de-briefing is summarized below:

There were sources of information not detectable through analyses of visual information gathering and verbal communication, like e.g. listening to the elevator driving the rods.

Operators did not look at the instruments they claimed to be looking at (both before the study and when watching the video).

All operators agreed on what information was important to solve the task. However, the discussion revealed that operators, even the most experienced ones, disagreed with respect to the priority of information acquisition, i.e. exactly how to access the relevant information (from what instrument, for how long etc.).

5. CONCLUSION

The de-briefing session showed us that it seems to be difficult to establish a standard means by which the operators acquire information during the rod removal task and no written procedure exists i.e. defining the norm for team situation awareness may be difficult. Although individual differences in the approach to information acquisition can be observed, these differences are not necessarily more or less correct. One feasible way to measure team situation awareness may be to look for the lowest common denominator for the information items, that the operators need to solve a task, and to avoid defining details concerning qualities of the operators information gathering activities like sequence and duration of the activities, except when there can be established a clear operational definition of these activities, e.g. where the operators activities under certain circumstances are specified in checklists. (See also Hauland 1996 on this issue).

Although operators did use instruments differently, they all relied on the same information. It was not possible from these example trials to observe variation neither in task performance nor intermediate activities like visual information gathering (type of information) and verbal communication. It seems like a (complex) measure of team situation awareness like the one proposed, is less useful for very simple tasks - tasks not likely to produce much variation.

Situation awareness is thought to be more than exceptional attention. It includes the integration of many elements in the situation, including the projection of how the situation will develop. One could ask if judging ordinal single variables like on/off, under/above calls for the type of overview we want to measure with team situation awareness. If it does, one could propose another explanation for the lack of variation: It is assumed that mental workload is tightly coupled with the concept of team situation awareness.

The relationship between workload and situation awareness is often claimed to constitute an inverse u-shaped curve: A low level of mental workload is associated with a low level of situation awareness, a medium level of mental workload is associated with a high level of situation awareness, and a very high level of mental workload is associated with a breakdown of situation awareness. Thus, if a very low level of task complexity is perceived, one would expect this to be reflected in a low level of mental workload, and consequently low situation awareness.

This explanation is probably not relevant here however, since all tasks were performed in accordance with regulations. It is more relevant, we think, to ask if this task really required team situation awareness in the way we have defined it. The difficulty in the current task – if any – is sensory-

motoric (pull slowly) and co-ordination (stop pulling on command from the control room operator). It may be relevant also to ask about situation awareness in very simple task, but the complexity of the measure (resolution and number of units of analyses) must be in proportion to the tasks to be performed. Thus, the team situation awareness measure we wish to develop aims at measuring awareness of complex tasks, where complexity may be reflected in, e.g., number of situation elements and the type of relationship between these elements.

REFERENCES

Andersen, H. H K. (1997). Cooperative Documentation Production in Engineering Design: The 'Mechanisms of Interaction' Perspective, Topics in Cognitive Science and HCI, Centre for Cognitive Informatics. Roskilde University, (9).

Andersen, H.B.; Pedersen, C.R.; Andersen,H.H.K. (2001) Using eye tracking data to indicate team situation awareness. In: Usability evaluation and interface design: Cognitive engineering, intelligent agents and virtual reality. Proc. of HCI International 2001. Vol. 1. 9. Int. conf. on human-computer interaction. Smith, M.J.; Salvendy, G.; Harris, D.; Koubek, R.J. (eds.), (LEA, Inc., Mahwah, NJ, 2001) p. 1318-1322.

Andersen, H.B., Sørensen, P.K., Weber, S., and Sørensen, C. (1996): A study of the performance of captains and crews in a full mission simulator. Risø National Laboratory, Roskilde. Risø-R-916.

Endsley, M.R. (1993). Situation Awareness in Dynamic Human Decision Making. Proceedings of the 1st International Conference on Situational Awareness in Complex Systems, Orlando, February 1993.

Endsley, M.R. (1995) Towards a Theory of Situation Awareness. Human Factors, 37 (1), 32-64.

Gerson, Elihu M., and Star, S.L. (1986), "Analyzing Due Process in the Workplace," TOIS, vol. 4, no. 3, pp. 257-270.

Hauland, G. (2003) Measuring Team Situation Awareness by means of Eye Movement Data. In: Proceedings. Vol. 3. Human-centred computing. Cognitive, social and ergonomic aspects. 10. International conference on human-computer interaction (HCI International 2003), Crete (GR), 22-27 Jun 2003. Jacko, J.; Stephanidis, C. (eds.), (Lawrence Erlbaum Associates, Inc., Mahwah, NJ, 2003) p. 230-234.

Heath, C., and Luff, P. (1991), Collaborative Activity and Technological Design: Task Coordination in London Underground Control Rooms, in ECSCW '91. Proceedings of the Second European Conference on Computer-Supported Cooperative Work, ed. by L. Bannon, M. Robinson and K. Schmidt, Kluwer Academic Publishers, Amsterdam, 1991, pp. 65-80.

Rasmussen, J., Pejtersen, A.M., & L.P., Goodstein. (1994). Cognitive Systems Engineering. New York: John Wiley.

Rochlin, G. I., La Porte, T. R., and Roberts, K. H. (1987) The Self designing High-Reliability Organization: Aircraft Carrier Flight Operations at Sea, Naval War College review, Autom.

Strauss, A., Fagerhaugh, S., Suczek, B. and Wiener, C. (1985) Social Organization of Medical Work, University of Chicago Press, Chicago and London.

This page intentionally left blank

CHALLENGE OF SAFETY DATA ANALYSIS – TOP MODELS WANTED

or "Don't call me a cab, when I ask for a map"

Jari Nisula

Operational Monitoring & Human Factors, Flight Operations Support, Airbus[16]

Abstract: Modern Flight Operations Monitoring tools have satisfied the hunger for safety data on minor events and deviations – and thus set the scene for very proactive safety management. There remains, however, the conceptual challenge of how to learn effectively from such data. While the initial case-by-case analysis is usually straightforward, the classical keyword-based analysis methods only give limited support to a more proactive trending type of analysis on the whole database of events. The paper suggests that new methods based on top-down safety models are a very promising option in facing the challenge. The promises and main uncertainties of such methods are discussed. The paper also argues that the term "risk management" is often used lightly in the context of safety processes, which are far from rigorous and systematic risk assessment and management.

Key words: Incident/accident analysis, aviation

1. INTRODUCTION

The aviation industry has traditionally used safety related data from everyday operations to make the aviation system safer. Reporting, data analysis and corrective actions are carried out by different organizations and at different levels.

[16] The views expressed in this paper are solely the personal opinions of the author and do not bind or necessarily reflect those of Airbus, any of its affiliates, or its advisors.

The last decade, and especially the last few years have gradually changed the methods, tools and available data to a significant degree. Some traditional difficulties were washed away almost in one go, just to show the underlying challenges clearer than ever.

It has been clear for a long time that accidents and serious incidents are too rare and random to be used as the single source for safety analysis and lessons learned. Different kinds of reporting systems have been used for decades to facilitate identification of threats before they escalate to serious incidents. The recent arrival of powerful Flight Operations Monitoring (FOM) tools has suddenly enabled the collection of vast amounts of detailed information covering virtually every single flight of an airline. Flight Data Monitoring (FDM) is a process where hundreds of flight parameters are constantly recorded on Quick Access Recorders (QAR) where the data is stored on optical discs. The discs are read by dedicated software at the airline, and pre-defined events are automatically detected. Specialists then carry out further analysis. FDM has already been mandated in some countries, and year 2005 will see a big part of the world mandating operators to run a FDM program. FDM has been complemented with observation techniques like LOSA (Line Operations Safety Audit) or the Airbus-developed LOAS (Line Operations Assessment System), which are based on human-made observations on the flight deck. Simultaneously, software tools have arrived to improve traditional reporting systems.

These new techniques and methods have established an extremely rich source of safety data, and – quite importantly - pushed the focus towards more and more minor events, like minor deviations from ideal flight path during final approach or excessive bank angles at lower altitudes. This extension of focus has underlined the challenges of analyzing data where the content is no longer a long sequence of events with a known (harmful) outcome, but rather a single (causal) factor, which in itself is not damaging, but could be in another context. How do we assess the risk of such an event? How do we store event information so that it can be used effectively in later analyses? The bottom line is: how do we prioritize safety actions. These challenges form the guiding theme of this paper.

The terms "Risk Management" or "Safety Management" are gradually becoming the reference for managing flight safety activities in an airline (Civil Aviation Authority 2002, Workshop on Risk Analysis and Safety 2002). Ideally, risk management would require taking a wide and holistic look in all safety threats for the activity in question, assessing different risks, and then tackling them – starting from the issue associated with the highest risk. The growing demand for such an approach highlights the difficulty in

trying to allocate risk to minor "everyday" events and threats. The situation is not made easier by the key terms themselves: "risk" and "safety" are abstract terms with much more in-built complications than their casual everyday use suggests (Nisula 2002). People talk fashionably about "risk management" when a detailed look into the concept would reveal just how difficult true risk management would be.

Despite the huge power of modern FOM tools and some very successful safety programs – especially FDM related – one could argue that the overall efficiency ratio of safety data analysis is not impressively high (Paries et al. 1996). Vast amounts of data are reported, collected, processed and analyzed without ever resulting in any improvement actions. On the other hand, numerous chances to learn more are certainly missed because patterns are not recognized in the vast databases of safety data – or because some valuable aspects of the events were not recorded in the first place.

This paper argues that the barrier of inefficiency is not that of lacking technical means, but that of lacking conceptual models. We need conceptual tools to show how different functions contribute to safe flight operations – and how different elements of threat and risk can combine to create incidents and accidents. Only such models can help us learn efficiently also from minor events. We do not need yet another piece of data, we need a map to show us where that piece of data plays a role.

2. TYPICAL ANALYSIS METHODS AND THE CHALLENGE

Whatever the source of the safety data, one can usually distinguish two[17] levels of analysis performed on them: case-by-case analysis (or clinical analysis) and the long-term analysis.

[17] We could consider the analysis of Flight Data to represent a third type of analysis. Significant events are analyzed one by one, but the main analysis is based on statistics on different event types, their trends and correlations with other event types. The statistics lead the analyst to take a deeper look into some particular issues and even in some individual flights – the process is thus a constant switch between statistical and clinical analyses. Regrettably, the scope of this paper does not allow a detailed discussion of this quite successful analysis process. See [9] for a discussion of the different aspects of the FDM process.

2.1 Case-by-case analysis

Typically, a flight safety manager receives an Air Safety Report (ASR) and reads it through. S/he sees if there are elements requiring immediate actions and tries to identify key issues in the report and perhaps allocate related keywords before storing the report in an ASR database. If further investigation is needed, s/he will coordinate getting the necessary inputs from all applicable departments.

Similarly, within an aerospace manufacturer, incoming reports are often analyzed in expert teams, ensuring the coverage of all aspects of the events (design, training, procedures, pilot proficiency, human factors, system knowledge, etc.).

The case-by-case analysis relies mainly on the technical (and human factor) skills of the people involved, and one could say that with a proper team with good knowledge, it is probably fulfilling the function well. Safety issues are identified and actioned.

There are two main reasons why limiting safety work to case-by-case analysis is insufficient by itself. First, treating every issue as a single case does not allow the safety manager to see the "big picture": the underlying patterns and problems and the development of different issues over time. Secondly, the scope of the events analyzed this way is often limited to the more serious events - an expert team is not called to analyze every minor deviation detected in the flight data. Consequently, this process is not proactive enough to be used as the only safety process. In line with this, a major accident investigation can be seen as a large-scale case-by-case analysis of one event.

2.2 Long-term analysis

The real challenge is the analysis based on a larger set of safety data – data where each event[18] has usually already been analyzed in a case-by-case manner when it was entered into the database. The long-term analysis should unfold hidden patterns in the safety data and create lessons learned which were not derived from the case-by-case analysis. Experience shows that this is not an easy job to do. Figures from aviation and other similar safety-minded industries (e.g. nuclear) show that typically less than 5 % of data in

[18] The term "event" is used here because most databases deal only with events – unfortunately. We could as well talk about "safety issues" or "safety concerns".

safety databases are used for launching concrete actions based on the long-term analysis (Safety data analysis workshop 2003).

The term *long-term analysis* is used here for two reasons. First, this type of analysis requires a reasonable amount of data - usually obtained over a long time period. Secondly, the term *statistical analysis* was deliberately avoided because its scope is too restricted. While statistics on event facts (e.g. weather, destination, time) and on some keywords can be invaluable, they are not enough to respond to the full challenge of long-term analysis. In the complex flight operations environment, the prioritization of safety issues cannot be done simply by comparing their frequencies. Similarly, even a clearly increasing frequency of a certain event type is not necessarily enough alone to justify a safety action. This also explains the author's persisting allergy to the term "trend analysis".

There is a link between the type of safety events to be analyzed and these analysis methods. Higher-level events (e.g. incidents, accidents) give ample material for a rich case-by-case analysis, whereas the "bits and pieces" of very minor occurrences do not really lend themselves for this analysis method. As said earlier, the safety management focus has been moving more and more towards these "bits and pieces" of safety information, and the pressure to find suitable analysis methods is increasing.

2.3 The challenge

Let's look at the core of the challenge: the type of safety data that a safety manager at an airline is typically dealing with.

Example 1: *"An 80-year old passenger found smoking in the toilet"*

Example 2: *"In cruise at flight level 350 the aircraft encountered standing wave activity and moderate turbulence. A cabin crew member was hurt while taking her seat when ordered to do so."*

Example 3: *"Crew: False Localizer capture ILS*[19] *15 following [approach procedure] arrival, failure occurred just before [location]"*

Example 4: *"Unsecured cargo pallet in rear hold."*

[19] ILS = Instrument Landing System

The reader can appreciate the difficulty of risk assessment of such minor events by trying to decide which of the events carries the highest risk. Asking other people to repeat the exercise reveals how subjective the assessment is.

Each of the above events contains one element of risk (or "causal factor"). The element alone – on a good day – is not enough to cause an accident, and *did not* cause an accident in the real life event either. However, such factors might cause an accident *together* with some other *additional* elements and/or in another context. The questions to answer now are[20]:

- What are those other elements?
- What would be their probability to combine with the reported element?
- What is then the *risk* involved with these events?
- How can we store the core information in these reports in such a way that it can contribute to safety lessons in the future, e.g. through the long-term analysis?

Before moving further, we must examine the concept of risk. Engineers see risk as the product of the *probability* of an outcome and the *severity* of that outcome. When we try to assess "the risk" in a historical event, what are we actually doing? Factually speaking, as long as traveling through time is ruled out, the risk that a historical event (which did not end in an accident) would end in an accident is zero. What we really are after is: "what is the risk that *something similar* in the future ends in an accident?" Using example 1, we would ask: "what is the risk that in the future a passenger will smoke in a toilet and the sequence will end in an accident?"

Answering such a question forces us to build imaginary scenarios, which develop into accidents thanks to additional risk elements (or "threats" or "causal factors"). The probability and the severity of each scenario outcome are a function of our choice of additional elements. To continue with the previous example, we can build different scenarios:

[20] The situation is quite the opposite in the clinical analysis of accidents (and the like) because the event sequence ties together all causal elements, disclosing their roles and consequences. This creates the opposite problem that in hindsight it seems like the sequence "couldn't have gone any other way", i.e. the probability of other causal paths and outcomes is underestimated – leading to the unjustified conclusion of "they should have seen this coming". This phenomenon is called the Hindsight Bias [11].

Scenario 1: *Passenger smoking in the toilet, detected by the smoke detection unit and/or the cabin crew. No consequences.*

Scenario 2: *Passenger smoking in the toilet during approach, detected by cabin crew but delaying the cabin preparation and distracting the cabin crew, thus leaving some bottles and glasses in the cabin, and some passengers fastening their seat belts late during final approach, creating the potential for injuries in case of turbulence.*

Scenario 3: *Passenger smoking in the toilet during cruise flight over the Pacific, paper towels in toilet catching fire, poor crew performance leading to spreading of the fire to the cabin, resulting in several fatalities due to smoke or loss of control of the aircraft.*

Using our imagination this way, we can see that *virtually any event* allows us to move from a "probable-non severe" scenario to a "very improbable-very severe" scenario, just by adding elements, i.e. varying what we include in *"something similar"*. The resulting scenarios also have different *risk* levels.

How do people then allocate risk to such events in real life? Many software tools offer some type of probability-severity matrix for this purpose, but how do analysts pick the "right" square? They must decide how remote scenarios they still consider reasonably possible, and then estimate their probability and the severity of the related outcome. While doing so, they are faced with several problems. Building scenarios, estimating probabilities[21] and judging severities are all based on the analyst's subjective, implicit and partly unconscious models of safety in the system under study. Biases and heuristics further distort the estimates. Recent discussions that the author had with some flight safety managers reveal that even their basic strategies for scenario building and risk assessment differ fundamentally: some want to stick to the event exactly as it happened in real life without any further scenario-building, whereas others stress they want to think about what *"could have,* realistically, happened". Here the word *realistically* is loaded with all the above-mentioned subjectivity. Furthermore, the analysts would only cover scenarios that they *know about,* and *think of* at the time of the analysis. They would also be unable to explore the different scenarios with their probability-severity combinations

[21] It frequently happens that analysts have to revisit a probability estimate because new events prove the scenario more probable than expected.

systematically like a computer – the human result is more at the level of a *feeling*.

Based on these limitations, the author argues that the validity and value of applying the current simplistic approaches in the risk assessment of minor events are questionable.

What is missing is a framework, which shows the roles of causal factors in different accident scenarios. We must acknowledge that for risk assessment, analysts always refer to "safety models". It is our responsibility to replace the subjective, implicit models by shared, explicit models, which can be built and challenged by expert groups. The real challenge is thus on the conceptual side of creating meaningful and practical safety models.

3. LOOKING FOR SOLUTIONS

The traditional answer for organizing the chaos of safety data has been the use of keywords. Typically, database entries are characterized with keywords, in the hope that these would catch the essence of what happened – and in the hope they would facilitate the long-term analysis by keyword-filtering. This approach has several known drawbacks:

- **Subjectivity:** Use of keywords and risk classification is not consistent between different analysts, even when trained identically.
- **Novelty of results:** Any keyword set is a self-fulfilling prophecy, because non-existing keywords cannot be used and matters represented as keywords may attract extra attention.
- **Causal synergy:** The living dynamism of the event sequence is lost. The database "does not care" if two keywords come from the same event or from two different events.
- **Causality:** It is not always clear which factors can be accepted as causes of an event and where to stop in the exploration of the causal sequence ("the big-bang syndrome").
- **Capturing positive lessons:** Keywords usually reflect negative aspects, failures. The valuable lessons of positive aspects and successes are largely missed.
- **Risk management:** A keyword structure typically does not provide any framework for risk assessment.
- **Productivity:** Typically only 5% of data is re-used thanks to the keywords (Safety data analysis workshop 2003). Most databases

also have a significant percentage of events without any keyword allocation.

Ambitious keyword systems easily become unpractical monsters. In the late 1990's The Australian Bureau of Air Safety Investigation (then called BASI, now part of the ATSB[22]) was studying a new safety reporting and analysis tool SIAM[23]. The old tool OASIS[24] had an impressive keyword structure of 1400 descriptive factors. The study revealed the following shocking but not unusual facts (Lee & De Landre 2000):

- On an average year, only 50% of descriptors were used
- 29% of descriptors were not used at all
- If 75% of least used descriptors had been removed, it would have affected only 0.5% of the events!!

Either we have to accept the highly imperfect situation where the main focus is on the case-by-case analysis and risk assessment is based more on personal intuition than a robust method, or we have to find completely new approaches to safety data analysis.

4. TOP-DOWN APPROACHES IN SAFETY MANAGEMENT

Perhaps the most promising way to go is the development of so-called top-down analysis methods. The idea is that before starting to analyse event data, models are created, showing how different factors contribute to different accident types. Typically, this is achieved by making a structured presentation of the vital safety functions necessary for safe operation (for the scenario in question). There is a finite number of vital safety functions, whereas trying to directly model the failure conditions would be an endless task. Safety functions can be developed into more detailed structures of safety assumptions (or *safety principles*) with the help of well-known techniques like FMEA[25]. Safety principles are positive statements like "technical quality of radio communication is acceptable", The model, which represents the "a priori" understanding of the situation, is then put to the test of reality by feeding it with safety data. Safety lessons come through changes in the model – either in its structure or in the confidence given to

[22] Australian Transport Safety Bureau
[23] Systemic Incident Analysis Model
[24] Occurrence Analysis and Safety Information System
[25] Failure Mode and Effect Analysis

particular safety principles. An advantage of the positive statements is that one can easily capture both negative *and positive* lessons from safety data. The models also highlight the high number of assumptions made on the human operators – assumptions, which are often not challenged enough.

This approach differs fundamentally from classical bottom-up approaches, where the raw data itself is used to help define the classes or keywords, which are then used to organize the data.

4.1 Promises of the top-down approach

The top-down approach is promising, because it could overcome most drawbacks of classical bottom-up applications:

- **Subjectivity:** A common, expert-group-made safety model is more accurate than individual, subjective models
- **Novelty of results:** The analysis process is opening questions, rather than trying to match the event with predefined keywords
- **Causal synergy:** The model is not used to re-create the events. The full event narratives are used.
- **Causality:** The safety model presents the *common and official defenses and safety assumptions,* thus any factor handicapping or eliminating them *is* a causal factor, and factors not presented on the model *are not* causal factors[26].
- **Capturing positive lessons:** Success of safety principles in each event are also recorded
- **Risk management:** The place of each safety principle in the overall safety model is visible, which gives a good starting point for risk management.
- **Productivity:** Even minor events typically link with at least one safety assumption, and lessons are captured from virtually all pieces of safety data.

[26] For example, the skill requirement of a typical line pilot to perform an ILS approach would be represented on the model, and a failure to perform an ILS approach would be considered a causal factor. Things like improvised flying techniques which could have prevented a certain accident would not be listed on the model, because they are not standard requirements for pilots, and the failure to perform such an in-hindsight-obvious escape maneuver would not be accepted as a causal factor.

4.2 Examples of top-down safety data analysis methods

There have been several attempts to apply top-down safety models – either using numerical probability values or more qualitative measures[27].

One of the first such tools in civil aviation was the British Airways–developed RATBAG (Risk Analysis Tool for British Airways Group). While looking like a classical Probabilistic Risk Assessment tool, it must be given the credit of having produced a large top-down framework of positive safety principles. The nominal outputs of RATBAG are updated probability figures. This enables the user to simulate the impact of changes in the operation. However, just having the qualitative model itself is at least half of the benefit (Savage 2000). Figure 1 illustrates the RATBAG model.

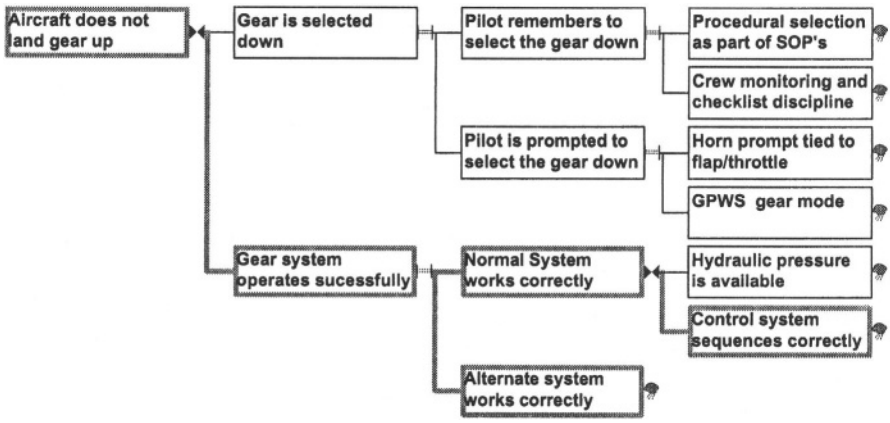

Figure 1. Extract of a RATBAG model (Savage 2000).

