Hitting the Target - Realising Safety in Human Subsystems

Carl Sandom and Derek Fowler, Praxis Critical Systems Ltd, Bath, UK

Keywords: safety, human factors, human integrity targets

Abstract

Human factors are often cited as both hazard initiators and hazard mitigators within safety-related systems analyses; however, these assertions are often made without a rigorous analysis of the individual and organizational human factors involved. This paper examines the problems associated with the determination and realisation of safety requirements for the human elements of a system for which a target level of safety is specified at the service level. The paper first challenges the prevalent view that safety is largely a matter of system reliability and an argument is made for safety requirements to consider functionality and the associated performance and integrity of each safety function. A generic approach is presented for the determination of both service-level and system-level safety requirements down to the allocation of functions and safety requirements to subsystems. The paper examines issues relating to the consideration of human subsystem safety and outlines the scope and activities necessary for a comprehensive human factors safety analysis. The aim is not to consider specific Human Factors techniques in detail but rather to show how the pragmatic CONTEXT method seeks to integrate such techniques into systems engineering solutions to take account of both human capabilities and human limitations, thereby addressing the major risks to systems safety.

Introduction

In the absence of clear safety targets at the overall system or service level, it is too tempting to concentrate safety assessment effort on what we understand (ie. hardware) and what we think we understand (ie. software), and to adopt a ‘head in the sand’ approach to what we perceive as too difficult – ie human factors. Humans are often the major causal factor for hazards in safety-related systems (ref. 1) and yet don’t receive proportionate attention in safety analysis. On the other hand, human operators also often provide substantial mitigation between hazards and their associated accidents; yet this too is often overlooked or, conversely, sometimes over-stated.

If human factors risks are not properly considered, a system will probably not achieve the required target level of safety, and even if the target were achieved (by luck as much as by deliberate intent), it would certainly not be possible to prove that fact. Effective human factors input is clearly crucial to system safety. Lack of proper attention to these issues not only threatens the safety of systems but also makes poor business sense in that:

- Costly failures arise from the inability of systems to meet the needs of users.
- Through-life support, training and maintenance costs are high.
- Design is overly focused on technical artefacts to the exclusion of human and organizational factors – thus the majority of risks are under-addressed.
- When human factors are addressed, requirements are often not integrated into the systems design – thus the goals of the system are not met.
- If human factors risks are not considered, the technical system components may be over engineered at additional cost to achieve a specified target level of safety.

This paper provides a framework for the determination and realisation of functional safety requirements in human-centred systems. However, in order to ensure that human safety requirements are correctly specified, we need firstly to consider the nature of safety-related systems (SRSs) and safety requirements in general.

Safety and Systems Considerations

There is a widespread belief that safety is largely a matter of reliability (ref. 2) and yet both theory (ref. 3) and experience (ref. 4) have already shown this to be far too narrow a view