Perhaps the most advanced and detailed work on this domain has been carried out by Airbus, Eurocontrol and the ADAMS[28] 2 research team, in three separate projects – all supported by the consulting company Dedale SA. A FMEA process was applied on high-level safety functions for a safe flight to identify initiating situations to accidents ("threat" in figure 2). A more precise content was defined for each initiator for each flight phase. The full model would consist of one Safety Architecture (SA) (see figure 2) per initiator per flight phase. Each SA contains a structure of safety principles (SP) whose role is to first prevent reaching the initiator (prevention), then

[27] It is interesting to note that the analysis power of Flight Data Monitoring is partly based on the fact that FDM is in some aspects a top-down method. In FDM, deviations are measured against pre-defined reference flight profiles: positive definitions of safe performance. This also forms the basis for risk assessment in FDM.

[28] Aircraft Dispatch and Maintenance Safety; under the European Commission 5th framework

prevent reaching the accident (recovery), and finally, in the case of an accident, mitigate its consequences (mitigation). Safety principles can be broken down to sub-structures of more detailed safety principles. For example, an SP "thrust asymmetry is detected" could split into two sub-SP's covering the detection by the pilots and by the aircraft systems. The Safety Architectures cover all aspects coming into play in the scenarios: aircraft, human operators and the environment. This sets high demands to the expert group responsible for building them.

Figure 2. A Safety Architecture gives a structured presentation of the Safety Principles protecting the aircraft against a given threat and accident type, in a given flight phase.

The ultimate idea would be to have a software tool incorporating all Safety Architectures and all necessary functions to implement the methodology. The analyst would be guided through a single event analysis with the help of the corresponding Safety Architecture(s). The tool would keep track of how many times different safety principles have failed and gradually build a global picture of all accident scenarios and the robustness of the corresponding defenses (SP's).

The initial experience of event analysis with the method was promising, but evaluating safety decision-making using the method would require a large set of Safety Architectures, which nobody has for the time being. The full set of SA's would amount to about 200 and represent a demanding and expensive exercise requiring long-term allocation of many experts.

4.3 The future

Even with the considerable effort put in the development of top-down safety analysis methods there are still several uncertainties about the final usefulness of such methods. Can such models represent the complexity of real life to the extent that the analysis results are reasonably correct? What

are the resources needed to create the initial a priori safety models, keep them up to date and do event analysis? Can this method detect issues, which would otherwise be missed? Many people are optimistic, but the final answers could only be given with a functional tool, which is man-years away. There is also the challenge of breaking existing mentalities: it is not easy to start thinking in terms of positive safety principles after years of thinking in terms of failures.

As the full top-down model for aviation does not seem to be available very soon, the development of local models should be encouraged. Many organizations have developed models for their own limited needs. The ADAMS 2 program applied the approach on a few maintenance error scenarios, and Eurocontrol on level busts and ATM[29]-related safety devices. Some airlines have identified precursors to different accident types and the Flight Safety Foundation has hosted the development of the Flight Operations Risk Assessment System (FORAS) (Hadji et al. 2002), which has looked into CFIT[30] scenarios. Such local models could one day contribute to a more global top-down safety model – especially if the developers keep this option in mind when developing their local models.

Concerning risk assessment of minor events in the absence of a top-down model, one could hope that we could move away from the simplistic method where the analyst picks one square from the risk matrix to represent the event. We could try to develop a method where the analyst studies systematically several additional causal factors and the resulting scenarios. Each scenario would be rated in terms of probability and severity, and the risk of the original event would be derived as a combination of these probabilities. Existing data in the database could be used to help estimate the probabilities and severities.

5. CONCLUSIONS

Terms like safety and risk, which are used very casually in everyday conversations, are in fact difficult concepts to work with. Many current practices of "risk management" would not fulfill the criteria of scientific rigor: too often risk assessment is based on rough subjective estimates.

[29] Air Traffic Management

[30] Controlled Flight Into Terrain: an airworthy aircraft under the control of the flight crew is flown unintentionally into terrain, obstacles or water, usually with no prior awareness by the crew.

The challenge of learning from safety data and *managing safety* is especially big when trying to deal with long-term analysis of multiple minor events, i.e. trying to learn extra lessons from a database of minor safety issues, which have already been analyzed clinically. Classical methods – often based on keywords – are quite disappointing in this sense.

Based on the initial experience, the so-called top-down methods seem promising in many ways, but practical applications of these methods are not visible in the near future. The final question to be answered is whether such methods optimize the effort invested in risk management. Any improvements in modeling accident scenarios and structuring risk assessment should be welcomed. Hopefully, the switch to next generation methods in both areas comes soon.

REFERENCES

Civil Aviation Authority (2002) (UK) – Safety Regulation Group. *CAP 712 – Safety Management Systems for Commercial Air Transport Operations.* Civil Aviation Authority. West Sussex, UK, 2002.

Hadji M. et al.(2002) Flight Operations Risk Assessment System (FORAS). *Proceedings of the Joint meeting of the FSF 55th Annual International Air Safety Seminar IASS, IFA 32nd International Conference, and IATA, Dublin, November 4-7.* Pages 367-373.

Lee R. & Joanne De Landre. (2000) *Systemic Incident Analysis Model (SIAM) – A new approach to safety information.* BASI (now ATSB). Presentation.

Nisula J. (2001) *Efficient use of Flight Data Monitoring as a part of airline Flight Safety Management.* Airbus, Blagnac, France, 2001. (copies available from jari.nisula@airbus.com)

Nisula J. (2002) *Flight Safety Management: Towards Risk Assessment and Safety Measurement.* Presentation given to the 15th Airbus Human Factors Symposium, Dubai, 18-20 June 2002. Airbus, Blagnac, France, 2002. (Symposium CD-ROM's available from jari.nisula@airbus.com)

Paries J. et al. (1996). *Development of a Methodology for Operational Incident Reporting and Analysis Systems; Final Report.* Direction Générale de l'Aviation Civile (DGAC). 1996. Pages 5-9.

Proceedings of the fourth Workshop on Risk Analysis and Safety Performance Measurement in Aviation. Atlantic City, New Jersey, August 27-29, 2002.

Proceedings of the National Workshop on Risk Analysis and Safety Performance Measurement in Commercial Air Transportation. Rutgers University, New Jersey, July 20-22, 1999.

Proceedings of the second Workshop on Risk Analysis and Safety Performance Measurement in Aviation. FAA William J. Hughes Technical Center, Atlantic City, New Jersey, August 22-23, 2000.

Proceedings of the third Workshop on Risk Analysis and Safety Performance Measurement in Aviation. 2001

Safety data analysis workshop at CENA 2003) (Centre d'Etudes de la Navigation Aérienne), Toulouse, 11.3.2003.

Savage J. (2000). Risk Analysis by Dependency Modelling. Presentation to the (British Airways Safety Information System) *BASIS User Conference 2000.* British Airways.

Transport Canada (2001). *Introduction to Safety Management Systems.* TP 13739E, 04.2001. Transport Canada, Ottawa, Canada.

Woods et al.(1994) *Behind Human Error: Cognitive systems, Computers and Hindsight.* CSERIAC State-of-the-art Report 94-01. The Ohio State University. Pages 177-183.

This page intentionally left blank

SEMOMAP

SEquential MOdel of the Maritime Accident Process

Jens-Uwe Schröder
World Maritime University, P.O. Box 500, 20 124 Malmö, Sweden

Abstract: An important issue of accident investigations is to find out how barriers installed in technical systems failed and enabled interactions of active failures and latent conditions to develop into the accident. For this purpose models and data taxonomies have been developed to facilitate the reactive risk assessment process after an accident. This paper is intended to focus on maritime accident investigation and the specific models and approaches used for the above-mentioned purposes. In line with current international research activities a sequential model of the maritime accident process (SEMOMAP) and a related data taxonomy will be introduced. Furthermore, results of a first trial application will be reported.

Key words: maritime accident analysis, maritime accident modelling, maritime accident data taxonomy

1. INTRODUCTION

Accident investigations are an important part of the safety standard definition process. They reveal existing safety gaps and offer the opportunity to review existing safety standards or to adopt new regulations. Safety gaps are often linked to failing barriers in human-machine interaction. For this reason many accident investigation and risk assessment approaches concentrate on barrier or defence analysis (e.g. Reason, 1990, with his *Swiss cheese analogy*). Safety barriers are integrated in each technical system. It is usually in the design phase, when decisions are made, that consideration should be given as to which barriers should be integrated and how they should function. An important question is how the evaluation of effectiveness of the barriers can be made in each system. The sad truth is that barriers are analyzed mostly in the follow-up process to an accident,

since day-to-day routine working data are not stored in industries other than the nuclear industry and incident reporting schemes have not yet been successful in many industries (e.g. *IMISS* – Ferguson, 1999). For the purpose of the more specific argument in this paper shipping was chosen as a transportation mode, because it is generally considered risky and therefore needs further consideration with regard to the improvement of safety standards. In the shipping industry the reactive approach to accidents is still the most significant driving force in ship safety (Psaraftis, 2002).

The International Maritime Organisation (IMO), as the specialised United Nations (UN) agency for maritime affairs, has always considered the improvement of ship safety through accident investigation as one of the most important objectives of the organization. In addition to the IMO, further approaches by international and national bodies and institutions were made in order to enhance maritime accident investigation. Apart from pure legal considerations, one of the main issues has always been how maritime accident investigation can be scientifically supported. In this regard some methods have been developed to be used for maritime accidents. Some of them have been adapted from other transportation disciplines, such as aviation (e.g. *SHEL* – Hawkins, 1987), whereas others were solely developed for shipping (e.g. *SAFIR* of the Norwegian company BASS). As far as human-machine interaction is concerned general models, e.g. Reason's (1990) *Hybrid model,* have been applied very often.

One point for discussion is certainly the question of whether accident investigations should focus on the characteristics of the specific industry branch or transportation mode involved. As far as the understanding of human-machine interaction is involved, general approaches are helpful. However, if the improvement of a certain system is the main focus of the investigation a high degree of abstraction might not always be desirable. This philosophy could have been the reason for the European Commission (EC) to stimulate research (such as BERTRANC, CASMET or THEMES) on specific maritime accident and risk assessment methods. In line with the findings and objectives of the above mentioned research activities *SEMOMAP – a SEquential MOdel of the Maritime Accident Process* was developed. This paper is intended to introduce *SEMOMAP* and a related data taxonomy developed for *SEMOMAP*. Furthermore, first applications will be reported.

2. CURRENT MODELS AND APPORACHES USED FOR MARITME CASUALTY ANALYSES

Accident causation models were first introduced when H.W. Heinrich (1931) started to look at accidents in the early 1930s. Overviews about the development of models and taxonomies and current approaches to accident and human reliability analysis can be found in Kirwan (1994), Reason (1990) and Hollnagel (1998). Johnson's (2003) recent book contributes to this topic as well.

The IMO makes specific reference to Reason's (1990) *Hybrid model* and *GEMS,* as well as to Hawkins' (1987) SHEL model in its documents when human failure is targeted during maritime accident investigation. Whereas *SHEL* describes the different types of interaction in human-machine interaction systems, the Hybrid Model is looking at accidents primarily from an organizational point of view. In addition, *GEMS* describes types of behaviour and related errors. All models can be applied to any accidents, no matter which transportation mode is involved.

Apart from the IMO, the EC has supported intensive research projects on specific maritime accident investigation and risk assessment approaches. The most prominent projects/actions are BERTRANC, CASMET and THEMES. In aviation, many specific models i.e. on decision-making processes in the cockpit are available, yet only a few specific maritime accident causation models have been developed over the years. The models/taxonomies that are discussed in the reports of the projects above are among others:

- *CASMET* approach (CASMET);
- *TRIPOD* (Reason, 1997);
- *Loss causation model* (DNV);
- *Systematic learning from incidents* (Kristiansen, 1995).

In addition the following databases were consulted:

- *DAMA* (Norway);
- *M-SCAT* (Norway, DNV);
- *SYNERGY* (Norway);
- *SAFIR* (Norway);
- *MINMOD* (USA).

The above-mentioned approaches/models/taxonomies have been successfully used in the maritime industry for years. Most of them are based on accident causation models. *TRIPOD,* the *Loss causation model, M-SCAT* and *SYNERGY* look on the accident primarily from the organisational point of view, identifying failed defences prior to an accident. The *CASMET*

approach is sequential in its data taxonomy. The model used in the approach is not an accident causation model but rather a human factors interaction model. Kristiansen's *Systematic learning from incidents* and the taxonomies of the remaining databases *SAFIR, MINMOD* and *DAMA* are not based on a specific accident causation models.

One question that might arise at this point is why it is necessary to focus on specific systems or transportation modes, rather than applying general models and taxonomies. One argument for the focused approach an accident investigation is certainly that the data resulting from these investigations are further used for detailed risk assessment purposes. In order to support detailed risk assessment solid data about the accident process as such are required, including human performance data. None of the above approaches collects information about the emergency management process. This type of data is important, too, specifically since some defences (e.g. structural fire protection on ships) start to function only after the accident was initiated. This was the motivation to develop *SEMOMAP,* a model that is more focused on the accident process and how defences function during this process rather than the context in which the accident process occurs. However, as many scholars (e.g. Hollnagel, 1998) point out, a process cannot be evaluated without taking the context in which it occurs into consideration. This is why data that contributes to the assessment of the impact of performance shaping factors (PSF) have been included into the *SEMOMAP* taxonomy too. The following paragraphs introduce *SEMOMAP* and its taxonomy in more detail.

3. SEMOMAP

SEMOMAP (Figure 1) focuses on the human operator during the maritime accident process, while specifically targeting the emergency management process. The model considers features of the approaches mentioned in the previous paragraph. The accident process has been subdivided in *SAFIR* (latent, initialising, escalating, critical phase) and *CASMET* (a step model is used for the coding of the data). The defence principle, which is used in *TRIPOD* can be found in an applied way in *SEMOMAP*. The approach of *SEMOMAP,* however, is slightly different from the above-mentioned models. *SEMOMAP* is based on the Model of Human Recovery and Human Error Management (van der Schaaf, 1992) and the Model of the Navigation Process in an Emergency Situation (Modell des Schiffsführungsprozesses in einer Notsituation – Hahne et al., 2001). It is much more focused on the question of why some accidents develop into total

losses and why others can be successfully mitigated at a certain level of the accident process. In this respect, defences in the human-machine interaction have a vital role to play.

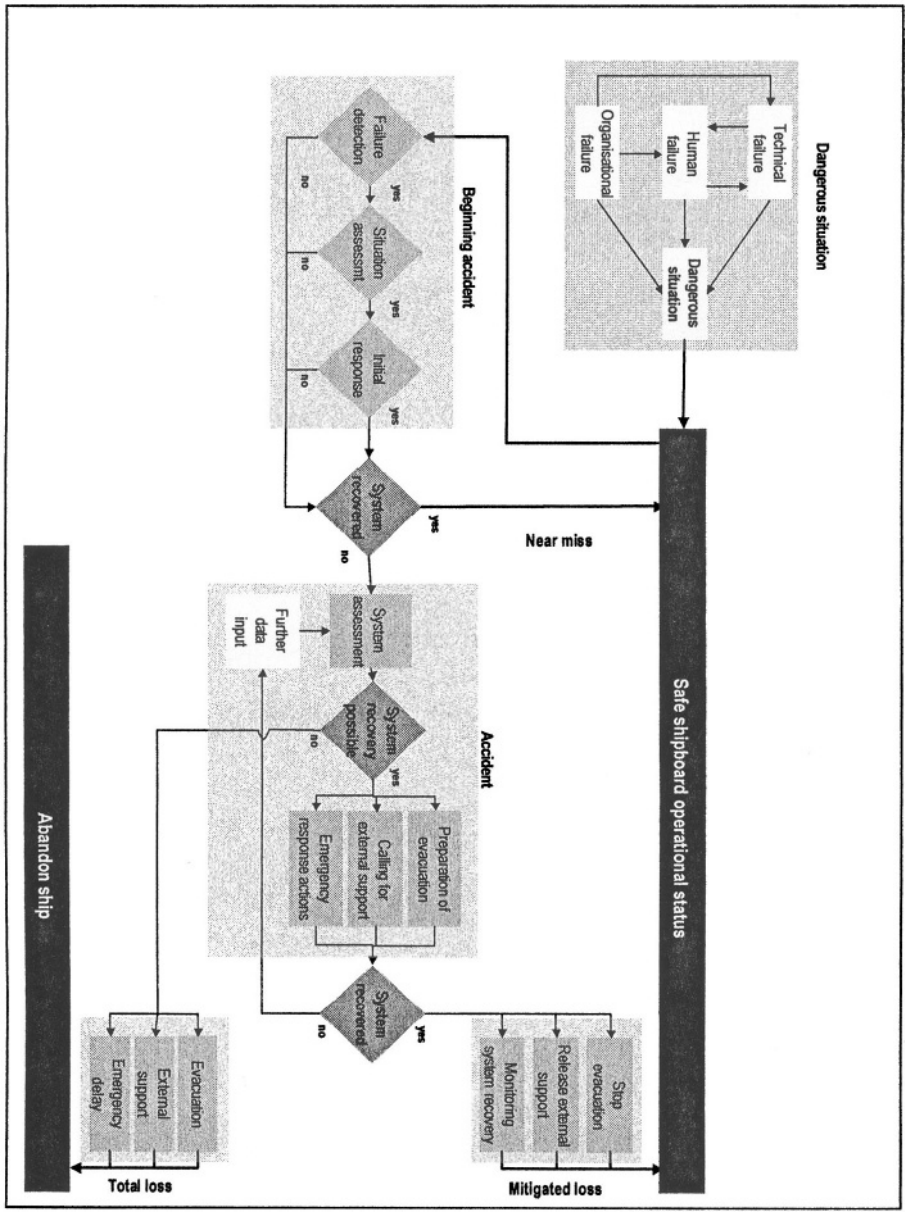

Figure 1. SEMOMAP – SEquential MOdel of the Maritime Accident Process (Source: Schröder, 2003)

SEMOMAP is a sequential model. In order to support the focus on the accident process as such, the impact of the human element context was not integrated. This was considered in the taxonomy, where information about influencing factors on human performance were collected. The idea of *SEMOMAP* is once again to focus on the question why in some systems defences fail in a way that an accident resulting in a total loss, where as in other systems the accident process can be stopped at a certain point due to appropriate emergency response measures. *SEMOMAP* therefore subdivides the accident process into six stages/results (Schröder & Hahne, 2003):

- Dangerous situation;
- Beginning accident;
- Near miss;
- Accident;
- Mitigated loss;
- Total loss.

4. SEMOMAP DATA TAXONOMY

Keeping Hollnagel's (1998) Method, Classification, Model (MCM) framework in mind a data taxonomy has been developed for *SEMOMAP*. An attempt was made to reconcile and combine the input of the relevant IMO instruments (i.e., Code for the investigation of marine casualties and incidents – Res. A.849(20), Res. A.884(21)) and the input of *SEMOMAP*. Special emphasis was laid on the requirement that observations and interpretations should be separated (Hollnagel, 1998). The data collected through application of the taxonomy should be free of interpretation to a high degree. The application of the taxonomy should also lead to the indication of safety critical issues for further improvement. The *SEMOMAP* taxonomy is displayed in Figure 2. Data sections in the figure which are marked with "…" indicate that additional data are required to be entered in the particular section. This would mean, e.g., that "Additional data for the accident category" distinguishes between the following four accident categories:

1. Collision;
2. Grounding;
3. Inrush of water;
4. Fire.

The IMO (Res. A.884(21)) focuses on pollution discharged from ships in addition to the four categories included in *SEMOMAP*. When *SEMOMAP* was designed the emphasis was laid on serious accidents. This is why the

pollution as such was not included under accident categories into *SEMOMAP* to date.

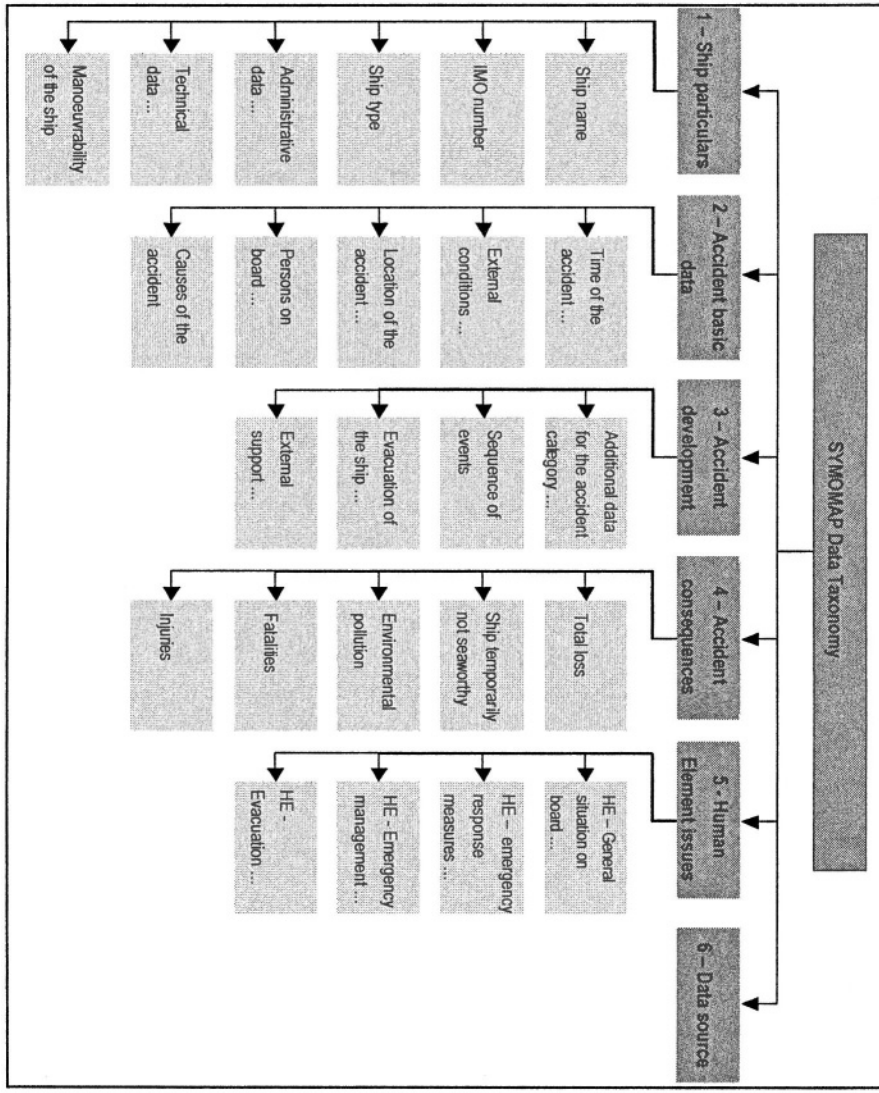

Figure 2. Structure of the SEMOMAP taxonomy

The single *SEMOMAP* accident categories are also further detailed. The accident category "Fire", as an example, is divided into the three sections, as Figure 3 demonstrates.

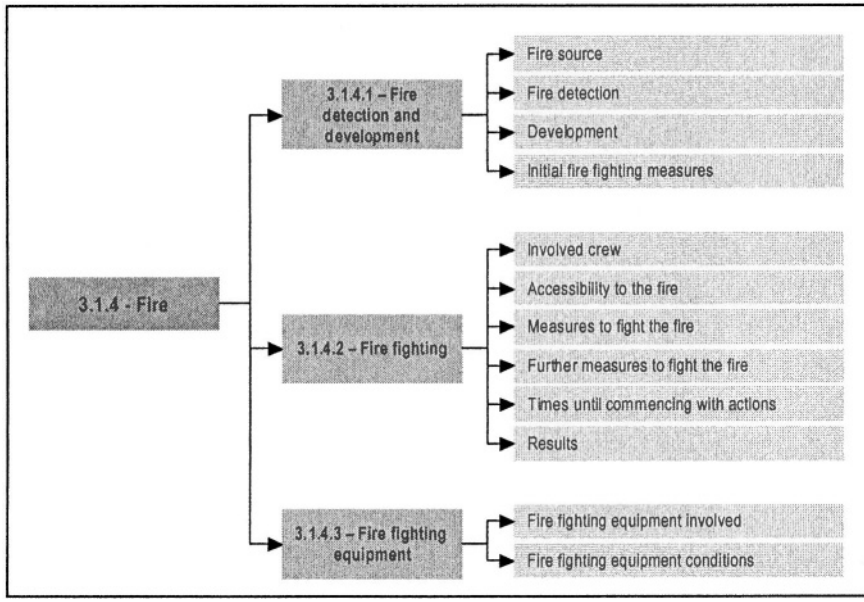

Figure 3. SYMOMAP data for the accident category "Fire" (Source: Schröder, 2003)

The subcategory fire fighting equipment deals with information that is not directly related to the accident. It is important information which is required, when, e.g., management influence on the safety policy of a company is the focus of an investigation. The taxonomy tries to follow the implications of Reason's (1990) models with regard to decisions made/or lack of control at certain management levels resulting into favourable conditions for accidents and incidents. It may appear that the data collected in this category does not directly relate to *SEMOMAP*. This is partly true, as *SEMOMAP* concentrates on the accident process as such. However, accidents are the result of a causal chain, which is influenced by latent conditions that Reason (1990) focused up on in his models. As a result of these considerations, the *SEMOMAP* taxonomy was extended to data providing indications about the latent conditions in the system to be investigated, e.g. the status of the fire fighting equipment.

This also applies to HE related data and data describing the PSF. It is not only the more technically focused data in the taxonomy that are specifically detailed. The HE part is detailed, too, as Figure 4 demonstrates. In the HE part of the taxonomy especially the PSF cover a substantial part. There are 130 different data pieces collected in this section. The data structure for the PSF related data "HE – General situation on board" can be taken from Figure 5.

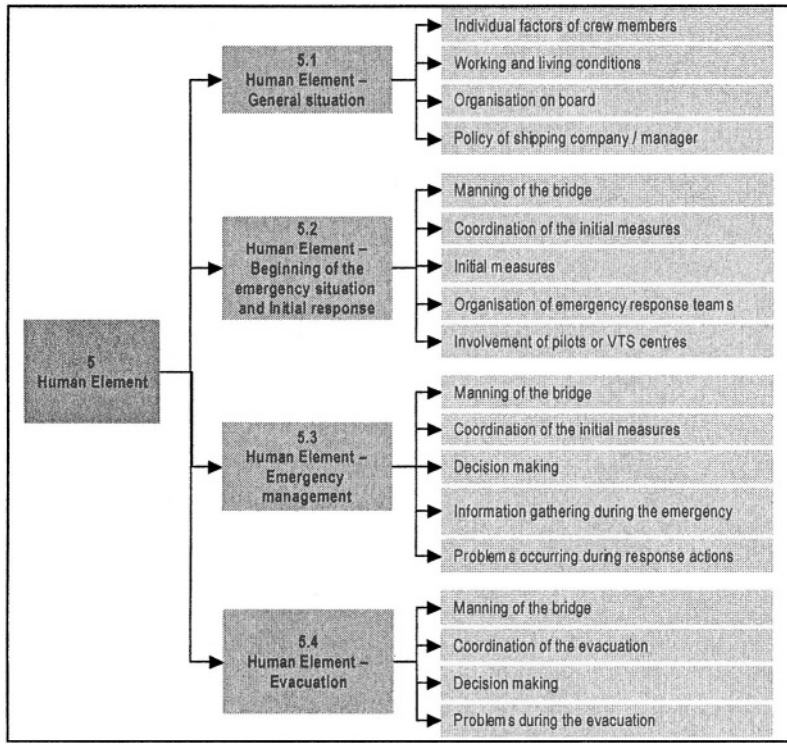

Figure 4. Overview about the HE related data in *SYMOMAP* (Source: Schröder, 2003)

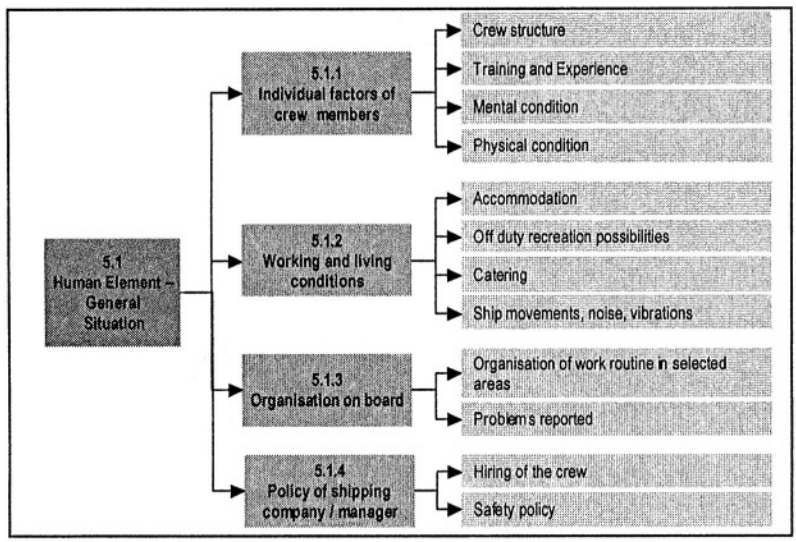

Figure 5. Data from section "HE – General situation on board" (Source: Schröder, 2003)

4.1 Relation between SEMOMAP and its taxonomy

The objective of the taxonomy was to reconcile and combine the relevant IMO guidance and *SEMOMAP* impact with an approach that can be used by practitioners. The taxonomy, then had to be structured in a way that it can be used for checklists or databases, following the general principles of the IMO guidelines (Res. A.849(20) and Res. A.884(21)). The priority was to assemble all data that belong to a certain technical issue at one point in the taxonomy. This resulted into the problem that the *SEMOMAP* taxonomy cannot be directly linked to the model. Table 1 highlights this issue and demonstrates how the link between model and taxonomy functions.

Table 1. Link between *SEMOMAP* and the *SEMOMAP* taxonomy in the beginning accident phase for the accident category fire (Source: Schröder, 2003)

SEMOMAP	Related taxonomy section	
2. Beginning accident		
2.1 Failure detection	Fire detection and development	• Fire detection • Development
2.2 Situation assessment	HE - Beginning of the emergency situation and initial response	• Organization of emergency response teams
2.3 Initial response	Fire detection and development HE - Beginning of the emergency situation and initial response	• Initial fire fighting measures • Manning of the bridge • Coordination of initial measures • Initial measures • Involvement of pilots or VTS centres

5. OBJECTIVES OF THE SEMOMAP APPROACH

The model and the taxonomy can be used for accident analysis and safety defence analysis.

5.1 SEMOMAP and accident analysis

The *SEMOMAP* approach is not exclusively focused on the accident process. It can also be used by stakeholders in the maritime sectors that do not focus primarily on the accident process. In order to avoid different systems being used for accident investigations it would be impracticable to design a system in which only the accident process is displayed. Therefore accident causation was included as a factor in the *SEMOMAP* framework. Accident causation data enable the analysis of the whole accident process.

The data scheme allows for the quantitative risk assessment. For this purpose, investigators can contribute to the database through a complete set of information required for those purposes. The model can be used to link the data in a meaningful way. Current data gaps could be filled through expert judgement. In this respect Bayesian networks could be a possible application for the risk assessment with the *SEMOMAP* framework.

5.2 SEMOMAP and defence analysis

Another objective of the taxonomy is to support defence analysis. The philosophy applied to this approach is that *SEMOMAP* should support the observation and in-depth analysis of a system. This means there should be no interpretations from the system in the data acquired, and only observations should be entered. If there are safety gaps or weak barriers the data will point into that specific direction. This means that the interpretation has to start once the observations from the system are recorded into the data scheme. This may sound unsystematic, however, the model links the information in a meaningful way. A similar approach can be found in the *TRIPOD* methods (Reason, 1997), where data is collected, that has to be combined in a specific way in order to highlight areas of improvement. Although defences are integrated with a purpose into a system, it is not always demonstrated from the beginning how effective they function. In order to allow for an unbiased assessment of those defences, a taxonomy should be used that is not directly focused on the specific barrier, but rather on the overall performance of the system as such.

6. TRIAL APPLICATIONS OF THE SEMOMAP TAXONOMY

The *SEMOMAP* taxonomy was applied to investigation reports of 42 passenger vessel accidents which have occurred between 1979 and 1999 (Schröder, 2003). The findings were basically in line with studies in which a larger number of accident reports have been used (Hahne et al., 2001; Dreissig, 1996), as far as descriptive indicators about the general circumstances of maritime accidents are concerned (e.g. distribution of the accident risk over the day). In general, the results could be used to identify simple trends in the casualties involved. Detailed and comparative observations were often not possible due to missing data. The accident investigation reports were often detailed as far as technical data for the description of the physical accident process are concerned as Table 2 shows. Therefore the general level of reported technical data can be considered as

sufficient enough to reach conclusions. However in many particular sectors, specific information is missing. In addition to missing data about actions of the crew it was often not possible to gather data in the human performance area, as Table 3 shows.

Table 2. Missing data in the accident reports of the accident category "Fire/Explosion" (Source: Schröder & Zade, 2002)

Data category	Data sub-category	Percentage of missing data [%]
Fire detection and development	Fire source	0
	Fire detection	0
	Development	12
	Initial fire fighting measures	0
Fire fighting	Involved crew	100
	Accessibility to the fire	18
	Measures to fight the fire	12
	Further measures to fight the fire	70
	Times until commencing with actions	33
	Results	6
Fire fighting equipment	Fire fighting equipment involved	27
	Fire fighting equipment condition	45

Table 3. Missing data with regard to HE aspects in the investigated accident reports (Source: Schröder & Zade, 2002)

Data category	Data sub-category	Percentage of missing data [%]
HE – Beginning of the emergency situation and initial measures	Manning of the bridge	81
	Coordination of the initial measures	40
	Initial measures	43
	Organization of emergency response teams	77
	Involvement of pilots or VTS centres	93
HE – Emergency management	Manning of the bridge	96
	Coordination of emergency response actions	59
	Decision making	93
	Information gathering during the emergency	74
	Problems occurring during the emergency response actions	90
HE – Evacuation	Manning of the bridge	96
	Coordination of the evacuation	14
	Decision making	89
	Problems during evacuation	81

7. SUMMARY

In this paper *SEMOMAP* and its related taxonomy were introduced together with a trial application of the taxonomy to accident reports on passenger ships. Although the taxonomy could be applied and general trends

could be derived, the following gap has to be considered, which is the missing in-depth data concerning human operator performance. Reasons for this occurrence are related to insufficient information provided on HE background information in the accident reports investigated. The period in which the accident investigations took place covers a period of 20 years, starting from 1979. At this time HE was not considered a critical issue. Many approaches to deal with HE in shipping were more systematically addressed after the capsizing of the Herald of Free Enterprise in 1987. Since that time HE has been on the agenda of the IMO. Accident reports issued after this time usually reveal HE related data too a much more detailed extent. However, the US Coast Guard (1998) concluded:

> "Human Factors: A high percentage of casualties are due to human error. However, quite frequently Investigating Officers fail to document the underlying reasons why the human error occurred. To provide a better understanding of these causes, a description of the human factors should be set forth in the facts."

The National Transportation Safety Board of the USA (NTSB, 2002) came to a similar conclusion after assessing the contents of databases to be used for accident investigation follow-up processes. However, even if not all data are included in the accident reports they are very often recorded during the investigations and remain filed at the investigation bodies. This would probably be a more accurate source for in-depth studies on PSF and related data. The results of the *SEMOMAP* investigation seem therefore to be in line with the general situation of HE related data in the maritime field. The question is if the *SEMOMAP* framework can contribute to overcome the described difficulties.

7.1 Strength and weaknesses of the SEMOMAP framework

The strength of the *SEMOMAP* framework is certainly its focus on the overall accident process. New technical achievements in shipping, such as the Voyage Data Recorder (VDR) or the Automatic Identification System (AIS) will certainly provide maritime accident investigators with an increased amount of data that provides a deeper insight into the emergency management. This is where models like *SEMOMAP* are needed. The limitation of the framework is the taxonomy. The objective was primarily to facilitate the establishment of a database for maritime accidents. The data taxonomy has been developed with this objective in mind. As a consequence

the relation between the model and the taxonomy is not always consistent. The taxonomy is also quite large. This was a result of the consideration that specifically the data on the PSF has to be included. A difficulty could occur if the final result is now impracticable for handling during an accident investigations process and difficult to analyse in the follow-up process to an accident. Those questions can only be answered during the evaluation of the *SEMOMAP* framework.

7.2 Outlook

Another reason for the partially unsatisfactory data situation mentioned in this paper is that accident reports are generally not the most appropriate data source for studies like this. Accident reports reflect only on the most important findings and details of the investigation. It can be concluded that although sufficient data was gathered in order to derive first trends of maritime accident developments, a final validation of *SEMOMAP* is pending. Such an evaluation could be done in different ways. One way would be to invite other researchers to apply the *SEMOMAP* method on a particular accident scenario. Similar evaluations have been done in the past, e.g. within the THEMES project. The results of such evaluations could be considered as subjective. Another problem is that for such an evaluation a well-known and well-documented accident has to be chosen. This could also lead to a certain bias by researchers involved in the evaluation that are familiar with the details of the accident. Assumptions and hypothesises could lead to different outcomes of different evaluation teams. In the absence of other possibilities for the validation of such a system it seems that this is a practicable way.

ACKNOWLEDGEMENTS

The views expressed in this paper are not necessarily those of the author's employers.

The contents of paragraph 2 of this chapter is based on studies of the reports of the EU sponsored projects/actions BERTRANC, CASMET and THEMES. The author wishes to thank the research teams of these projects for this source of information.

REFERENCES

Dreissig, D.(1996): *Wassereinbruchserkennung und hinreichend genaue Prozessdarstellung des Schiffszustandes.* PhD thesis. University of Wuppertal.

Ferguson, S.J. (1999): *International Maritime Information Safety System (MISS).* http://www.uscg.mil/hq/g-m/moa/docs/imissbp.htm (Accessed on 28 April 2004)

Hahne, J., Baaske, G., Moser, H.-J., Rothe, R. (2001): *Identifikations- und Anwendungsprogramme zur Ermittlung von Gefährungssituationen in der Seeschiffahrt.* Schriftenreihe der Bundesanstalt für Arbeitsschutz und Arbeitsmedizin. Forschung Fb 924, Bremerhaven: Wirtschaftsverlag NW.

Hawkins, F.H. (1987): *Human factors in flight,* Aldershot: Gower Technical Press.

Heinrich, H.W.(1931): Industrial accident prevention. New York: McGraw-Hill.

Hollnagel, E. (1998): *Cognitive Reliability and Error Analysis Method CREAM.* Oxford, New York, Tokyo: Elsevier.

Hollnagel, E. (2002): *Accident Models and Accident Analysis.* http://www.ida.liu.se/~eriho/AccidentModels_M.htm (Accessed on 10 February 2004).

Johnson, C.W. (2003): *A Handbook of Accident and Incident Reporting.* Glasgow: University of Glasgow Press.

IMO (1997): *Res. A 849(20) "Code for the investigation of Marine Casualties and Incidents".* London: IMO.

IMO (1999): *Res. A 884 (21) "Amendments to the Code for the investigation of Marine Casualties and Incidents".* London: IMO.

Kirwan, B. (1994): *A Guide to Practical Human Reliability Assessment.* London: Taylor & Francis.

Kristiansen, S. (1995): An approach to systematic learning from accidents. The Institute of Marine Engineers Conf. Proc. on Management and Operation of Ships - Practical Techniques for Today and Tomorrow (IMAS 95). Vol. 107, n°2. London: The Institute of Marine Engineers.

National Transportation Safety Board (2002): Transportation Safety Bases. Safety Study NTSB SR-02/02. Washington: NTSB.

Psaraftis, H.N. (2002): Maritime Safety: To Be or Not to Be Proactive. WMU Journal of Maritime Affairs, 1(2002), 3-16.

Reason, J. (1990): Human error, Cambridge: Cambridge University Press, 1990.

Reason, J. (1997): Managing the risks of organizational accidents. Aldershot: Ashgate.

Schaaf, T.W. van der (1992): Near miss reporting in the chemical process industry. PhD thesis. Eindhoven University of Technology.

Schröder, J.-U. (2003): Zur Ermittlung von Unfallursachen und begünstigenden Faktoren für Unfälle in der Seeschifffahrt. PhD thesis. University of Wuppertal.

Schröder, J.-U., Hahne, J. (2003): Maritime casualty analysis – an adequate basis for simulation during maritime education and training? MARSIM' 03 International Conf. on Marine Simulation and Ship Manoeuvrability: The Society of Naval Architects of Japan, Japan Inst. of Navigation, Int. Marine Simulator Forum, Volume I, RA-24.

Schröder, J.-U., Zade, G. (2002): The impact of maritime casualty investigation on maritime administration and maritime education and training. International Congress on Maritime Technological Innovations and Research, Bilbao, 6-8 November 2002. Bilbao: University of the Basque Country, 287 – 293.

US Coast Guard (1998): Marine Safety Manual. Vol. 5: Investigations, Ch. 3 Marine Casualty Investigations. http://www.uscg.mil/hq/g-m/nmc/pubs/msm/v5/c3.htm (accessed on 10 February 2004).

This page intentionally left blank

THE TEAM-BASED OPERATION OF SAFETY-CRITICAL PROGRAMMABLE SYSTEMS IN US COMMERCIAL AVIATION AND THE UK MARITIME INDUSTRIES

C.W. Johnson
Department of Computing Science, University of Glasgow, Glasgow, G12 9QQ, Scotland.
http://www.dcs.gla.ac.uk/~johnson

Abstract: This paper analyzes a range of incidents involving team-based interaction with safety-critical programmable systems. The incidents were submitted to NASA's Aviation Safety Reporting System (ASRS) and to the UK Marine Accident Investigation Branch (MAIB) between December 2001 and February 2003. Our results show that incidents, which complicated the team-based operation of safety-critical, computer systems in commercial aviation, are now being reported within the UK maritime industry. This reflects the increasing use of programmable navigation and collision avoidance devices both in ferry operations and in commercial fishing. For example, many incidents in both industries now stem from operators making inappropriate assumptions about the likely behavior of co-workers and their programmable systems even though part of their task is to actively monitor those activities. Our results also show that a growing number of incidents are triggered when teams must rapidly reprogram complex, safety-critical systems in response to unpredictable changes in their operational requirements.

Key words: Programmable Systems, Accidents, Human Error, Team Decision Making.

1. INTRODUCTION

The introduction of computer-controlled systems has led to novel forms of failure in many different industries. For example, a UK MAIB (2002) report recently described how a fishing vessel deviated from its course and grounded; "the autopilot had developed a fault prior to arriving at the port,

and although the skipper had attempted to have it repaired, the fault remained unresolved. He was aware of the wisdom of checking the autopilot against the compass heading, but apparently failed to do so on this occasion. With no obvious indication to remind him that the autopilot was not working, he engaged it with misplaced confidence". Such incidents remind us that the introduction of complex, computer-controlled systems can paradoxically increase the need for team-based interaction. The MAIB argued, "A second person on watch would have enabled the autopilot malfunction to be identified, and remedial action to be taken. With no redundancy, the skipper was reliant on the correct operation of the navigational equipment and his ability to maintain a proper lookout".

The importance of team-based interaction for the operation of safety-critical systems has led to the development of training techniques such as Crew and Bridge Resource Management (Sexton et al, 2000). These provide guidance on how to coordinate teams of co-workers during adverse events. They also include training in more routine team-based operating procedures, including the call back of commands. CRM and BRM are widely perceived to have averted many potential accidents. For instance, reporting systems such as NASA's Aviation Safety Reporting System (ASRS) provide important insights into the successful team-based interventions that avoid potential incidents. In the following incident, computer-related warning systems and the vigilance of the crew resolve a potentially dangerous situation created by Air Traffic control. A commercial aircraft taxied to the approach end of the runway. The Captain then noticed an aircraft on TCAS, which appeared to be landing. The First Officer confirmed the pilot's observation; "When the TCAS was indicating 700 and 500 [feet] for the aircraft on Final, I asked the First Officer if the aircraft was landing. He stated that it was still landing. I initiated a turn off the runway and advised the Tower that we were clearing the runway. Tower asked if we needed assistance. I stated, '[No. I just didn't want to sit on the runway with that aircraft on short final'. As I turned the aircraft around towards the runway, the other aircraft, a Learjet, landed on the runway exactly where we had been in position" (ASRS, 2002c).

The team-based operation of safety-critical computer systems provides a barrier against individual human error. Co-workers can monitor and intervene to support interaction between colleagues and increasingly complex systems. However, team-based operation also creates opportunities for different forms of 'error'. Individuals may rely on their colleagues to correct their mistakes; co-workers can introduce distractions and can exacerbate the effects of individual 'errors' (Sasou and Reason, 1999, Sexton, Thomas and Helmreich, 2000). Incident reports provide insights into these behaviors. They provide glimpses of the failures that characterize

everyday interaction with safety-critical computer systems. Incident reports also describe near misses. They, therefore, provide insights into team-based interaction as a barrier to more serious failures.

This study looks at two very different domains. US commercial aviation represents a high-technology industry characterized by a relatively small number of large companies. In contrast, the UK maritime industry has a far larger proportion of owner-operators. These industries also differ in terms of the computational technologies that they rely upon. Computer-based control and navigation systems are part of the fabric of US commercial aviation. In contrast, many fishing vessels are just beginning to incorporate computer-based control systems. Ferry operations exploit these programmable devices in greater numbers. In both cases, there is arguably a greater degree of redundancy and a larger margin for error than is the case in US commercial aviation. As we shall see, however, there are strong similarities between the incidents that complicate the team-based operation of safety-critical, programmable systems across these different domains.

2. TEAM BASED 'ERROR'

Many incidents in US commercial aviation and the UK maritime industry reveal the limitations of team-based problem solving (Johnson, 2003). For instance, the second officer of a roll-on roll-off passenger vessel recently attempted to close the vessel's bow doors prior to leaving Calais. He experienced a series of problems in operating the automated control system and called for assistance from the chief and third engineers. He also requested help from an electrical officer. They eventually abandoned the automated system and attempted to close the visor manually using instructions displayed next to the control station. The starboard visor ram and support arm began to buckle and the operation was stopped immediately. An investigation revealed that the starboard support arm-locking bolt was still in the engaged position. None of the team had noticed a light on the control panel indication, which indicated that the doors were still secured. As a result, two additional control system-indicating lights were fitted to show the position of the locking bolts and modifications were made to the operational instructions (MAIB, 2001). This incident illustrates the importance of good interface design for team-based interaction with complex, programmable systems. The ergonomics of control panel design can prevent operators from observing important warnings when colleagues obscure their view. This incident also illustrates the importance of incident reports in identifying the limitations of team-based problem solving. Groups can compound adverse events as well as resolve them. Rather than

explore the reasons why the Second Officer could not complete the operation using the automated system, the group started to manually close the visor even though it was still secured in the open position. A number of researchers have attempted to explain such team-based behavior. Green, Muir, James, Gradwell and Green (1999) describe how "many pilots like to be thought of as fairly bold individuals, and combining a set of such individuals into a crew can make for an unduly bold outcome". This 'risky shift' represents a form of polarization in which groups of individuals whose members are predisposed to accept or to reject a risk will have their predispositions reinforced by being members of that group. Conversely, others have sought to stress the positive role that team-based decision making has upon the operation of safety-critical systems. Bowers, Bickensderfer and Morgan (1998) argue that there is no legacy of 'rugged individuals within air traffic management and so "there may be no need for awareness-phase seminars or other interventions designed to improve negative attitudes". They stress the ability of Air Traffic Control teams to construct shared mental models both of the computer systems that they operate and of the intentions of their colleagues.

In order to understand the dual nature of teamwork in both promoting safety and introducing new hazards, it is important to summarize the key factors that distinguish group performance from individual human factors. Tjosvold (1989) observes that as groups grow member, participation declines. He also argues that conflicts increase and co-operation decreases in larger groups. This reduction in co-operation can partly be explained by 'social loafing'. Some individuals contribute less to a group than when they are individually accountable (Latane et al, 1979). Team members can distract operators from safety-critical tasks. Diverting attention away from a task can also make operators worried about performing poorly with the result that they become anxious. There is, however, considerable disagreement about the factors that affect group-based performance. The problems of 'social loafing' must be contrasted with Zajonc's (1965) 'drive theory'. He argues that the presence of others can improve performance. Seta, Seta and Hundt (2001) observe that this improvement increases if co-workers are slightly superior to the person operating the system.

3. THE FLIGHT DECK GRADIENT

It can be difficult to find unambiguous evidence for such theories in the documents that are collected by reporting systems. The 'flight deck gradient' refers to the difference in authority or status that can exist between the Captain and First Officer in commercial aviation. Seta et al would argue

that this difference might promote rather than inhibit group interaction. However, the following report describes flight-crew interaction with a Visual Approach Slope Indicator (VASI); "(The Captain) is an experienced pilot, capable and in no way overbearing…The aircraft begins to descend below the VASI indications, giving finally four reds…I presume the descent *(below the correct glide-path)* is intentional ...I inform the Captain we are floating. He seems surprised by my call, but removed power and lands. However, we are between 1/3 to 1/2 of the way down the runway. The Captain appears transfixed by the runway and hasn't engaged reversers as per SOP. I call for reversers and query the autobrake setting of level three out of five available levels. He makes no response. I state that I am increasing autobrake to level four. He doesn't acknowledge. With hindsight I allowed my attitude of respect and friendliness toward the Captain to influence my actions. I was insufficiently assertive once the incident was in progress and prior to the incident I presumed rather than checked the reasons for his flight profile" (CHIRP, 1998). This incident would seem to contradict Seta et al's observation that operator performance improves when higher status colleagues monitor an individual's performance. The pilot reports that he felt inhibited from questioning the actions of a respected co-worker. However, the pilot did eventually intervene. It can, therefore, also be argued that flight crew interaction prevented the incident from having worse consequences. In this interpretation the incident vindicates team-based decision making rather than pointing to a problem with the flight deck 'gradient' (Johnson, 2003).

4. MISPLACED TRUST

Team-based interaction often relies upon a form of skepticism about the ability of co-workers to perform necessary tasks. This alienation helps to ensure that operators check and re-check critical commands during the operation of safety-critical systems. For example, the crew of a B737-800 was informed that Air Traffic Control (ATC) training was being conducted in their sector. The plane leveled off at 2,500 feet, following ATC instructions. The Captain was then instructed to turn right onto a heading of 080 degrees. This would have directed them towards terrain rising over 7,500 feet in approximately 2 miles. The crew refused to turn. ATC again replied, "Right turn 080 degrees." The crew stated that they were "unable to comply due to rising terrain to our right and in front of us". They started to turn left in order to clear the terrain. ATC then asked if the plane was level at 3,500 feet. The crew replied that they were at 2,500 feet. This was the level that ATC had initial assigned them to. Shortly afterwards the left turn

initiated by the crew brought them into sight of the airport. They were then cleared for a visual approach (ASRS, 2002a). This incident illustrates how the successful team-based operation of safety-critical systems paradoxically depends upon the crews' refusal to comply or cooperate with the instructions of their colleagues. If they had done as they were requested then the flight would have been placed in jeopardy.

It can also seem paradoxical that distrust is a necessary prerequisite for crew-based interaction with complex systems. However, complacency and a failure to monitor computer-related systems are two of the most common features of team-related incidents in both US commercial aviation and the UK maritime industries. For instance, the UK MAIB describe the grounding of a container ship even though she followed the same route every week. The vessel also had three qualified deck officers in addition to the master and was equipped with a full range of navigational equipment, including two radars and a Global Positioning System (GPS). Visibility was good and it was a clear dark night. The second officer relieved the third officer at midnight and the ship's position, derived from the GPS, was being plotted on the chart from time to time. The charts in use had the courses to steer marked in black ink and could not be erased. An Assistant Bosun shared the bridge watch. Course was altered at 0025, and again at 0047, with the ship's position being plotted on the chart each time she settled on to a new course. At 0243 she altered course again, to 237° and, once again, the position was plotted. About 45 minutes later, the ship grounded at full speed. The MAIB argued that this incident occurred to a well-equipped vessel with fully qualified officers who were familiar with the passage and had no problems in establishing the ship's position in good visibility. However, the ship was on a regular route, and the courses had been indelibly marked on the chart. The numerals 237 were clearly evident, as was the reciprocal 157 for the return voyage. After the grounding it was found that the automatic steering had been set to 257°. The investigators argued that the officer of the watch had inadvertently set the wrong course, having mixed up 237 with 157 (MAIB, 2001c). In this case, team members failed to detect a transposition error in the programming of the automated steering system. In the previous incident, the Captain failed to detect the First Officer's omission of two fixes in the Flight Management Computer.

5. WRONG INFERENCES ABOUT CO-WORKERS

In the aviation domain, programmable systems can be so complex that incident reporting agencies often comment on the 'unusual' or

'extraordinary' performance of the crew in diagnosing the cause of an adverse event. This is illustrated by a recent report submitted by the Captain of a Boeing 737-300. The following report also describes a similar transposition error to the previous example involving the automated marine steering system. This emphasizes further similarities between aviation and maritime incidents in the team-based operation of programmable systems; "We were at Flight Level 250 when Center cleared us to cross 30 miles west of ABC VOR (very high frequency omni directional range transmission navigational beacon). at 17,000 feet. The First Officer was flying on autopilot and dialed in 17,000 feet in the altitude alerter then started programming the Flight Management Computer (FMC) for the crossing restriction. I dialed in ABC on my VOR...There was no intersection for the crossing point so the First Officer had to build it, which takes time. When the FMC finished thinking, it indicated that we were well below profile, so the First Officer hit VNAV (vertical navigation system), which brought the descent back to 1000 fpm. That didn't make sense so I looked at the descent profile, which verified what the First Officer had indicated. My VOR readout and the FMC did not agree, but I did not realize what was wrong at the time. The First Officer was as confused as I was, but accepted the idle power descent profile... I realized in hindsight that he had no idea what I was basing my concern on… It took a while, but I finally realized that the First Officer had constructed the crossing waypoint correctly but had inserted it after the next intersection instead of before it. The FMC assumed that we were going to fly to the pre-existing intersection then back to the crossing point. Unfortunately the error was caused by a reliance on modern technology which is wonderful but relies upon correct inputs". After the flight, the Captain "showed the First Officer how to verify that constructed intersections are inserted correctly".

This incident illustrates how individuals must perform 'extreme problem solving' in order to address the potential errors that are made by their colleagues. It also illustrates the manner in which the actions of other groups within the aviation system can impose those burdens upon their co-workers. The reporter argued that the entire incident might have been avoided if Air Traffic Controllers could help modern FMC-equipped aircraft by giving crossing restrictions based on predefined intersections that are likely to already be in the on-board database; "any time you have to construct a crossing point, it takes a lot more time and introduces a significant opportunity for error" (ASRS, 2003a).

Many incidents in both the UK maritime industries and US commercial aviation stem from inappropriate assumptions about the intention and actions of co-workers. These assumptions can persuade operators to disregard the evidence provided by computerized warning systems. For example, a ferry

recently touched bottom on departure from a Scottish port. There was clear visibility. The master, chief officer, second officer and a quartermaster manned the bridge. They were all very familiar with navigation in the area. The chief officer monitored the vessel's progress using radar and the electronic chart system. There had been no communication between the master and chief officer about the intentions for the passage out of harbor. The Chief Officer thought the master intended to slow down when a navigation buoy was observed on the starboard bow. The chief officer noticed the vessel was swinging too slowly, and moving south of the safe track. He warned the master on the enclosed bridge wing, who immediately instructed the helmsman to apply more port helm. The order was too late. The MAIB argued that the repetitive nature of ferry work could lead to complacency: "everyone knows exactly what to do and there is no need for anyone to communicate". The vessel was fitted with modern navigational aids. The chief officer, who had sight of the navigational instruments, was monitoring events. He could not, however, accurately interpret the significance of the information provided by automated warning and navigation systems without knowing the Master's intentions once the fishing vessels were seen ahead. A deviation was made from the usual departure plan but the chief officer could not monitor the master's intentions because he had not been told what they were (MAIB, 2001b).

Further incidents stem from misplaced trust in the programmable devices that perform functions, which would otherwise have been performed by crewmembers. For instance, a recent MAIB report describes how a crew of three operated a fishing vessel. Two of them were cooking breakfast, cutting up bait, pumping out the bilges and cleaning pump filters while also maintaining the watch. Meanwhile, the skipper was asleep on the deck of the wheelhouse. The vessel's planned track passed 0.35 miles from a rig. The automated radar alarm system was set to a third of a mile. The vessel's VHP radio was turned off because the skipper argued there was too much distracting radio traffic. The crew of the rig called for help from a stand-by safety vessel that put alongside the boat. Nobody could be seen on the bridge or on deck even after they sounded their horns. The rig went to 'abandon platform stations' as a precautionary measure. A crewmember from the support vessel boarded the fishing boat and found the skipper asleep in a sleeping bag. When the skipper was awakened he was instructed to slow down and steer way from the platform. He did so but protested about being awakened. He claimed that the situation was under control (MAIB, 2002b). This incident illustrates several important aspects of the interaction between teams of operators and programmable control systems, such as the automated radar warning application. In this case, the skipper assumed that his co-workers would maintain an active watch even though they were

engaged in several other tasks. The radar warning system should have been used as a form of safety net or as a final safeguard. However, the group working practices seem to indicate a more routine reliance on this device to prevent the vessel from encroaching upon hazards such as the rig.

6. DISTRACTIONS AND PLAN REVISIONS

As mentioned, many incidents involving team-based 'failures' seem to stem from a form of complacency. There is an assumption that colleagues or automated systems will perform complex tasks in a reliable manner. Unfortunately, as we have seen this is not always the case. The failure to adequately monitor colleagues and programmable systems not only stems from complacency. It can also be the result of competing tasks and other distractions that eat into the time crewmembers have available to perform necessary checks. A previous NASA study of 107 ASRS incident reports identified 21 different types of routine tasks that crews neglected while attending to another task (ASRS, 1998). It is difficult to determine how many of these interruptions related to the operation of programmable systems. However, 69% of the neglected tasks involved either the failure to monitor the current status or position of the aircraft, or failure to monitor the actions of the pilot who was flying or taxiing. 90% of the competing activities fell into one of four broad categories: (1) communication (e.g., discussion among crew or radio communication), (2) head-down work (e.g., programming the Flight Management System or reviewing approach plates), (3) searching for traffic, or (4) responding to abnormal situations. In 68 of the 107 incidents, the crews reported being distracted by some form of communication, most commonly discussion between the pilots, or between a pilot and a flight attendant. This paper avoids such statistical analyses because incident reports are inevitably affected by submission bias. It is difficult to know whether the 107 selected incidents were in any way representative of those adverse events that complicate the team-based operation of commercial aviation systems. A number of statistical techniques can be transferred from the field of epidemiology to address these biases. The NASA study did not exploit these techniques and they remain the subject of current research (Johnson, 2003). In contrast, the remainder of this paper relies on a more subjective comparison based on an exhaustive analysis of incidents reported by the ASRS and MAIB over the last fifteen months.

Having raised these caveats it is important to stress that both the NASA study and our analysis identify the importance of distractions as a precursor to adverse events in the team-based interaction with safety-critical

programmable systems. This can be illustrated by a recent incident in which the First Officer was forced to go 'heads down' in order to reprogram the Flight Management System when there was a late change to their departure runway. Late changes involving the reprogramming of on-board systems create acute vulnerabilities. In this instance, the First Officer glanced up to see an aircraft at the arrival end of the runway in position with all its lights on. "I said to the Captain, 'No. No. No. We are on the runway!' We were supposed to have turned... At the same time, ATC advised us that we had crossed an active runway. The Captain then understood his mistake… He had heard, "Taxi to" and saw the aircraft on Runway 12, so he thought he had been cleared to cross Runway 12… He stated that something did not seem right". Another incident report describes a situation in which neither crewmember detected reprogramming errors that were introduced in response to a late change. The initial departure was rushed to make the airline and Air Traffic Control schedule. The initial "Computer flight plan was route ABC. However, ATC clearance was via route D-E-F. Original flight plan should have been crossed out or destroyed, so as not to accidentally revert to [the] planned route. [The] First Officer was very experienced and I had complete trust that he was capable of loading the correct waypoints, but both he and [I] failed to use a visible method of marking the computer flight plan. ...99% of the time, the cleared route is the same as the computer flight plan, but not always, as I found out the hard way. ATC caught my error". The crew attempted to fly the original route even though Air Traffic Control had confirmed with them that they were only authorized the fly the revised route (ASRS, 2002b).

7. CREW FATIGUE

The previous examples illustrate how relatively complex changes to original plans can induce errors in the programming of automated systems. Incident reports in the maritime industry also reveal problems that stem from more mundane issues including crew fatigue. For example, a vessel recently struck a well-known building in a busy estuary in spite of being equipped with ARPA radar sets and an electronic chart system with GPS overlay. As the Master approached the building, he thought he saw a red light close on the starboard bow. Assuming it was another vessel, he ordered starboard helm. The Filipino second officer confirmed the sighting and when no further lights were seen ahead, the Master ordered hard to port to resume his course. Shortly afterwards, the vessel collided with the building's foundation. The incident investigators argued that the building was conspicuous and the vessel was equipped with advanced navigational aids.

They concluded that the crews' 'errors' could only be explained in terms of the fatigue that is created by hours of operation and by disturbed circadian rhythms (MAIB, 2001f). Similar causes were identified for the 'mistake' that led to a fishing vessel running aground off the Shetland Islands. The skipper had not slept for about 23 hours and attempted to alter course of the vessel using a joystick control. He did not follow the correct procedure for changing from automatic to manual steering. As a result, he did not realize the vessel had failed to turn until immediately before it grounded (MAIB, 2002c).

Aviation crew operating schedules have arguably been more extensively studied and controlled than those of their maritime counterparts. Fatigue plays less of a role in team-based failures in this domain. There are further differences. In US commercial aviation, Air Traffic Control often detects errors in the interaction between crews and on-board automated systems. Maritime incidents often have more serious consequences because they lack this additional safety net. For instance, a roll-on, roll-off ferry recently grounded in the UK. At the time of the grounding the master, the chief officer, a seaman lookout and the bosun as helmsman manned her bridge. The weather and visibility were both good, however, the approach was through a very narrow channel between drying sandbanks. The bridge team followed a familiar passage plan, which involved the master conning the vessel from the bridge. The chief officer was operating the engine controls according to the master's instructions while the duty second officer monitored the navigation using radar parallel index techniques. However, on departure from the berth the second officer had duties at a mooring station and no one monitored the radar in his absence. The rival tasks that preoccupied the Second Officer created the precondition for this incident to occur. This combined with a navigational mistake that was triggered by a critical buoy that was not lit (MAIB, 2001d).

8. RESPONSE TO FAILURE

Operators often incorrectly assume that programmable systems and their colleague will perform the tasks to which they have been assigned. Their assumptions are often based upon previous observations about the reliability of their co-workers and the systems that they operate (Johnson, 2003). Previous incidents have shown that fatigue, distraction and a failure to communicate key intentions can undermine the validity of these assumptions. In other situations, equipment failures impose burdens upon operators that prevent them from fulfilling the expectations of their colleagues and co-workers. A control system failure on a Scottish ferry

illustrates this point. The vessel had two propulsion units, one forward and the other aft. On the morning of the incident, the forward engine had to be started using jump leads from the aft battery. It had insufficient charge to start using its own batteries. Routine pre-operational checks were carried out but, before the main steering controls were tested, the electrical supply was changed from emergency batteries to main power. Following successful tests, the ferry started work for the day. Just before she arrived back, the motorman was given permission to disconnect the emergency batteries to replace a dead cell with a new one. He did so, but found the connecting bridge for the cell was too short. He went ashore to the nearest garage to get a longer connecting bridge. The other crewman also left the vessel to get stores, while the charge hand remained on board. Shortly afterwards, an alarm showed that electrical power had been lost on the main steering controls. The charge hand cancelled the alarm but was unable to restore power. He changed to emergency power and regained control. Shortly afterwards the alarm sounded for the forward main engine. On this occasion he was unable to cancel it and the stern began to slew to starboard. He attempted to correct the movement by using the aft unit but, once again, the controls failed. He tried to restore both main and emergency power, but neither would engage. Unable to do anything further, he allowed the vessel to slew until it settled against the shore. He then called the harbormaster asking him to contact the other two-crew members. The vessel was now at a 90° angle to the slip. The charge hand tried to shut down the forward main engine so that his crew could board over the ramp. The engine failed to respond. With the vessel now moving slowly along a beach, the motorman finally managed to get onboard through the car deck gate. He was assisted by the charge hand, who had left the bridge to help him. Once aboard, the motorman went to the engine room where he found that the emergency battery charger switch had tripped. He reset it and went to the bridge to assist the charge hand (MAIB, 2001e).

This incident again illustrates the dual nature of many incident reports. They provide insights into the problems that can arise when operators fail to intervene successfully in the operation of complex, programmable systems. Equally, they also provide compelling insights into the ways in which team-members respond to initial 'mishaps' and thereby prevent them from developing into more serious accidents. The following report provides a further, more complex example from the aviation domain. Given the increasing introduction of computer-related systems into the maritime industries it may only be a matter of time before the MAIB receive reports of incidents that are similar in complexity to those of the ASRS. A Fokker 70 was descending through 7,000 feet, on radar vectors for a landing when the "on-board computers generated a level III alert, 'Landing gear not down'".

They were well above the alert envelope and traveling faster than the maximum speed at which it would have been safe to operate the landing gear. The pilot noticed that the left seat radar altimeter was reading zero feet. The right seat radar altimeter was indicating the correct altitude and so the crew attempted to switch control to the First Officer's side. "As the descent continued, the flight warning computer added the aural warning, 'Too low gear'. About this time we were given a heading to intercept the instrument landing system final while still descending to 3,000 feet... It was at this time the traffic alert and collision avoidance system (TCAS) added, 'Traffic, Traffic!' As I was looking for the traffic I had to compete with a continuous level III alert chime, "Too low gear" aural alert and now the aural TCAS traffic alert. Again, none of these warnings can be silenced. I looked for the traffic... Sure enough, there was a single-engine high wing aircraft in a left climbing turn. I called out "traffic in sight" about the same time the TCAS started calling, "Climb, Climb!" The pilot flying followed the TCAS guidance and we narrowly missed this aircraft. Somewhere in this sequence the landing gear alert ended... I changed to Tower and the rest of the approach and landing was normal". On the one hand, it can be argued that the crew successfully responded in a flexible manner to this equipment failure. Control was transferred immediately after they noticed the radar altimeter failure. They then divided tasks appropriately throughout the rest of the flight. However, this apparently successful intervention was marred by a number of problems. In particular, the crew were troubled in debrief by their communication over the TCAS warning. The First Officer stated that "a couple of things bother me... I communicated to the pilot flying that I had the aircraft in sight. He could have interpreted this to mean there's no immediate conflict... Had he not followed the TCAS guidance, I think we would have hit the other aircraft" (ASRS, 2002).

9. CONCLUSIONS

This paper has analyzed a range of incidents involving team-based interaction with safety-critical programmable systems. The incidents were submitted to NASA's Aviation Safety Reporting System (ASRS) and to the UK Marine Accident Investigation Branch (MAIB) between December 2001 and February 2003. We have identified strong similarities between incidents in the team-based operation of programmable systems in commercial aviation and the maritime industries. Many incidents in both industries now stem from operators making inappropriate assumptions about the likely behavior of co-workers and their programmable systems even though part of their task is to actively monitor those activities. In the aftermath of adverse

events, operators often argue that monitoring was unnecessary because of the previous reliability record. This seems to indicate that greater training is required in order for operators to understand the likely limitations both of their co-workers and the programmable systems that they operate. Initiatives to introduce Crew and Bridge Resource Management are a partial panacea (Johnson, 2003). They provide operators with general training on the error-inducing mechanisms that complicate the team-based operation of complex systems. However, our results also indicate a number of specific problems that complicate interaction with computer-related systems. In particular, many incidents are triggered when teams must rapidly reprogram complex, safety-critical systems in response to unpredictable changes in operational requirements. The reprogramming tasks are exacerbated by problems of interface design that permit the easy omission or transposition of necessary steps in a sequence of instructions, including navigational markers. They also stem from inappropriate assumptions by co-workers about the ease of reprogramming complex systems, for instance Air Traffic Control may underestimate the difficult crews experience in constructing crossing points for Flight Management Computers.

This paper has relied upon a qualitative analysis of the incidents that were submitted to the ASRS and the MAIB over the last fifteen months. A number of factors biased our work. In particular, we are dependent upon respondents notifying the relevant authorities that an incident has occurred. This elicitation bias is an inevitable problem in using any form of incident reporting to support the management of safety-critical applications. This issue explains our reluctance to perform any direct statistical analysis of incident frequencies given that it is impossible to estimate the under-reporting of particular forms of adverse event. In particular, it is likely that team-based incidents may not be reported if groups of co-workers feel implicated by the events that they have witnessed (Johnson, 2003). In other projects, we are using ethnographic and observational techniques to identify those healthcare incidents that are never reported through more formal channels (Randell and Johnson, 2002). This work has yielded some surprising results. In particular, we have identified coping strategies that users will exploit in order to 'get the job done'. These coping strategies include the 'hot' rebooting of safety-critical programmable control systems. Further work is needed to determine whether these techniques might yield similar insights within commercial aviation or the maritime industries.

REFERENCES

ASRS Callback, NASA Ames Research Centre, (1999) No. 239, (2002) No 273, (2002a) No 274, (2002b) No 275, (2002c) No 276, (2003) No281, (2003a) No 280,

ASRS (1998), Cockpit Interruptions and Distractions, Directline Issue 10, NASA Ames Reasearch Centre, http://asrs.arc.nasa.gov./directline_issues/d110_distract.htm

C.A. Bowers, E.L. Bickensderfer and B.B Morgan (1998), Air Traffic Control Specialist team Coordination. In M.W. Smolensky and E.S, Stein (eds.) Human Factors in Air Traffic Control, 215-236, Academic Press, London.

M. Blakely (2002). Remarks for the Annual Management Conference of the National Railroad Construction and Maintenance Association, National Transportation Safety Board, Miami, Florida, January 12, 2002. http://www.ntsb.gov/speeches/blakey/mcb020112.htm

CHIRP (1998), Feedback No. 46 http://www.chirp.co.uk/air/default.htm

R.G. Green, H. Muir, M. James, D. Gradwell and R.L. Green (1999) Human Factors for Pilots, Ashgate, Aldershot, U.K.

C.W. Johnson, (2003, in press). Handbook of Incident Reporting, Springer Verlag, London.

B. Latane, K. Williams and S. Harkins (1979) Many Hands Make Light Work: The Causes and Consequences of Social Loafing., Journal of Personality and Social Psychology 37:822-832.

MAIB Safety Digest, UK Department of Transport: (2001)Vol.1, Case2. (2001b) Vol 3, Case 9. (2001c) Vol.3, Case 1. (2001d) Vol. 2, Case 3. (2001e) Vol. 2, Case 11. (2001f) Vol. 1, Case 7. (2002), Vol. 3, Case 19. (2002b), Vol. 1, Case 19. (2002c) Vol 1, Case 21.

R. Randell and C.W. Johnson (2002), User Adaptation of Medical Devices. In C.W. Johnson (ed.) Proceedings of the 21st European Conference on Human Decision Making and Control, Department of Computing Science, University of Glasgow, Scotland.

K. Sasou and J. Reason (1999) Team Errors: Definition and Taxonomy, Reliability Engineering and System Safety, 65:1-9.

J.J. Seta, C.E. Seta and G.M. Hundt (2001) Exaggerating the Differences Between Relatively Successful and Unsuccessful Groups: Identity Orientation as a Perceptual Lens. Group Dynamics: Theory, Research, and Practice, (5)1:19–32.

J.B. Sexton, E.J, Thomas and R.L. Helmreich (2000) Error, Stress and Teamwork in Medicine and aviation: cross sectional surveys. British Medical Journal (320)7237:745-749.

D. Tjosvold (1989). Interdependence approach to conflict in organizations. In M. A. Rahim (Ed.). Managing Conflict: An Interdisciplinary Approach. 41-50, Praeger, New York.

R.B. Zajonc (1965) Social facilitation. Science, 149, 269-274.

This page intentionally left blank

TOWARDS A FRAMEWORK FOR SYSTEMATICALLY ANALYSING COLLABORATIVE ERROR

Angela Miguel and Peter Wright
Univeristy of York, Department of Computer Science, Heslington, York, YO10 5DD, UK

Abstract: One of the main difficulties in creating a model-based, predictive error analysis method for collaborative work is establishing a useful perspective with which to describe collaborative work and failure systematically. This paper addresses the need for a systematic approach to analysing possible failures of collaborative work in order to create such a method. An iterative approach to the design of the error analysis method has been taken. This paper first discusses the original approach taken to question development for an error analysis of collaborative work, and then focuses on the creation of an improved approach to question development using the results of an evaluation of the original approach as a guide. The new approach involves enhancing the model of collaboration used, breaking collaboration into three aspects (coordination, cooperation and co-construction) and creating a framework to structure a systematic examination of collaboration in terms of cognitive stage

Key words: Collaborative Work; Collaborative Error; Human Error Analysis.

1. INTRODUCTION

Although the importance of evaluating safety-critical systems for possible human error is well recognised, few of the many existing error analysis techniques consider collaborative error. Most human error analysis techniques focus on errors that might happen during the interaction between a single individual and the system they are using, despite the fact that most work takes place in groups or teams. Collaborative work is susceptible to errors that emerge as a result of the distributed knowledge that this type of work involves, which places extra demands on participants. Collaborative

errors may be caused by factors such as a lack of situation awareness or awareness of each other, misunderstandings between participants, conflicts, and failures of co-ordination. An error analysis method that can tackle these issues is required.

It is now widely accepted in the Human Reliability Analysis literature that a model-based approach to analysing human error is the most valid (see for example Hollnagel, 1998). However, a key difficulty in creating a model-based, predictive error analysis method for collaborative work is establishing a systematic description of collaborative work and failure. The systematic error analysis techniques currently available largely ignore collaborative work. Although fields such as Computer Supported Cooperative Work (CSCW) contain many descriptions of collaborative work there are few helpful models and no systematic approaches for identifying possible types of collaborative error and reasons for it. Models for the architecture of groupware do exist (such as Clover (Laurillau and Nigay, 2002)), but these focus on functionality and not error. Therefore they do not provide detailed information about collaboration of the type needed to base an error analysis of collaborative work on.

The process of designing an error analysis method is complex and it is important to consider the user in designing such a method. Thus, learning from HCI approaches to design, an iterative approach to the design of the error analysis method has been taken. The next section of this paper discusses the original approach taken to using a model of collaboration for question development for the CHLOE technique. The problems discovered with this approach through an evaluation are then briefly discussed. The remainder of the paper focuses on how, using the results of the evaluation as a guide, an improved approach to analysing collaborative error (and hence creating error analysis questions) is created for the error analysis technique.

2. CHLOE

The CHLOE human error analysis technique (Miguel and Wright, 2003) was developed as a model-based method to analyse collaborative work, and help make redesign suggestions. CHLOE considers issues that are important for collaborative work, which would be missed if using other existing error analysis or evaluation approaches. Analysis techniques not designed for collaborative work miss the combinations of factors that may lead to collaborative error. Following the tradition of model-based error analysis approaches such as THEA (Pocock et al, 2001) and CREAM (Hollnagel, 1998), the CHLOE process consists of several stages. These are: Scenario Description, Goal Decomposition, and Error Analysis. The Error Analysis

questions are based on a model of collaboration. Finally, Design Issues are considered according to the outcomes of the Error Analysis stage.

2.1 CHLOE's Model of Collaboration

The error analysis stage in CHLOE is based on a model developed from a basic framework of collaboration by Dix (1998, p.495) and shows various types of communication involved in collaborative work. Failures in collaboration are framed as being failures in communication and understanding between participants. The model is composed of participants (P), an artefact of work (A), different possible types of communication, and shared understanding which develops through the process of communication. A simple cognitive loop (Norman, 2002) has been added to the basic model to represent the cognition of each participant when interacting with other humans or machines.

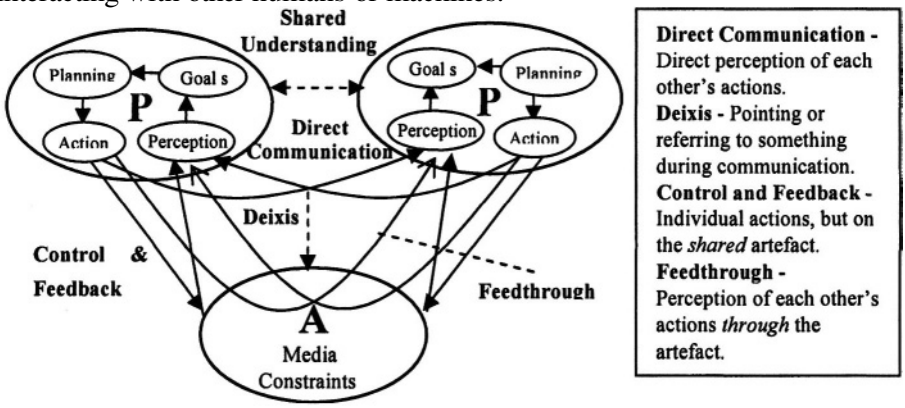

*Figure 1.*The Model of Communication Used in CHLOE

The arcs in the diagram indicate how the participants and artefacts are linked in each type of communication. All types of communication can help to create and support *shared understanding,* which helps collaboration to work effectively.

2.2 Breakdowns in Collaboration

CHLOE views collaborative failures as stemming from breakdowns in the cognition of the participants involved. Basing the error analysis on failures within a cognitive model of collaboration can help lead to design solutions. This is because it provides a perspective on *why* the observable failures occur that is useful for helping to consider design requirements. Error analysis questions were developed by applying guidewords describing types of failure to the cognitive stages in the model of collaboration (see

Table 1). The analysis questions developed from these failures are therefore generally concerned with failures of perception and evaluation, goals, planning, and actions. For example:

- failures of perception/evaluation – e.g. the collaborating participants may not all perceive or interpret the information in the same way, or may not be able to perceive what each other is doing
- failures of goals – e.g. each participant may not know what he/she is supposed to be doing and when, or their goals may conflict

Table 1. Cognitive Stages and Failure Guidewords Used in CHLOE

Cognitive Stage	Failure Guidewords	Cognitive Stage	Failure Guidewords
Perception/Evaluation	Error Failure/None	Planning (Coordination)	Incomplete Error Failure/None
Goals	Conflict Error Failure/None	Action	Error Failure/None Conflict Incomplete

The guidewords used were selected from the SUSI (Chudleigh and Clare, 1993) modified HAZOP (Kletz, 1999) technique, which was developed to analyse user-system interaction.

The types of failure shown in Table 1 were considered in relation to the types of communication shown in the model to create examples of failure such as, a *Failure* of *Perception* in *Feedthrough* (e.g. one participant cannot see the result of the other's actions) or an *Error* of *Perception* in *Deixis* (e.g. one participant refers to something, but the other participant misunderstands what is being referred to). Forty-eight possible failures were created from the combination of the guidewords, cognitive stages, and types of communication in collaboration according to the model used. Twenty-one questions were then developed around the possible reasons for these failures for each type of communication. Example questions include: (Goals Q5) *'Are participant's goals or sub-goals likely to come into conflict?'* and (Planning Q4) *'Is there a shared representation which is consistently visible and understood by all concerned, which can be referred to (e.g. pointing) when sharing information?'*

2.3 Problems with the CHLOE Approach

As part of the iterative approach taken to the development of the method, a small evaluation of CHLOE was performed using seven participants analysing a short air traffic control scenario. These participants were either PhD students or Research Associates in the Department of Computer

Science and had varying amounts of knowledge about Human Factors and Human Error Analysis techniques. They were given an introduction to the CHLOE technique and then provided with written material about both CHLOE and the air traffic control domain. After analysing the air traffic control scenario using CHLOE, the participants completed a questionnaire about the method and its application. The questionnaire was designed to draw out any doubts or dissatisfaction with the method so that improvements can be made. It was split into five sections: modelling, error analysis questions, re-design issues, usability, and effectiveness. Some of the participants were also interviewed later about the answers they had provided in the questionnaire. This was done to clear up ambiguities and collect additional information about their answers.

The evaluation revealed that CHLOE sometimes failed to elicit answers that were specific enough to support detailed failure analysis. It also revealed a lack of consistency between analysts. Vague error analysis questions were a major cause of these problems. These difficulties exist for several reasons. First, the number of error analysis questions is severely constrained by the use of a scenario and task-based approach, which requires questions to be repeated many times over. The amount of effort required for each round of questioning has to be minimized with such an approach, or the overall effort required for analysis would become too great. The questions were kept to a minimum by forcing them to cover a number of issues at once. This increased their tendency to be vague. Second, the few, general failures within the model create too simple a view of collaborative failure to systematically support the development of detailed error analysis questions. Finally, CHLOE uses a model of *communication* in isolation. It is not clear how the wider *collaborative* work context affects it.

The results of the evaluation suggest that to be useful, the error analysis questions need to be more precise in the directions they give to users. If more error analysis questions could be supported (by taking a different approach to analysis), questions could be more specific without a loss of coverage of different types of possible errors. To improve the error analysis, a stronger basis for understanding collaborative work (what collaborative error is, what collaborative work involves, and what it requires to be successful) is needed. The model of communication itself needs to be enhanced to provide more detail about what communication involves. This will help the error analysis questions to become more specific. The model of communication also needs to be interpreted within the wider context of collaborative work to allow the systematic construction of the reasons behind failure and the conditions for success. A framework is then needed to structure the systematic identification of possible collaborative failure using this extra information.

The following two sections explain the approach taken to improve the model of communication and view of collaborative work as a whole. These improvements enable the creation of a framework to identify possible collaborative errors, on which to build a more reliable and effective error analysis method for collaborative work

3. ENHANCING THE MODEL

The evaluation results suggested that the model of communication needed to contain more detail about possible types of communication to enable a more thorough analysis of collaborative work. In order to consider how the model needed to be enhanced, a number of types of collaborative work were examined (for example, work in a neo-natal intensive care unit). The basic Dix model (1998) was used as a basis to structure a categorisation of such collaborative work. It was established that changes to the model were needed in the following ways.

Firstly, directly communicating face-to-face is not considered separately from communicating through a video conferencing system or using email. These are all examples of what is labelled Direct Communication. There are however important differences in these forms of communication that will affect collaboration. These differences (for example temporal and spatial differences) may become important when considering possible errors in collaboration and how the design of the system helps to prevent these. The mode of communication may be important in an error analysis. Mediated communication should therefore be recognised as separate from direct face-to-face communication. The model of communication can easily be altered to do this (see the extra line labelled M linking the participants in Figure 2). The line indicating feedthrough has also been altered to indicate that feedthrough occurs as a result of the interactions of one person with the shared artefact.

The second limitation of the model is that it shows only two participants for the sake of simplicity. Direct communication may be one-to-one as the model depicts, but consideration needs to be given to how communication may be altered if there are more than two participants involved. It is important to consider whether communication is to one, some, or all of the group. It is also necessary to examine whether communication is direct and specific, or general (such as verbalisation or talking out loud). These issues are interesting because they relate to observation and overhearing issues, which are recognised as important for collaborative work because they affect awareness (for example Clark (1996) and Segal (1995)). Segal refers to the unintentional communication that results from deliberate communication

and interaction between participants and artefacts in collaborative work as 'consequential communication'. It is a crucial ingredient for helping collaborative work to function effectively. It is therefore important to establish how these issues relate to the model of communication being used. The level of analysis that the model uses, the elements involved and the terms in which collaboration is described are particularly well suited to the consideration of these issues.

Using the model, possible patterns of observation can be considered in relation to the types of communication shown. This covers both observation of the actions themselves and the results of actions. This information is generally important for awareness concerns, but also more specifically for issues such as checking, reacting to one another's actions, and supporting each other. The ability to observe control and feedback or observe actions through feedthrough may be used deliberately as a means of non-verbal directed communication, or it may simply be a useful side-product of actions that help awareness of others' actions. Observation and overhearing can also be taken a step further to consider the ability for one, some, or all of the group to overhear or see interactions between other participants.

It is also important to consider deixis in relation to overhearing and observation issues. It can be either purely verbal or involve pointing at something, and this may be influenced by the type of communication that it is part of. Deixis may involve an artefact that is visible for all participants communicating, for only one or some of those participating, or it may not be present at all. For this last example, the artefact that is being referred to is not necessarily a physical artefact, but rather something such as procedures or rules (what activity theorists would refer to as psychological artefacts (Nardi, 1996)).

Considering issues such as observation and overhearing in relation to the model of communication allows it to deal better with both intentional and unintentional communication. It can therefore be used more effectively to account for both verbal and non-verbal communication between participants.

Finally, more detail is required about shared understanding. Further enhancement can be made to the model by specifying in more detail what participants may need to develop shared understanding about (for example about each other, or about the work situation). A useful way to structure a more detailed examination of shared understanding is to break it down into past, present and future issues. Participants in collaborative work may require to interpret their interactions with one another in terms of what has already happened (the past), what the current situation is (the present) and/or what they believe the future will be or what they want to it be (team goals) (the future). This refinement of the model of communication enables a more

specific and systematic identification of where shared understanding may fail (Figure 2).

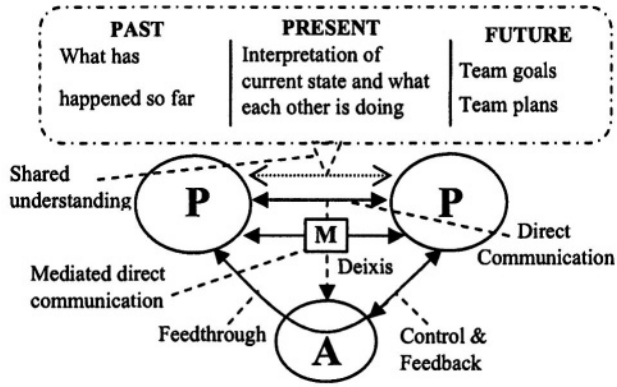

Figure 2. Possible New Version of the Dix Model of Collaboration (the cognitive stages for P are omitted for the sake of simplicity)

There is more to collaboration than just communication and interaction. Having created a more detailed view of communication in collaboration, to understand what is required of these communications it is necessary to take a wider view of what shapes them.

4. A BREAKDOWN OF COLLABORATION

The model and approach used in CHLOE to create possible failure types resulted in an error analysis that does not effectively capture what is necessary for collaboration to work successfully and what a failure of collaboration may be. This is because the model used examined the communication and interaction involved in collaborative work in isolation. It included no representation of the larger activity of which communication and interaction are parts, which makes it difficult to interpret the model. The interactions shown in the model may happen for multiple reasons within collaborative work. What is required of these interactions and how they may fail depends on what they are attempting to achieve and in what conditions. It is necessary to begin with a higher-level consideration of what collaboration involves in order to properly examine work at the level of the communications and interactions shown in the model.

Collaborative work has many different types and aspects. Different forms of collaboration may have different requirements to enable success, or causes for error. Splitting work into types using, for example, the time/space matrix (Dix, 1998, p.488) highlights important differences in types of work, but does not provide a suitable structure with which to examine

collaboration itself. The alternative of dividing collaboration into its component aspects in order to deal with each more specifically was therefore investigated. Through examining work on Distributed Cognition (Hutchins, 1995), Coordination Theory (Malone and Crowston, 1990), Coordination Mechanisms (Schmidt and Simone, 1996), and other existing models of collaboration (for example, Annett and Cunningham (2000), and Gutwin and Greenberg (2002)), the concepts of communication, coordination, cooperation and control were isolated as potentially helpful. Bardram's (1998) split of collaborative work into coordination, cooperation and co-construction and the associated transformations between these levels (Figure 3) was identified as the most useful way to view collaboration to complement the model of communication. It crystallizes what has been identified as important by various other approaches to studying collaborative work, and places them in a helpful structure. The breakdown of collaboration into levels with movements between them provides a dynamic view of the collaborative process. These levels clearly capture different aspects of collaboration that require different relationships between the participants involved in the work to succeed. Thus, this structure provides a useful context for the model of communication because it identifies what shapes the requirements of what this model shows to enable successful collaboration. The levels can also be easily described in terms of goals, planning and actions. This helps to clarify what is involved in work and what is required in the extended model of communication, which already uses these terms to describe cognition.

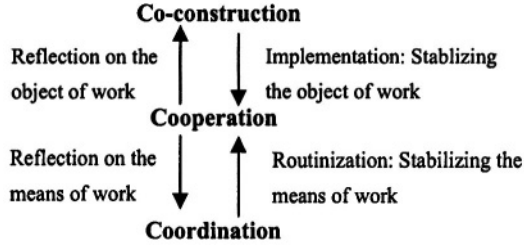

Figure 3. The Dynamics of Collaborative Work (from Bardram, 1998)

In *Coordinated* work, *goals* and *plans* already exist and participants focus on the performance of their individual *actions.* In *Cooperative* work the participants' focus is on their common *goal* and how to achieve it rather than their individual roles and actions. This level of work is more concerned with the *planning* of work because the way to achieve the common *goal* may either not exist or be open to change. Cooperation is therefore concerned with the participants' ability to work with each other to find a way to achieve their shared goal. Finally, *Co-construction* is about the *goals* of work because it involves participants coming together to question the whole aim

of the work and hence goals and sub-goals associated with this. All collaborative work involves all three levels. Therefore all three must be considered to analyse collaboration thoroughly for possible error.

Movement between levels of work can also be explained with reference to group goals, plans and actions. At the coordination level *goals* and *plans* are set. If there is a problem with performing the prescribed *actions,* individuals may have to move into the cooperation level to *plan* together to find a solution. If the *goals* of the activities are brought into question then the participants move to the co-construction level of work. Movement down the levels is caused by resolving the problems with the goals and constructing plans to perform the work.

Now that a useful structure for describing collaborative work has been identified, the model of communication can be analysed in the context of each of these three different levels of work (with their goals, plans and actions mapping) to create a systematic error identification framework. This is created by isolating what is important within the model of collaboration according to the requirements of each of the levels (or aspects) of collaboration. The failure of collaboration is related to the failure of these requirements.

5. NEXT STEPS TOWARDS A NEW ERROR IDENTIFICATION FRAMEWORK

A systematic means of examining what is important within the model of collaboration is needed, according to the requirements of each aspect of collaboration (coordination, cooperation and co-construction). For all aspects of collaborative work there are certain relationships between the elements involved in the work that are important for success. The term 'collaborative error' describes errors that occur because of problems relating to the ability of individuals to perform activities with one another as a group to fulfil group aims. Therefore, the key relationships to examine are concerned with how the individuals relate to one another and the group as a whole.

For collaboration to work effectively it is necessary to consider the relationships between the individuals involved. Individuals must often have shared or complementary goals, interpret a situation in the same way, or perform actions that complement each other. Disagreement between individuals is a potential indication or cause of error. It is also necessary to consider how individuals relate to the group as a whole. Error could be caused by a problem with an individual's understanding of the group's goals or how they are to be reached. Failure could also occur because of the failure

of an individual to comply with the group's aims. The group must also be considered as a whole. The following key relationships in the success of collaborative work have been identified:

- Individual↔Individual (how participants relate to each other)
- Individual↔Group (how participants relate to the group as a whole)
- Group (participants' collective actions have to achieve group purposes)

These relationships are built through, and rely on the communication shown in the model. Hence, they are a useful structure with which to explain the requirements for each level (or aspect) of collaboration, in terms of the model used. All of these relationships are important for each of the three aspects of collaboration (coordination, cooperation and co-construction) for different reasons. Table 2 shows how these relationships need to be considered to examine collaboration.

Table 2. The Structure of the Framework to Analyse the Model of Collaboration

Collaboration								
Coordination			Cooperation			Co-construction		
I↔I	I↔Group	Group	I↔I	I↔Group	Group	I↔I	I↔Group	Group

The requirements of the relationships between individuals and the group can be expressed more precisely in terms of cognitive stages. For example:

- ***Interpretation(I)↔Interpretation(I)*** (e.g. individuals in the group may have to have the same interpretation of something)
- ***Plans(I)↔Goals(Group)*** (e.g. individuals plans may have to contribute to or be in agreement with the group goals)

The relationships between individuals and the group can be examined more systematically and in detail this way. To identify specific ways in which collaboration may fail, the cognitive stages in the model are first used to establish precisely what is required of the relationships between individuals and the group for each aspect of collaboration (coordination, cooperation and co-construction). So to identify the requirements of collaboration using these cognitive stage relationships, the requirements of these relationships for each aspect of collaboration are considered, i.e. the requirements of ***I↔I, I↔Group,*** and *Group* in *coordination* are described in terms of cognitive stage relationship requirements, the requirements of *cooperation* are described in terms of cognitive stage relationship requirements, and the requirements for *co-construction* are described in terms of cognitive stage relationships. For example:

- Coordination:
  - ***actions(I)↔actions(I)*** etc.
  - ***goals(I)↔goals(Group)*** etc.
  - *actions (Group)* etc.

For the purposes of this analysis, the group is considered to have goals, plans and actions, and individuals have the whole of Norman's cognitive loop (2002) as shown in the model of collaboration (Figure 2).

Exactly what is required of which cognitive stage relationships depends on the aspect of work being examined. The coordination, cooperation, co-construction split provides the context of the wider activity with which to understand how the individuals and group have to relate to each other. What is important about these cognitive stage relationships, and in what way, will be different for different aspects of work because of the different structures of the work. Each aspect of collaboration has a different level of completeness of group goals and plans, and different 'awareness of each other' requirements. What is required of a cognitive stage relationship for each aspect of work to be successful can be established based on the description of each aspect in terms of goals, plans and actions. For example, the ***goals(I)↔goals(I)*** relationship for *cooperation* requires that a goal be shared, whereas for *coordination* it is only necessary that the goals are complementary so that they do not conflict. In the context of Table 2, the cognitive stage relationships can therefore be used to consider issues such as: the degree of awareness required of each others' goals and activities, the potential need for shared interpretation or development of shared understanding, the degree of awareness required of group goals and plans, and what controls the coordination of individual work to fulfil the whole group aims. The systematic examination of what is important about these relationships for each aspect of work and why it is important will help to clarify the requirements of collaborative work to function effectively and therefore also ways in which it may fail. Potential errors of collaboration are framed as breakdowns of these cognitive stage relationships. The need to move between these levels of work will also be considered, as they may be a source of error.

Failure guidewords can be applied to these relationships to establish more specific types of failure. For example:

- Cooperation:
  - ***goals(I)↔goals(I)*** - *conflict* (individuals goals conflict)
  - ***actions(I)↔plans(Group)*** - *Incorrect* (one or more individuals are performing actions that disagree with the group plans)
  - *goals(Group) - Failure* (the goals of individuals in the group do not come together to fulfil the group goals)

The failure guidewords appropriate to the cognitive stage relationships for each aspect of work depend on the requirements of these relationships for each aspect. Potential failures can be identified according to how the requirements of cognitive stage relationships may fail within each aspect of

work. This process produces a list of possible ways in which collaboration may fail.

Having identified which cognitive stage relationships are important for each aspect of work and how they are important, and then having considered how these may fail, it is necessary to examine the possible reasons *why* these may fail within each aspect of collaboration. This will allow the creation of error analysis questions. Deeper examination of the enhanced model of communication, using the framework in Table 2, allows possible causes of failure to be considered systematically. This involves taking the requirements of each aspect of work for success (in terms of cognitive stage relationships as discussed above) and then identifying the elements responsible for maintaining these cognitive stage relationships in that aspect of work (for example, the degree of awareness of certain aspects of work required to create shared understanding, or rules for work). This information will originate from different places in different forms of work. What the participants need to create these cognitive stage relationships successfully (for example, instructions or rules and environmental triggers for these in coordination, or a shared awareness of past, present, and future issues for cooperation) can be used to question design issues. The model of communication allows the systematic consideration of how these requirements may be supported, or of design reasons why they may fail. Error analysis questions can then be constructed that tackle these design issues raised using the model.

6. CONCLUSION AND FUTURE WORK

An iterative approach to designing an error analysis method for collaborative work has been taken. From an evaluation of the original approach to using a model of collaboration to create error analysis questions it was shown that the process of creating the CHLOE error analysis led to questions that did not tackle collaborative failure in enough depth. The results of the evaluation have been used to guide a new approach to creating error analysis questions. Through enhancing the model of communication, identifying a breakdown of collaboration to provide context for this model, and isolating key relationships in collaboration, an improved analysis framework has been created that can structure the description of the requirements of collaborative work. This framework will now be used as a more systematic and detailed way of understanding collaborative work and identifying possible failures within it, upon which to build an improved error analysis method.

ACKNOWLEDGEMENTS

Angela Miguel is supported by an EPSRC studentship on the DIRC project (http://www.dirc.org.uk), UK EPSRC Grant N13999. We would also like to thank our anonymous referees for their insightful comments.

REFERENCES

Annett J. & Cunningham D. (2000) Analysing Command Team Skills In Cognitive Task Analysis, Shraagen J. M., Chipman S. F., Shalin V. L. Eds. LEA.

Bardram, Jakob E. (1998): Designing for the Dynamics of Cooperative Work Activities. In Proceedings of 1998 ACM Conference on Computer Supported Cooperative Work, Seattle, Washington, USA. ACM Press.

Chudleigh, M.F, & Clare, J.N. (1993) The Benefits of SUSI: Safety Analysis of User System Interaction. In J. Gorski (Ed.) SAFECOMP'93. Proceedings of the 12th International Conference on Computer Safety, Reliability & Security, Poznan-Kiekrz, Poland. Springer Verlag.pp 123-132.

Clark, H. H. (1996) Using Language. Cambridge, UK. Cambridge University Press.

Dix, A., Finlay, J., Abowd, G. & Beale, R. (1998) Human-Computer Interaction 2nd Ed. Prentice Hall Europe, Harlow, Essex.

Gutwin, C. & Greenberg, S. (2002) A Descriptive Framework of Workspace Awareness for Real-Time Groupware Computer Supported Cooperative Work 11,411-446.

Hollnagel, E. (1998) Cognitive Reliability and Error Analysis Method (CREAM). Elsevier Science Ltd.

Hutchins, E. (1995) Cognition in the Wild. MIT Press.

Kletz, A. (1999) Hazop and Hazan: Identifying and Assessing Process Industry Hazards.4th Ed. Rugby, UK

Laurillau, Y. & Nigay, L. (2002) Clover Architecture for Groupware. In Proceedings of 2002 ACM Conference on Computer Supported Cooperative Work, New Orleans, Louisianna, USA. ACM Press. pp.236-245.

Malone, T. W. & Crowston, K. (1990) *What is Coordination Theory and How Can it Help Design Cooperative Work Systems.* Proc. of the 3rd Conf. on CSCW, ACM, p 357-370.

Miguel, A. & Wright, P. (2003) CHLOE: A Technique for Analysing Collaborative Systems. Proc. of the 9th CSAPC, G. van der Veer & J. F. Hoorn (Eds), pp53-60.

Nardi, B. A. (1996) Context and Consciousness: Activity Theory and Human-Computer Interaction. MIT Press.

Norman, D. A. (2002) The Design of Everyday Things. Doubleday, New York.

Pocock, S., Harrison, M.D., Wright, P.C. and Johnson, P.D. (2001). THEA: a technique for human error assessment early in design. In M, Hirose (Ed.) IFIP TC 13 International Conference on Human-Computer Interaction. IOS Press. Ohmsha. pp. 247-254.

Schmidt K. & Simone C. (1996) Coordination Mechanisms: Towards a Conceptual Foundation of CSCW Systems Design, Computer Supported Cooperative Work (CSCW), vol. 5, no. 2/3,1996.

Segal L. D. (1995) Designing Team Workstations: The Choreography of Teamwork In Local Applications of the Ecological Approach to Human-Machine Systems Vol. 2, Hancock P., Flach J., Caird J. & Vicente K. Eds. LEA.

INTEGRATING HUMAN FACTORS IN THE DESIGN OF SAFETY CRITICAL SYSTEMS

A barrier based approach

Bastiaan A. Schupp[1], Shamus P. Smith[1], Peter C. Wright[1], Louis H.J. Goossens[2]
[1]*University of York, Department of Computer Science, Heslington, York, YO10 5DD, United Kingdom, { bastiaan.schupp / shamus.smith / peter.wright }@cs.york.ac.uk*

[2]*Delft University of Technology, Department of Technology, Policy and Management, Safety Science Group, Jaffalaan 5 2828 BX Delft, the Netherlands, l.h.j.goossens@tbm.tudelft.nl*

Abstract: Human factors contribute to risk in safety critical systems. However, current approaches to integrating human factors issues in the development of safety critical systems appear not fully sufficient. In this paper a new approach is proposed based on a technique from chemical engineering risk analysis called Safety Modelling Language (SML). SML provides a way to conceptually design risk reduction based on barriers. The approach further helps to design and implement safety barriers. The approach is demonstrated using a case in which human factors play an important role from the medical domain.

Keywords: Human Factors, Design, Safety, Barriers, Methods, Risk, Risk Reduction

1. INTRODUCTION

Risk reduction is a key factor in the design of safety critical systems. Human factors are an essential part of the risk reduction process. When systems become operational the human may either create accidents, or help as part of the system to prevent them. The integration of human factors analysis into systems design is traditionally a difficult problem (Hollnagel 1993). Designers need to be able to explore and evaluate solutions. In this paper an approach is presented that allows the design of risk reduction and safety barriers and supports barrier implementation.

Traditionally risk reduction is dealt with late in the system design process when all details are clear, and economic and safety benefits can be gained from designing risk reduction from the design onset (Arthur D. Little 2001, Schupp et al. 2002). Usually, at an early stage, multiple options exist to achieve the most optimal design. However if risk reduction is considered too late in the design process options are lost as changing earlier design decisions becomes prohibitively expensive. Hence, designers must have the ability to identify potential problems, to estimate the risk, and to find and evaluate solutions early in design.

The format for the remainder of this paper is as follows: Section 2 outlines our approach and briefly describes its main components, (i) the barrier concept, (ii) Safety Modelling Language (SML) and (iii) the mapping between SML and a design. Section 3 presents the case study of this paper. It is a case from the medical domain that deals with human factor issues in the design of a computer assisted detection system for mammography. Section 4 presents concluding remarks.

2. INTEGRATING HUMAN FACTORS IN DESIGN

When a system becomes operational its properties may create adverse effects, i.e. hazards, to the system itself, or to its environment. A design team will use tools to identify such potential hazards. If humans participate in the system, the designers may resort to well established human factors methods such as human reliability analysis (HRA), THERP (Kirwan 1994), or HEART (Williams 1986), or to human factor experts.

After identifying issues that lead to unacceptable risks, the designers may either decide to change the design to prevent hazards or to add barriers that protect the target. Barriers are designed in a similar manner as the initial system. A barrier may also have adverse effects and when it fails, it will no longer prevent or protect against the hazard it was designed for. Hence the designers of the barrier analyse it in a similar manner as the initial system. This process is complex, as the barrier may be implemented across multiple system components.

One of the obstacles to integration of human factors analysis into conceptual system design is the lack of a suitably expressive language to represent and analyse safety at a conceptual level. Most existing human factor methods are for identification or quantification, and do not help in finding solutions. Similar observations are made by Swuste (Swuste 1996) and Harms-Ringdahl (Harms-Ringdahl 2003). Hence, our method is solution oriented and provides a framework for designers to explore and evaluate solutions. This is based on the previously developed SML (Schupp et al.

2001), which aids designers in conceptually designing risk reduction. It was developed for use in the chemical process domain, but we will show that it can be used in other domains as well, and how to integrate human factors into it.

The proposed method deals with the design of two systems: The safety critical system itself, and the barrier systems that mitigate its risks. As barriers are important in risk reduction these are the main building block of our method. The explicit but conceptual representation of barriers which we describe in this paper translates the results of human factors methods into a design representation to create an overview of how barriers will become implemented and how risk reduction is achieved. Usually it is not the original designers of the safety critical part of the system that are involved in designing, implementing, maintaining and operating barriers, but other actors. Hence, by creating overview, the method also helps to disseminate information to these actors.

2.1 Barriers

The word barrier is commonly used in normal language as well as in risk management and human factors domains. However, A well accepted 'barrier theory' does not exist. Nevertheless it has been shown that barriers can be a viable way to study human factors (Kecklund et al. 1996), to analyze systems (Johnson 1980), and there exists attempts to classify barriers by their physical implementation e.g. (American Institute of Chemical Engineers 1993) and (Hollnagel 1999). Sometimes barriers are named differently, for instance layers of protection (Dowell 1998). Our own notion of barriers is based on Haddon's (Haddon 1973) fundamental strategies for risk reduction which culminates in the hazard barrier target model (described in Section 2.2).

Here barriers are *always* considered as systems as they almost always have multiple components. We define barriers as the combination of technical, human and organisational measures that prevent or protect against an adverse effect. A typical barrier may have three components, one to detect, one to decide, and one to deflect. An example of a barrier is a non-smoking sign. However the barrier is not the sign as such; it includes awareness of how smoking may cause fire, awareness of the significance of the sign, its state and location, its maintenance, training of the smokers, and its relation to other barrier systems. If a non-smoking sign is put in the wrong place, it will not work. Hence the location is part of the barrier. Similarly, a brake is not a barrier but the actuator in a braking system also involving detection and decision. Its function is not to brake, but is part of a system by which, for instance, collisions are avoided. Here our approach

differs from some other sources in literature where the sign or the brake are considered to be the barrier, for instance Hollnagel (Hollnagel 1999).

Humans interact closely with barriers. Obviously barriers may protect humans, but humans can be part of barriers, can make them fail, and must maintain them.

Partly based on a classification made by Swuste (Swuste 1996), we study barriers here at three levels:

- At the *safety function* level; this concerns the role of the barrier in system safety. For example, the role of the non-smoking barrier in the greater context of preventing fire in a building. At this level barriers are considered black boxes, i.e. the internal structure and functions are ignored.
- At the *barrier form* level; this concerns how a barrier functions and what its components are, thus what is inside the black box introduced at the safety function level. For example, the components introduced when discussing the non-smoking sign; the sign, training, and maintenance. This level demonstrates which functions the system should provide to allow the barrier to function.
- At the *embodiment* level; this concerns the detailed design of the barrier, and its physical representation and implementation in the safety critical system. For example, the requirement that a non-smoking sign with specified size should be placed at a specified position on every access door.

These three levels are the core of our method. We use SML at the safety function level to design an optimal risk reduction strategy, based on information from the form level and the embodiment level. At the form level the basic design and analysis of barriers takes place, while the whole of the safety critical system and the implementation of the barriers therein are addressed at the embodiment level.

2.2 The Safety Modelling Language

SML provides a means to design and document the function of barriers in a system. It provides a *framework* that allows a designer to define the problem, to analyze knowledge for solving the problem, to synthesize possible solutions, and to analyze the performance of these solutions. Furthermore, it helps the designer to communicate. A very important use of SML not discussed in this paper is that its relational structure offers a efficient means for storing information, for example for storing experience about barrier performance, for later reuse. The language can be used during all life-cycle stages of a system, thus including all design stages, implementation and operation.

SML is based on the Hazard-Barrier-Target (H-B-T) model (Schupp et al. 2001), which assumes that targets are vulnerable to the effects of hazards. In some respects it is similar to other barrier models, such as the accident evolution and barrier function model (Svenson 1991), and the 'Swiss-cheese' model (Reason 1990). However, its main focus is on design and communication, not on analysis. The main means of communication in the SML are diagrams, as in Figure 1. This diagram shows how toxic fumes are hazardous to workers, as these poison them. However the worker is protected by a containment system that contains the fumes, thus being a barrier that prevents exposure. As this may not be completely adequate, the worker is further protected by Personal Protective Equipment (PPE). Alternatively prevention is realized by removing the hazard, for example by using a non-toxic substance.

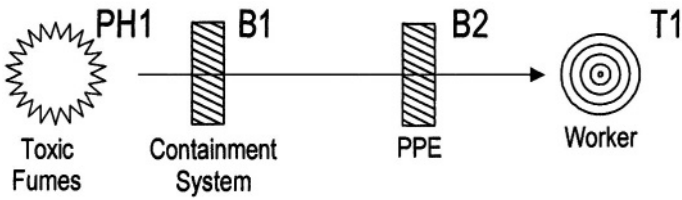

Figure 1. A typical H-B-T diagram. Toxic fumes are hazardous to workers. Hence the workers are protected by a containment system and Personal Protective Equipment (PPE)

The use of SML is relational. A relation between the Hazard and the Target is called an H-T relation. Thus, toxic fumes and workers are an example of an H-T relation. This H-T relation is itself related to a number of barriers that can act to reduce risk. In Figure 1 these barriers are the containment system and PPE, but may possibly also include barriers such as ventilation, scrubbing, or access procedures. Designers can use the H-T relation to find the optimal barriers for use in their system.

SML models hazards in a more complex manner than the basic H-B-T model in Figure 1. A hazard is something that has the potential to cause an adverse effect to a target. A hazard is a 'label' that humans apply to complex phenomena perceived as hazardous. It is modelled using two components: Causal elements that provide a link to the mechanism of the hazard, and effects, that provide the link to the targets. For instance, when the elements 'flammable substance', 'oxygen', and 'ignition source' are present in a design, these will cause a fire hazard, having heat radiation, smoke and high temperature as effects. This is shown in Figure 2a. A human factors related hazard is a misdiagnosis in interpreting an X-ray photograph in a medical domain. This can for instance be caused by the causal elements 'training',

'available time', and issues such as 'X-ray clarity'. Effects of a misdiagnosis for instance are false positive or false negative readings (see Figure 2b).

SML does not model the underlying mechanisms of the hazardous phenomena; the modelled hazard is not a direct representation of the mechanism (e.g. physics, or psychology), but it is linked to it by the causal factors. Thus SML does not provide insight into the hazardous phenomenon itself but into the relations this phenomenon has with the rest of the design/system.

Two further aspects of barriers which are relevant to design, but are only briefly discussed in this paper are (i) the types of barriers that can be used, and (ii) modelling how barriers can fail.

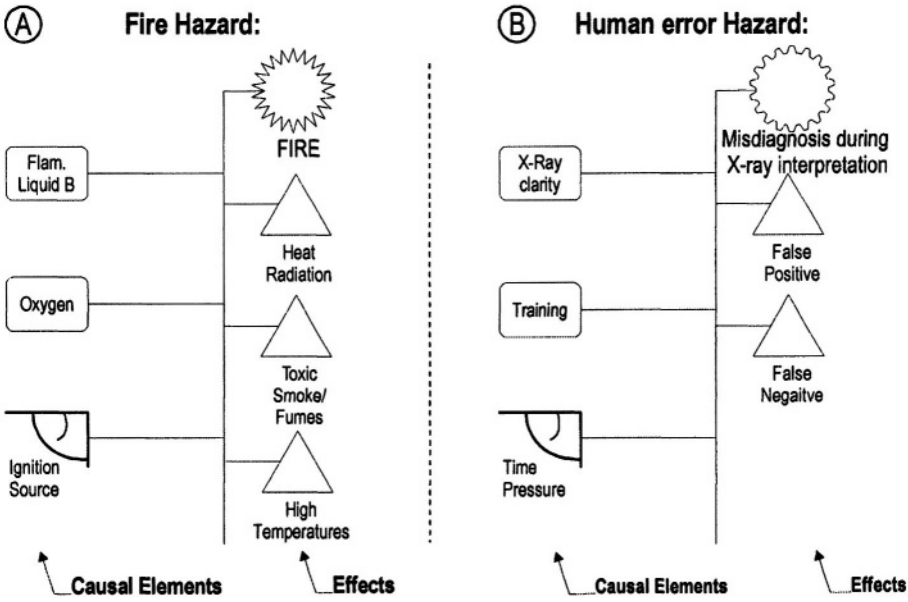

Figure 2. Diagram showing the models of a fire hazard (a), and of a human error hazard (b). The causal elements map to components of the system, while the effects map to targets. The different symbols used for the hazards designate a primary (a) and functional (b) hazard

The types of barriers that are used in SML are classified using three binary dimensions; protective versus mitigative, inherent versus add-on, and preventive versus defensive. This is based on their role in the design, rather then on the nature of the barrier. Thus classifying supports designers in making the role of the barrier clear, without having to worry about the exact form or embodiment of the barriers. The same barrier (e.g. fire protection) may have a different classification in different implementations.

To model the failure of barriers, SML defines primary and functional hazards, the symbols are shown in figure 2. Primary hazards cause direct

harm to humans, neighbouring installations, and the environment. The barriers in between primary hazards and primary targets are called primary barriers. Functional hazards are phenomena due to either human factors or other causes that adversely affect other barriers, thus making these fail. In this way, a risk reduction problem is defined recursively; when a barrier is used, it can fail due to a functional hazard. It can however be protected by defining functional barriers that protect against this hazard, which can fail as well.

A consequence of this is that the list of primary hazards quickly provides insight in why the systems' safety is critical. Next, accident mechanisms, and the role humans play in these can be understood via recursions. This however is not further explained in this paper.

2.3 Mapping SML to Form and Embodiment

To be useful, the SML must be mapped to the form and embodiment level. A detailed discussion of this cannot be provided here. In brief, safety is designed using the SML. Then each barrier is conceptually designed at form level. Subsequently, it is implemented in the design of the overall system at embodiment level. In this paper we represent the conceptual design of the barrier at the form and embodiment level, and of the system using a simplified version of the SADT[31] approach. This approach allows hierarchically representing multiple system levels, which is a useful way to study the implementation of barriers. How this occurs is demonstrated in the next section.

3. MAMMOGRAPHY CASE STUDY

This section describes a case study originating from the medical domain that illustrates our approach. It involves a socio-technical system in which humans play both a central role as a safety barrier and also are a source of error and potential hazard for the system. The case studies centres on a classic human-factors design decision, of how to combine a level of automation with a level of human intervention to minimise the risk of a system failure.

[31] SADT is Structured Analysis and Design Technique, more recently IDEF (not an acronym) was developed, an elaboration on this. For details see www.idef.com/default.html.

3.1 The Domain

The UK Breast Screening Program is a national service that involves a number of screening clinics, each with two or more radiologists. Initial screening tests are by mammography, where one or more X-ray films (mammograms) are taken by a radiographer. Each mammogram is then examined for evidence of abnormality an experienced radiologist (Williams et al. 1998). A decision is then made on whether to recall a patient for further tests because there is suspicion of cancer (Alberdi et al. 2003). Within the screening process it is desirable to achieve the minimum number of false positives (FPs), so that fewer women are recalled for further tests unnecessarily, and the maximum true positive (TP) rate, so that few cancers will be missed (Williams et al. 1998). Unfortunately the radiologists' task is a difficult one because the small number of cancers hidden among a large number of normal cases. The traditional solution for reducing reading errors is to let a peer do double readings. This obviously increases workload. Another solution that is being explored is the use of computer-based image analysis techniques to enable a single radiologist to achieve performance that is equivalent or similar to that achieved by double readings (Williams et al. 1998, Boggis et al. 2000). Computer-aided detection systems can provide radiologists with a useful 'second opinion' (Zheng et al. 2002). The case study in this section involves the introduction of such a system as an aid in screening mammograms. When it is used, the radiologist initially views the mammogram and records a recall decision. Then the system marks a digitised version of the X-ray film with 'prompts' that the radiologist should examine. A final decision on a patient's recall is then taken by the human radiologist based on the original decision and the examination of the marked-up X-ray.

3.2 Analysis

In this domain the breast cancer screening programme is in itself a barrier. It mitigates the chance that breast cancer that is difficult to treat develops in women. This barrier could be the starting point for our approach. For the sake of brevity however, our analysis starts at the screening procedure, that is a part of the breast cancer screening programme. It is defined as level 0, shown as a simplified SADT diagram in Figure 3.

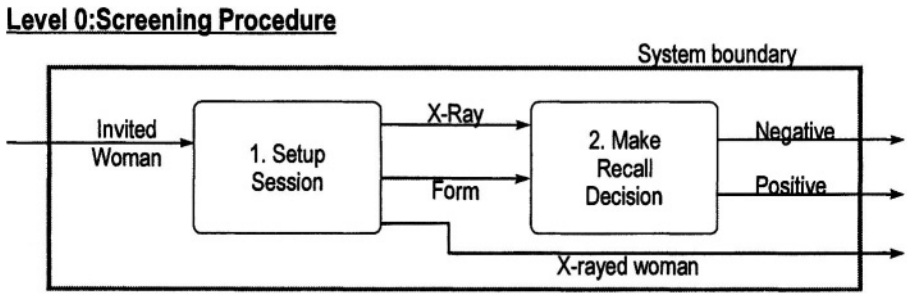

Figure 3. Design of the screening procedure, the top level in this analysis

The screening procedure is a two stage activity. First the session is set-up, then the result of this session (an X-Ray and a form with data). In this case, the hazards are identified using a HAZOP (Kletz 1999) procedure, which is not explained in this paper (see Smith and Harrison 2003). Other identification techniques such as forms of task analysis may be used additionally. The results of applying this HAZOP are shown in Table 1.

The next step is to convert the results of the HAZOP into hazards. Further analysis may be required to model the hazards more precisely in terms of causal elements. This can occur for instance by reusing hazard models identified in other systems, or by using expert knowledge.

In this case we further analyse ref. 1.b and 2.b in table 1. Ref. 2.b is used to illustrate how our approach is used in facilitating decision making, while ref 1.b demonstrates how barriers are implemented via the form and embodiment level in different parts of a system.

Table 1. Hazop applied at level 0. Ref. 1.b and 2.b are used in the examples in the text

Ref	*Item*	*Guideword*	*Cause*	*Consequence*
1.a	Setup Session	More	Too many invited women	Overload of treatment system
1.b	**Setup Session**	**Wrong**	**X-rays and screening forms not matched**	**Decision will be faulty**
1.c	Setup Session	Late	Not ready when radiologist ready for viewing	Woman will be waiting too long; Performance problems
2.a	Make recall decision	Omit	Radiologist error	No decision, woman will be waiting too long
2.b	**Make Recall Decision**	**Wrong**	**Something went wrong in decision; at this level we cannot determine what.**	**Possible False Positive or False Negative**
2.c	Make Recall Decision	Late	Slow processing; at this level we cannot determine why.	Woman will be waiting too long; Performance problems
2.d	Make Recall	More	Something went wrong in decision; at this level we	Confusion in down stream activities; possible

Decision	cannot determine what.	performance problems

3.3 Barrier based design decisions

The cause consequence pair indicated by ref 2.b in Table 1 identifies a hazard with two effects: false positives and false negatives (compare Figure 2). It is a primary hazard, as one of these effects directly affects the woman. A false positive requires recalling the woman. The risk here is clearly unacceptable, as a false recall decision has adverse consequences to women, for example stress, and may occur frequently. Hence, a barrier must be designed. This occurs at level 1, which models the recall decision. In principle, the only activity at this level is to make the decision. The barrier is an additional element. The corresponding IDEF diagram is shown in Figure 4, the decision being element 1.1, the barrier 1.2.

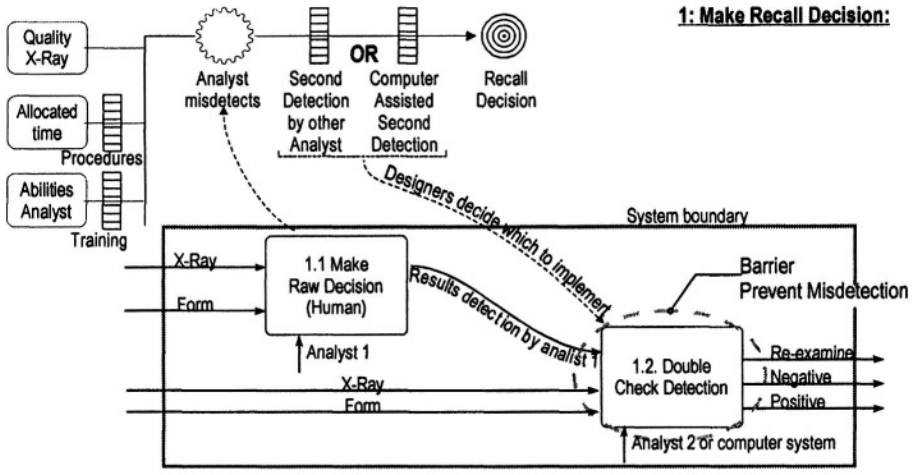

Figure 4. An analyst may make an error when detecting. Hence, system designers must decide on which barrier to use, a second reading by another analyst, or computer assisted prompting

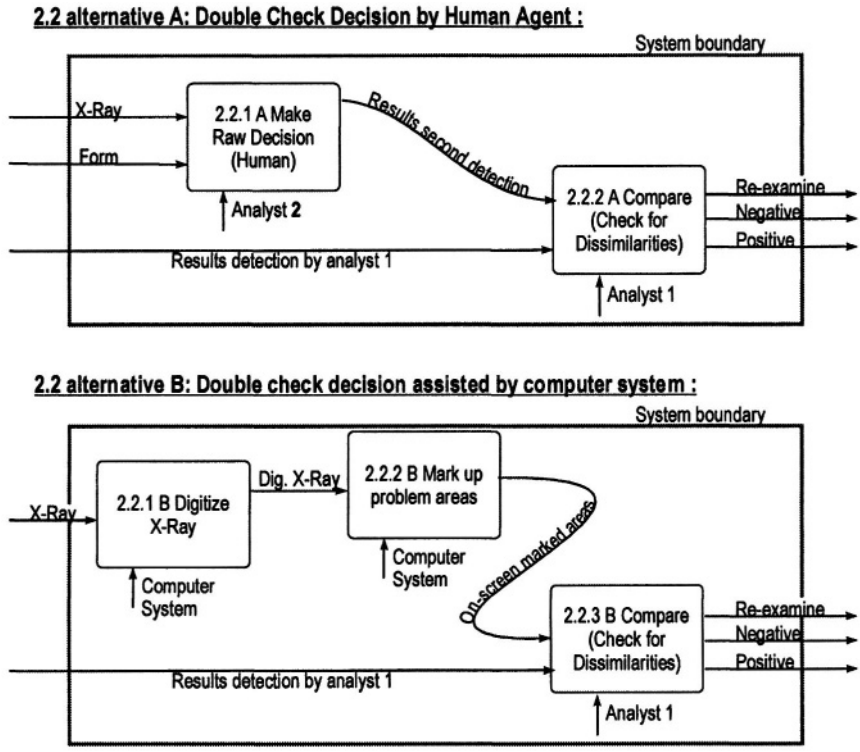

Figure 5. Form level design of Double check barrier: Alternative A: using a human agent; alternative B using the computer aided detection technique

Two alternative solutions for a barrier that defend against misdetection by the analyst are conceivable: (a) A double check of the recall decision by a human agent, and (b) a recall decision assisted by the novel computer system. The first solution has two activities. A *second* analyst repeats the detection of the *first* analyst, thus providing a second opinion. The first analyst then compares this to his own, making use of notes on problem areas provided by the second analyst. The second solution functions differently. After the analyst completed an initial marking, a computer system prompts detected problem areas. The analyst then compares these to the original marking. Both alternative solutions are shown in Figure 5.

The solutions can be analysed in terms of functional hazards. Typical design questions may be which functional hazards may disable each alternative and which new functional and primary hazards are created by these each option. Many of these hazards are caused by human factors, and can be found with traditional human factor analysis methodologies such as task analysis, a process which is not further discussed here. The result of this analysis is shown in Table 2.

Table 2. Hazards created by the two alternative barriers for double checking the raw decision

Alternative A: Human agent		*Alternative B: Computer System*	
Functional Hazards	***New* hazards to *other* systems**	**Functional Hazards**	***New* hazards to *other* systems**
FH A.1: Analyst 2 makes same mistake as analyst 1 (e.g. because relatively few positives occur)	FH A.4: Analyst 1 overly relies on analyst 2. (the B/C screening decision is directly affected by this)	FH B.1: Misdetection because of poor software	FH B.4: Analyst overly relies on computers system. (the B/C screening decision is directly affected by this)
FH A.2: Analyst 1 ignores second detection	FH A.5: Human resources used up	FH B.2: Analyst ignores computer system	
FH A.3: Communication Error		FH B.3: Analyst oversees dissimilarities in mark-up.	

Based on this analysis, the system designers will now decide which alternative to use. The key to this decision is determining which hazards can best be prevented, protected against, or mitigated. Thus identifying which option yields the lowest risk, when barrier effectiveness, costs and tradeoffs are taken into account. In this case it is difficult to mitigate FH A.1 (see Table 2); the analysts carry out the same activity, and thus may produce the same mistake. Also, alternative A can cause performance problems as it uses more human resources. Alternative B however may also create a hazard that is difficult to overcome, FH B.1, poor software. If software development, testing and training is adequate, alternative B is favourable, as the other functional hazards are similar to those associated with alternative A, or easier to overcome.

To carry out this barrier analysis, SML diagrams are drawn, as shown for alternative A in Figure 6. This displays the H-T relations. Now, barriers can be added, such as adding a procedure to prevent the first analyst from ignores the second analysts' detection, as shown. It might appear much more difficult to find a barrier between other H-T pairs however, such as creating a barrier between FH A.1 and T2.2.

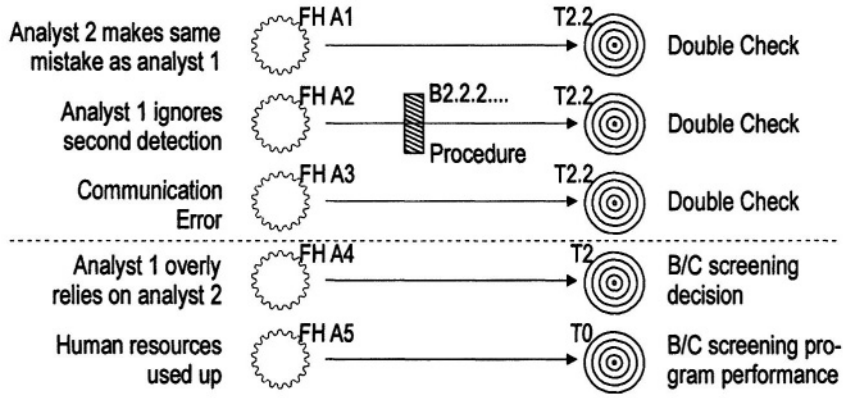

Figure 6. Hazard-Target relations relating to alternative A with one potential barrier shown

3.4 Designing Barrier Implementation

Our example now continues with ref 1.b in Table 1, which is used to demonstrate the implementation of a barrier. The hazard exists that either the X-ray or the form are accidentally swapped with that of another woman. This may be due to a human error, for instance a mistake, or it may have another technical cause. The hazard in Figure 7 is due to the simultaneous presence of multiple X-rays photographs, forms and the subsequent processing.

The effect of jumbling forms and X-rays affects the recall decision, but its cause originates earlier in the process, during the initial session. The designers of that session might be aware of this but can only solve it within that context. By analysing the problem using the SML representation instead of focussing at specific parts of the design, at embodiment level, the designers will become better able to avoid an ad-hoc solution. With this overview of the system as a whole they can now decide to use the application of barcodes to both the form and the X-ray and a subsequent check by the computer system which does the mark-up as the barrier.

At form level this barrier is comprised of two parts, (i) the procedure in which the barcodes are applied, and (ii) the subsequent check. These parts determine how the barrier must be embodied in the system. It is not embodied as an isolated part, but in multiple system components; for instance a unique ID is assigned to each women during invitation, applying the barcodes during the X-ray session, and carrying out the check during the computerized mark-up. The barrier is implemented in two ways: as new activities and as controls on existing activities.

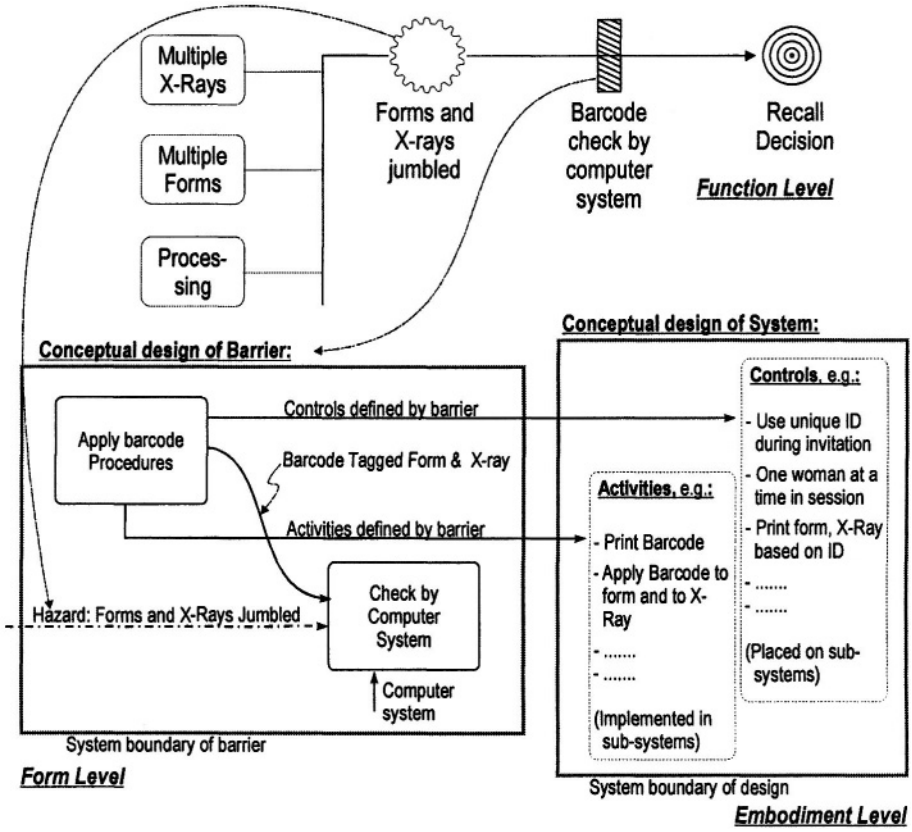

Figure 7. A barrier that defends against jumbling of forms and X-rays. It is represented differently at three different levels: The function, form and embodiment level

Another interesting aspect of this approach is that different barriers may share components at the embodiment level. The computer system was introduced as a barrier to provide the second opinion, however now parts of it have also become part of the new barrier that avoids jumbling of the forms. If a system is subsequently changed, persistent mapping to SML thus prevents inadvertent elimination of a barrier.

4. CONCLUDING REMARKS

In this paper a new approach was outlined. The SML helps to structure risk reduction, thus facilitating its design. The conceptual design of barriers and of their embodiment further contributes to this. It was shown how the design of the human role as part of the system is facilitated as well, by not

directly applying human factors methods to the design of the system, but to the conceptual design of safety and barriers.

Our research will continue on the relation between humans and barriers, and on further extending the vocabulary of the SML and its mappings to systems. One of the disadvantages of the method is that the diagrams used sometimes become quite complex. Therefore, a computer based tool is being developed that helps to manage this information. Also, such a tool will allow for ready reuse and dissemination of diagrams by storing them in a database. The approach can then be used to for instance to measure failure frequencies or operating costs of barriers to facilitate future decision making in design.

ACKNOWLEDGEMENTS

The collaboration between Delft and York on this subject would not have been possible without funding by the European Union as part of the Advises research training network, GR/N 006R02527. This work was also partly supported by the UK EPSRC DIRC project, GR/N13999. The authors also wish to thank Michael Harrison for his helpful comments.

REFERENCES

Hollnagel, E. (1993) *Human Reliability Analysis : Context and Control.* Computers and People Series. London; San Diego, CA: Academic Press. xxvi, 326 p.p.

Arthur D. Little Inc. (2001), American Institute of Chemical Engineers. Center for Waste Reduction Technologies, and American Institute of Chemical Engineers. Center for Chemical Process Safety, *Making Ehs an Integral Part of Process Design.* New York: CWRT CCPS, American Institute of Chemical Engineers. xvi, 164 p.p.

Schupp, B.A., S.M. Lemkowitz, L.H.J. Goossens, A.R. Hale, and H.J. Pasman. (2002) *Modeling Safety in a Distributed Technology Management Environment for More Cost-Effective Conceptual Design of Chemical Process Plants.* In Computer-Aided Chemical Engineering; European Symposium on Computer Aided Process Engineering - 12. ELSEVIER SCIENCE BV: p. 337-42.

Kirwan, B. (1994) *A Guide to Practical Human Reliability Assessment.* Bristol, PA: Taylor & Francis. p.p. 592.

Williams, J. (1986) *Heart - a Proposed Method for Assessing and Reducing Human Error,* 9th Advances in Reliability Technology Symposium.University of Bradford.

Swuste, P. (1996) *Occupational Hazards, Risks and Solutions,* thesis, Delft University of technology, 217 p.

Harms-Ringdahl, L. (2003), *Assessing Safety Functions - Results from a Case Study at an Industrial Workplace.* Safety Science, **41**(8): p. 701-20.

Schupp, B.A., S.M.L. Lemkowitz, and H.J. Pasman. (2001) *Application of the Hazard-Barrier-Target (Hbt) Model for More Effective Design for Safety in a Computer-Based Technology Management Environment;* CCPS ICW: Making Process Safety Pay: the business case, AIChE/CCPS.

Kecklund, L.J., A. Edland, P. Wedin, and O. Svenson, (1996) *Safety Barrier Function Analysis in a Process Industry: A Nuclear Power Application.* International Journal of Industrial Ergonomics. **17**(3): p. 275-84.

Johnson, W.G. (1980) *Mart Safety Assurance Systems.* New York: Marcel Dekkerp.

American Institute of Chemical Engineers. (1993) Center for Chemical Process Safety., *Guidelines for Safe Automation of Chemical Processes.* New York, N.Y.: Center for Chemical Process Safety of the American Institute of Chemical Engineers. xxiv, 424p.

Hollnagel, E. (1999) *Accidents and Barriers*; Lez Valenciennes. Presses Universitaires de Valenciennes: p. 175-82.

Dowell, A.M. (1998) *Layer of Protection Analysis for Determining Safety Integrity Level.* Isa Transactions,. **37**(3): p. 155-65.

Haddon jr., W. (1973) *Energy Damage and the Ten Countermeasure Strategies.* Human Factors, **15**(4): p. 355-66.

Svenson, O. (1991) *The Accident Evolution and Barrier Function (Aeb) Model Applied to Incident Analysis in the Processing Industries.* Risk Analysis. **11**(3): p. 499-507.

Reason, J.T. (1990) *Human Error.* Cambridge: Cambridge University Press. 302p.

Williams, L.J., M. Hartswood, and R.J. Prescott (1998) *Methodological Issues in Mammography Double Reading Studies.* Journal of Medical Screening, **5**(4): p. 202-6.

Alberdi, E., A. Povyakalo, L. Strigini, and P. Ayton (2003) *Does Incorrect Computer Prompting Affect Human Decision Making? A Case Study in Mammography.* In International Congress Series; Proceedings Cars 2003: Computer Assisted Radiology and Surgery. ELSEVIER SCIENCE BV, Amsterdam: p. 938-43.

Boggis, C.R. and S.M. Astley (2000) *Computer-Assisted Mammographic Imaging.* Breast Cancer Research. **2**(6): p. 392-5.

Zheng, B., R. Shah, L. Wallace, C. Hakim, M.A. Ganott, and D. Gur (2002), *Computer-Aided Detection in Mammography: An Assessment of Performance on Current and Prior Images.* Academic Radiology. **9**(11): p. 1245-50.

Kletz, T.A. (1999) *Hazop and Hazan Identifying and Assessing Process Industry Hazards.* 4th ed. Philadelphia, PA: Taylor & Francis, xi, 232 p.p.

Smith, S.P. and M.D. Harrison (2003). *Reuse in Hazard Analysis: Identification and Support.* In Computer Safety, Reliability, and Security, LNCS 2788; SAFECOMP. Springer: p. 382-95.

DESIGNING DISTRIBUTED TASK PERFORMANCE IN SAFETY-CRITICAL SYSTEMS EQUIPPED WITH MOBILE DEVICES

Ana-Maria-Marhan, Fabio Paternò, Carmen Santoro
ISTI-CNR, Via G.Moruzzi 1, 56124 Pisa, Italy

Abstract: This paper describes a method aiming to support the design of interactive-safety critical systems. The method proposes an original integration of approaches usually considered separately, such as task modelling and distributed cognition. The basic idea is that analysing task performance requires a clear understanding of the information needed to accomplish the task and how to derive such information from both internal cognitive representations and external representations provided by various types of artefacts. We also report on a first application of the method to a case study in the Air Traffic Control (ATC) domain.

Key words: Human-Computer Interaction, Methodologies, Design, Task Models.

1. INTRODUCTION

Recent developments in wireless communication, distributed systems, together with increases in the power and interactive capabilities of handheld and portable devices provide users with the possibility of a wide-ranging, continuous access to computing resources in a variety of contexts. One of the controversial issues we intend to address in this paper is whether, and in which way, new mobile technology can offer meaningful support to accomplish critical tasks while preserving safety and usability.

To this end, we have extended the analysis of deviations developed in our previous work (Paternò and Santoro, 2002), and provided more explicit consideration of the distributed cognitive resources supporting task performance. The idea guiding this work is that a criticality, or breakdown in

task performance is the consequence of an *inadequate access to information* supporting task accomplishment. By integrating a task-based approach to design and evaluation of interactive systems with a distributed cognition analysis (Hutchins, 1995), we aim to achieve a twofold objective:

- To identify a design method that systematically analyses task accomplishment, detects potential deviations, and provides design criteria grounded on distributed cognition analysis;
- To support designers in analysing the impact of introducing mobile, wireless devices in existing technological contexts, and its potential implications in terms of user support.

In the following sections, we first discuss related work, then we move on to describe the proposed method; further on we will introduce a case study in the area of ATC, and show how the proposed method can be applied considering a specific situation.

2. RELATED WORK

While the theoretical framework of distributed cognition seeks to offer a systemic attention on cognition (as embedded in its environmental setting), the main limitation of distributed cognition-based approaches is that it seems to fail to provide designers with systematic support in order to translate some interesting findings and general principles in specific design criteria. Hence, in this paper we attempt to integrate a systematic task-based approach to design and evaluation with criteria derived from a distribute cognition analysis.

Task-based approaches have long been considered in system design and evaluation. Recently, automatic tool support became available. For instance, CTTE environment provides an integrated set of modelling tools (Mori, Paternò, and Santoro, 2002) that allows designers to analyse particular sequence of tasks and to compare alternative paths available (i.e., when the tasks are carried on in different contexts, or using different computing platforms) in order to attain a specific goal.

Such features are particularly relevant when Air Traffic Control (ATC) domain is considered. ATC is an application with well defined goals (ensuring safe and efficient air traffic flow), while a various flexible ways are available to achieving such goals. Complex flows of information can be identified within the interaction among a number of actors (controllers, pilots, aerodrome technicians, etc.), and their technical contexts (cockpit, ATC control centres, aprons, etc.), co-operating for the common purpose of ensuring air traffic safety and efficiency.

Although the flexibility introduced by mobile devices can find interesting applications in this area, little research work has been addressing the problem of mobile devices in safety-critical contexts. Especially PDAs seems to capture both researchers' and industry's interest: Mertz et al (2000), Buisson & Jestin (2001) present some scenarios that use PDAs for navigating in the ATC information space. A prototype developed at CENA on the basis of DigiStrips (http://www.tls.cena.fr/divisions/PII/digistrips/), addresses the problem of electronic stripping using a PDA platform. However, the presentation of information using a PDA raises some interesting issues, primarily related to the limitations in displaying information on small-screen devices (particularly when it is question of relatively complex information like procedures or rules which, in ATC, can include maps and long structured text).

3. A METHOD FOR DESIGN AND EVALUATION OF SAFETY-CRITICAL SYSTEMS SUPPORTED BY MOBILE DEVICES

One starting point for this research has been the deviation analysis (Paternò and Santoro, 2002). This method involves three main steps:

- *Development of the task model of the application considered;* the purpose is to provide a description logically structured in a hierarchical manner of tasks that have to be performed, including their temporal relationships, objects manipulated and tasks' attributes.
- *Analysis of deviations related to the basic tasks,* which are tasks that the designer deems should be considered as units.
- *Analysis of deviations in high-level tasks,* these tasks allow designer to identify *group of tasks* and consequently analyse deviations that involve more than one basic task (e.g. deviations concerning whether the appropriate tasks are accomplished following a correct ordering).

In this method, the ConcurTaskTrees (CTT) notation has been used (Paterno', 2003) for task modelling. CTT task models are structured in a hierarchical manner that allows an analysis at various levels of abstractions, and support a detailed analysis of tasks, temporal relationships between tasks, and objects manipulated by tasks. CTT objects are classified in *perceivable* (objects represented at the interface level, that can be directly perceived by the user), *cognitive* (User internal representations of the informational entities) and *domain* objects (entities that are internally represented and manipulated by the system). Tasks are classified depending on their performance allocation: *user tasks* require only an internal

performance, *interactive tasks* require interaction between the user and the external world, *and system tasks* are completely automatic tasks.

While analysis of deviations based on CTT task models has proved to be useful in generating some interesting results, we realised that its effectiveness can be increased in order to identify possible breakdowns in task performance.

3.1 Integrating CTT task modelling and DC analysis

This approach is based on the idea that a criticality in user - system interaction is the consequence of an inadequate access to the information distributed among the interacting components. Based on the description of the (idealised or real) *plan of tasks* the user needs to follow in order to achieve a goal, the analyst will identify the *possible configurations* of the resources supporting user actions while carrying out such plans. The method we propose allows a pro-active estimation of those aspects that might affect safety following the introduction of new artefacts in the work setting, and the identification of alternative design options.

The basic steps considered are:

- Identifying tasks (the task considered could be basic tasks in the CTT model, or higher level tasks);
- Determining the resources required to perform the task;
- Defining the most dangerous potential safety-critical deviations for each task;
- Based on the questions stimulated by a distributed cognition approach, identify and evaluate new design solutions that imply less safety-critical interactions
- A set of heuristics extracted from the Distributed Cognition literature orients the analyst's exploring the distributed resources represented within the task space, identifying potential breakdowns or failures can occur owing to a specific distribution of resources across the interactive system, and allows him to systematically reason about alternative ways of distribution of resources during task performance and their impact.

3.2 Analysis steps

In carrying on the analysis, several aspects need to be carefully identified : role, task, representations, and deviations.

1. *Analysis of the task and related properties:* this phase has to identify the goal of the task and a number of task properties that could be relevant for analysing whether its current design is appropriate or possible improvements in terms of system safety and usability are needed:

1.a Task category and type: depending on task category , different types of actions are meaningful (e.g. for an interactive task, possible actions are: monitoring, control, edit, etc.)

1.b Task platforms: it is possible to specify the platforms embedding the resources supporting task performance.

1.c Task frequency: the amount of time the user devotes to the considered activity; it might indicate the overall 'criticality' or 'importance' of the task.

1.d Other properties (that could affect task 'criticality'): need of real time performance/responses, task urgency, strong dependency with other tasks.

2. *Analysis of the representations associated with the task* – For each task, the supporting representations (and related properties) are identified:

2.a Availability: it evaluates the type of "presence" of a certain representation in the user's context, differentiating between *external* or *internal* (i.e., represented at the user's level) representations. Depending on the value of this property, it could be meaningful to evaluate other properties such as:

2.b. Accessibility: type of access user has to a representation. Different types of accessibility (sequential/concurrent) could be exploited for externally available representations. Also, the analysis of *visibility* might be connected with the type of media/device used.

2.c. Mobility: whether the access to the information requires the user to move or the user can move while accessing information.

2.d Observability: refers to the extent the perception of a representation is: i) *Local* to individuals; ii) *Shared* (e.g. by the members of a team); iii) *Globally available* to all. This property might be connected with the type of supporting platform (for example if controllers annotate a strip on their PDAs, this information will be available locally to them).

2.e Persistence: whether transient or permanent access to information is allowed.

2.f Flexibility of modifying the representation, ability to flexibly update and modify the representation, for example allowing a person to annotate an external representation (i.e. strips).

2.g Operations and actions supported

2.g.1 Comparability: with other objects / representations available in the user's context;

2.g.2 Combinability: allowing users to combine information from different sources;

2.g.3 Ease of production: allowing reconfiguring and multiple views of information;

3. *Analysing deviations from the task plan* – In this phase a combined analysis and evaluation of the information gathered in the previous steps has to be carried out. Such analysis has to evaluate if the current configuration of representations (with their own specific properties) is effective for the task considered (and in the way this task is supposed to be carried out in the considered system). For example, if a certain task results to be a frequent *monitoring* task and in the considered system the representation of the current situation cannot easily be compared with the expected situation (because, e.g. the expected situation is an internal representation or it is an external representation but it does not reflect the controller's mental model), this could put a heavy (and risky) workload on the controller who continuously has to accommodate this information before actually using it. More in detail, our evaluation will be driven by means of a number of guidewords that will be mainly focused on possible 'deviations' occurring on representations with regard to the performance of the considered task, namely:

   3.a None: the representation supporting task performance is either not available in the task space, either not visible, or not observable (or a combination of them)

   3.b Other than:

      3.b.1 Less information than required is provided by the considered representation.

      3.b.2 More information than required is provided by the considered representation.

      3.b.3 Different information than required may be available.

   3.c Wrong timing: the resources required is available, but either

      3.c.1 Later than required, or

      3.c.2 Earlier than required

The results of such analysis may be stored in a table with the following information:

- *Task:* the activity currently analysed, together with some properties relevant to our analysis;
- *Representation distribution:* the resources supporting task performance and their distribution;
- *Guideword:* the type of interaction failure considered;
- *Explanation:* how an interaction failure has been interpreted for that task;
- *Causes:* the potential causes for the interaction failure considered and which configuration of resources might have generated the problem;
- *Consequences:* the possible effects of the occurrence of the interaction failure in the system;

- *Recommendation:* suggestions for an alternative (if any) distribution of resources, able to better cope with the considered interaction failure.

Thus, our analysis aims at identifying better representations (or distributions of them) that could be more suitable for carrying out the considered tasks. Not only the evaluation has to consider if a different allocation of resources may be envisaged but also if different representations of information -which could involve considering different devices- may result in a significant improvement for the overall system's safety and usability.

4. THE CASE STUDY

In order to show potential application of our method we consider a case study related to a real setting: the Rome-Ciampino ACC (Area Control Centre). We visited the centre and interviewed a number of controllers working in it. It hosts a number of en-route controllers' teams plus an approach working position in charge of controlling and sequencing the aircraft access (e.g.: landing) to the runways of the close major Fiumicino airport. As far as the en-route working position is concerned, the airspace is partitioned, by horizontal and vertical divisions, into a number of geographical regions known as *sectors*. Aircraft in an en route sector are managed by two controllers working closely, but having individually different roles and concerns: the *executive* controller maintains continuous contact with aircraft using the VHF radio and headphones and is directly responsible for maintaining the appropriate separation distance between aircraft; the *strategic or planning* controller basically performs medium-long term planning (identify future conflicts, planning future traffic), updates the system and decides how to separate flights also co-ordinating with strategic controllers of adjacent sectors.

As Figure 1 shows, the en-route position is equipped with five screens on which not only the aircraft are visualised (colourful and interactive flight labels are used to manage the different planes in the sector), but also electronic information (flight strips, monitoring data, sector boundaries, airways, etc.) is displayed.

If, on the one hand, the activities performed in the *en-route* working position make effective the paper-stripless environment available in Ciampino ACC centre, on the other hand this situation is different for the *approach* working position, where strict time constraints require an even quicker interaction between the different controllers working in the same position and a fully real-time awareness of the current ongoing activities, which can be reached only by *paper* strips that enable rapid annotations, and

facilitates information sharing (the paper strip rack is generally located in-between the two controllers).

Apart from other roles (a *chief* controller, a *technician* supervisor, one *flow controller),* there are also three (or more) *supervisors* that receive various types of alarms and have also the responsibilities of making decision about closing/opening sectors (usually dynamically divided in a vertical manner), depending on data about the estimated traffic size and the airport capacity, and also handling personnel resources. They can be regarded as the only role without any "dedicated" position within the control room.

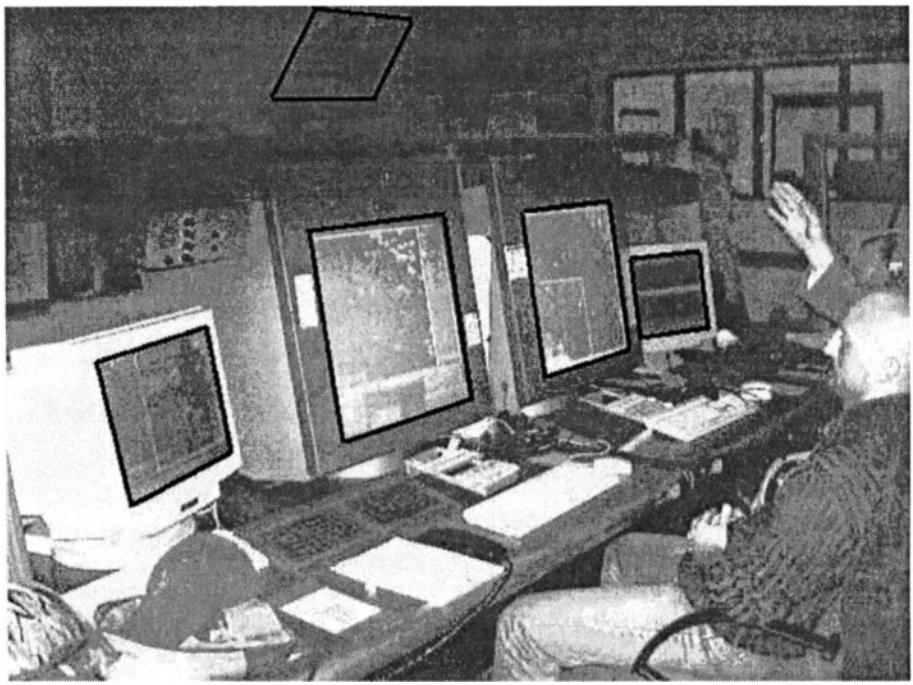

Figure 2. En-route position at the Air Traffic Control Centre in Ciampino.

5. THE METHOD APPLICATION TO THE CASE STUDY

We consider two controllers' activities related to the Ciampino case study in order to show how to apply our method.

5.1 Take over position between two controller teams (approach working position)

5.1.1 Description of the scenario

During the working day there is a number of shifts and handovers of positions (when a different team of controllers prepares to take-over responsibility of the sector) within the control room, since no more than two hours is recommended in front of a radar screen. For example, in the Approach working position –the busiest in the control room and the only one in which *paper* strips are still used– the handover requires from the new team to build the mental picture and reconstruct the traffic events in the previous working period by accessing the traffic context (e.g.: observing a few minutes the air traffic, asking questions, hearing communications between controllers in charge, ...) and looking at the paper strips (which provide them with the history of the clearances given to the pilots and useful hints to derive the current strategies used for resolving conflicts and expedite traffic). However, especially during peak hours and in situation of high traffic, the need of taking over a position by giving *verbal* reports to the new team could be less than desirable as it might strongly interfere with controllers' activity because of interruptions, distractions and interferences of acoustical information.

5.1.2 Motivations for the chosen scenario

In the envisioned system, a possible solution to such interferences is providing (during the 'overlapping' period) the new team with a PDA, a device suitable for storing history and context, allowing them to reconstruct the current situation (while they are still able to hear communications in the room) without disturbing too much the operative team. In this way the need for verbal communications among the two teams (e.g.: verbal reporting from the controllers in charge, request of clarification from the new team, etc.) for taking over a position should be reduced to a minimum.

5.1.3 Description of the task model

In the task model shown in Figure 2 we analyse the issues involved during such 'teamwork overlapping' period and the consequent take over. Who is going to become in short time the 'new' controller has to understand what is currently going on by gathering information from the paper strips and from other information sources available (PDA, vocal information, ..) so as to continuously build/update the mental picture of the situation until

actually "take over", which means replacing the other controller and start to fully control the ATC equipment.

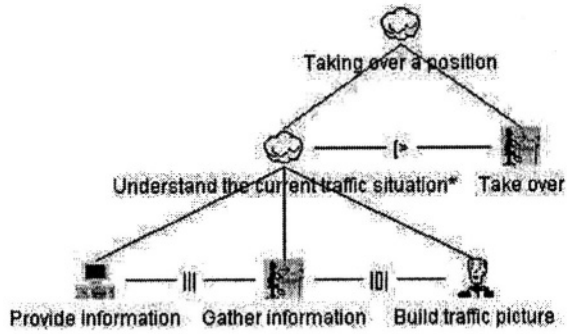

Figure 2. The task Taking over a position.

If we consider the parent task as a whole, diverse representations come into play: before actually taking over there are paper strips and vocal communications exchanged in the close proximity of the suite (which are globally shared by all the teams); information on the PDA (which is visible only to the non-operative controllers who will manage the traffic in the next future). After taking over, the PDA is supposed not to be available anymore to the operative controller who can access to a plenty of information through the screens and the headphones. One of the most interesting issues raised by the scenario is the switch between the different roles (and consequent different work contexts in terms of tools, media and platforms used) that the same person has to carry out when moving from non-operative to operative role.

Role: Approach controller taking over a position

Task: Taking over a position (replace the controller in charge and start to control the system)

1. *Analysis of the task and related properties*

1.a Task category: abstract task

1.b Task platforms: PDA, paper strips, screens (with related tools), telephone, headphone, cognitive internal information gathered during overlapping period (by hearing communications, interacting with their PDA, and eventually asking questions to controllers in charge).

1.c Task frequency: medium

1.d Other properties (that could affect task 'criticality'):

Need of starting immediately to operate in the real system (e.g.: respond to pilots) as soon as take over is completed.

2. *Analysis of the representations associated with the task.*

The representations supporting the task are: i)Mental picture of the current traffic situation (built within the overlapping period by hearing communications in the ATC environment, interacting with the PDA, asking questions to controllers in charge, etc.); ii)Graphical representation on the screens (distance might prevent the non-operative team to access all the data); iii)Audio data coming from headphones.

2.a Availability: both *external* representations (graphical: on the screens; audio: from headphones) and *internal* representation (mental picture) of ATC system.

2.b Accessibility: concurrent access to audio and graphical information (audio/graphical data continuously compared to the information maintained in controller's mental picture). Sequential access to verbal data from headphones.

2.c Mobility: the users of the PDA are supposed to hear other colleagues' conversations while using PDA: then, they should not need to move for accessing information, albeit they could

2.d Perceptibility: Data from headphones and PDA: Local; Information on screens: Shared by team members

2.e Persistence: transient access to audio information;

2.f Flexibility of cognitive tracing and interactivity: high (full control of the system).

2.g Operations and actions supported

2.g.1 Comparability with other objects/representations available in the task space: Low. Representations on the screens not immediately comparable to that they received on the PDA (used during the overlapping period), because of diversities between the supporting platforms;

2.g.2 Combinability (should allow user to select novel forms of combinations of information): Low. Controllers have to switch from using PDA to using huge screens; no combination is allowed.

2.g.3 Ease of production: Low. Every personalisation allowed on the PDA (personal device, with limited capabilities and no possibility to update the real system) is banned on the huge screens operating in the real system (which are shared by different teams, and have plenty of interaction techniques to modify the system), in order to prevent the new team to have difficulties in interpreting views resulting from adaptation processes.

3. Analysing deviations from the task plan

3.a Guideword: Other than (when talcing over, the information provided to controllers is different from that expected)

3.a. 1 Less: the information provided is less than required

Causes: controllers have annotated some information on their PDA, and this device is supposed not to be used anymore in the fully operative system.

Consequences: possible overload on the memory of the controllers, who tries to recall their annotations and possible time wastage for rebuilding such information; possible distractions for the controller.

Recommendations: enable controllers to have quickly available their annotations (also providing automatic deletion of 'obsolete' information) in the real system

3.a.2 More: the information provided is more than expected

Causes: the fully operative system has to provide information on every aircraft (a/c) controlled, whereas during the overlapping period the controllers had put their attention just on some selected a/c (those with 'highest priority', according to some priority criteria)

Consequences: controller wastes time identifying concerned a/c

Recommendations: enable the controller to e.g. select an a/c on the PDA and make it highlighted on the huge screen (improve combinability) and comparability (easiness in making associations between the two views)

3.a.3 Different: the information provided is different from expected

Causes: the representations provided by the PDA in the overlapping period might have slightly biased the controller's mental picture against the representations available in the fully operative system with huge screens and headphones (the representations may strongly differ), if the two representations are not immediately comparable.

Consequences: overload on the controllers' memory (they try to suit their mental picture to the current situation as it is represented in the fully operative system)

Recommendations: provide mechanisms to "smoothly" move from one device/picture to another, fully exploiting any possible cognitive progress the controller might have performed in the overlapping period (in anticipating future actions, decision making, strategy planning, etc.); consider the possibility of actively transferring information from PDA to real system (controllers should be enabled to easily make the correspondences between the different views: comparability) plus the possibility of using PDA as a control device for the real system -although for a short period of time (combinability).

The recommendations highlight under which conditions the use of a mobile device can provide useful support in the context considered.

5.2 De-combining air traffic sectors

5.2.1 Description of the scenario

In a situation of critical meteorological conditions, the Control Center supervisor *manages information about the upcoming traffic* in order to

trigger de-combining of two sectors. He is also *monitoring* the situation of the control working positions in order to re-distribute the controllers' workload across the control room. The ATC *supervisor* has the responsibility of making decision about closing/opening sectors (usually dynamically divided in a vertical manner), depending on data about the estimated traffic size and the airport capacity, and also personnel resources available on site. The supervisor is permanently contacted from various fixed working positions in the control room, receive various types of alarms, and need to base his decisions on integrating various sources of information.

5.2.2 Motivations for the chosen scenario

ATC supervisor can be regarded as the only role without any "dedicated" position within the control room. The need of having permanent access to real time traffic information may imply a high level of mobility of the supervisor in the control room. In this case, our hypotheses is that a mobile device, as a PDA, could offer a valuable support in carrying out his tasks.

5.2.3 Description of the task model

In order to de-combine two air space sectors, the supervisor has to identify overloaded air traffic sectors, and the level of criticality of the upcoming air traffic; at the same time, he has to evaluate the on-site workload allocation of the controllers, and also to identify available personnel for taking up the control of a new sector. The complex information supporting supervisor's task is available from several sources distributed in the task space: flight information system, air traffic monitoring system, radar, flight progress strips, meteorological information, etc. The graphical representation of the task considered is presented in Figure 3.

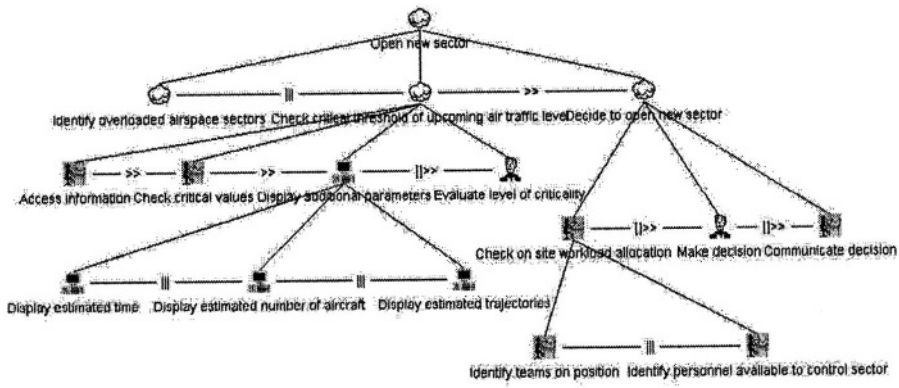

Figure 3. The "open new sector" task.

In the followings, we will look in more details at one of the subtasks identified in the above model, and reason about the possibilities of introducing a mobile device in order to support the considered task.

Role: Control Centre supervisor *Task*: Open new sector

SubTask: Check critical threshold of the upcoming air traffic level

1. Analysis of the task and related properties

1.a Task category: abstract task

1.b Task platforms: an integrated set of tools (computer displays, telephones, paper-based documentation) is available at the level of the dedicated working position in order to allow the access to radar information, flight information system, flight progress strips, meteorological information, etc.; alternatively, a mobile platform, i.e. PDA, could support the access to (some of) the required information.

1.c Task frequency: high

1.d. Other properties impacting on task 'criticality': they are, for instance, the need for real time access to information about the current and estimated up-coming air traffic level; strong dependency with the consequent task (decision to open a new sector will be further based on the information manipulated by this task).

2. Analysis of the representations associated with the task

Several information objects supporting task performance may be identified: normal and critical threshold of upcoming air traffic level, additional parameters as estimated numbers of aircraft together with time intervals planned trajectories, etc.

2.a *Availability:* they are available in both graphical and numerical representation forms.

2.b *Accessibility:* the representation forms and media (in this case, large computer screens) allow user a concurrent, easy access to a variety of information. If accessing the same information with a PDA, it is expected that its physical constraints (i.e. screen size) will make sequential the access to information, therefore increasing the time and effort needed to visualise the same items. On the other hand, a PDA would allow a permanent access to the required information, even if user changes his position across the control room.

2.c. Mobility: The user is expected to move in the control room for accessing the needed information.

2.d. Observability: similarly, using a small screen device is likely to change the observability of information, from being easily *shared* with other members of the team, to *locally* available to the user of the device.

2.e Persistence: critical information (e.g.: threshold of the upcoming air traffic level) is graphically represented, so allowing a non-transient access.

2.f Flexibility of cognitive tracing and interactivity. In the considered case, the parameters of interest are changing *autonomously,* in accordance with the real-time situation of the air traffic flow. Controllers have no or minimal permission to operate changes, to annotate or update an external electronic representation. A standard working position will be equipped with no input device (i.e., a keyboard), being allowed only direct manipulation of the objects already available on the screen.

2.g Operations and actions supported

2.g.1 Comparability. the graphical representation of information employed (i.e., clustered columns) gives user the possibility to directly *compare* various values of the monitored variables (i.e., by rapidly perceiving differences between the high of two columns).

2.g.2, 2.g.3 Combinability - possibility to combine and reconfigure or re-represent the information ofinterest: highly supported in the current ATC work settings. For instance, the information contained in an electronic flight strip can be displayed in two different formats; the values of the upcoming traffic may be represented graphically as well as numerically, etc. For the hypothetical situation of using a PDA, the question is how to effectively display the relevant information in the perimeter of a very small screen space, while maintaining a high level of interactivity. For instance, a solution would be to reduce the amount of graphics, rely mainly on the numerical representation of information, and using additional codes (i.e. sounds) in order to facilitate user's rapid discrimination of critical information.

Analysing deviations from the task plan.

By applying the proposed guidewords to some of the properties described above, it is possible to identify potential deviations from the task plan. For instance:

3.a Guideword: None (the property of the representation of interest is not available)

Information not visible.

Causes: difficulties in perceiving the relevant information due to some ergonomic issues: the object represented is too small, ambiguous shape, wrong choice of colour, etc.; but also supervisor' s distraction / interruption by other activities, etc.

Representation not persistent.

Causes: Rapid change of information values; does not allow the time user needs to internalise the perceived information and to integrate it with the other information supporting his decision making.

Consequences: if there is no information available in the task space (not visible, not persistent) then, various types of task failure can occur (i.e., stop task performance, delay, etc.).

Recommendation. Rely on multiple ways of representing the same information (i.e., visual and auditory): access to concurrent representation of the same information could be especially important for users 'on the move', who allocate their attention on several competing tasks.

Design should facilitate a rapid perception of the relevant information, and support an accurate interpretation of its signification (i.e., estimation of the air traffic flow - under/ over a critical level, or its approximate value). For instance, the discrimination of the critical information could be facilitated by adequate use of colour, use of multimedia facilities as animation, blinking images, the use of sound, etc.

6. CONCLUSIONS

Some authors have criticised task-based models of interaction, claiming that such approaches are not able to address the importance of context in interaction, and the distinction between tasks as described and task as observed in practice. A key concern in this case is the problem of characterising the context of action.

Distributed cognition is particularly concerned with the context of work and the notion of distributed representational state, and the importance of mutual knowledge in guiding action. However, DC analysis is criticised for its high qualitative approach and difficulties of translating its results in the design practice.

An integration of the two approaches could be beneficial, producing a mutual reinforcement at their both conceptual and methodological level. In this paper we have presented a method aiming at achieving such a goal. The method has been discussed within a case study in the air traffic control domain.

Future work will be dedicated to further apply the method presented in this paper to the design of interactive safety-critical systems exploiting the use of mobile devices.

ACKNOWLEDGEMENTS

This work has been supported by the EU TMR ADVISES (http://www.dcs.gla.ac.uk/advises). We also thank ENAV for allowing us to access the Ciampino ATC Centre.

REFERENCES

Buisson, M., Yannick Jestin, Design issues in distributed interaction supporting tools: mobile devices in an ATC working position, Mobile HCI 2001.

Fields, R.E., Wright, P.C., Marti, P. & Palmonari, M. (1998). Air traffic control as a distributed cognitive system: a study of external representation. Proceedings of the 9th European Conference on Cognitive Ergonomics - ECCE-9, EACE Press.

Fields, R., Paternò, F., Santoro, C., Tahmassebi, S.(1999). A method for the comparison of design options for allocating communication media in a cooperative and safety-critical context. ACM Transactions in Computer-Human Interaction Vol.6, N.4, pp.370-398, ACM Press, December 1999.

Hollan, J., Hutchins, E. & Kirsch, D. (2000). Distributed Cognition: Toward a New Foundation for Human-Computer Interaction Research, in *ACM Transactions on Computer-Human Interaction,* 7 (2), p. 174-196.

Hutchins, E. (1995) How a cockpit remembers its speeds. Cognitive Science, 19, pp. 265-288.

McDonough (2002), Boeing Addresses Air Safety with Wireless 'Flight Bag'. Wireless NewsFactor. http://wireless.newsfactor.com/story.xhtml?story_id=18486

Mertz, C., Chatty, S. and Vinot, J.-L.. The influence of design techniques on user interfaces: the DigiStrips experiment for air traffic control. In Proceedings of HCI Aero IFIP 13.5, 2000.

Mori, G., F. Paternò, C. Santoro (2002). CTTE: Support for Developing and Analyzing Task Models for Interactive System Design. IEEE Transactions on Software Engineering, pp. 797-813, August 2002 (Vol. 28, No. 8).

Norman, D. A. (1988). The psychology of everyday things, Harper Collins: Basic Books.

Paternò, F, Santoro, C. (2002). Preventing user errors by systematic analysis of deviations from the system task model. International Journal Human-Computer Studies, Elsevier Science, Vol.56, N.2, pp. 225-245.

Paternò, F. (2003). ConcurTaskTrees: an engineered notation for task models. In Dan Diaper & Neville A.Stanton (Eds.), The Handbook of Task Analysis for Human Computer Interaction. London: Lawrence Erlbaum Associates, pp.483-503.

This page intentionally left blank

Index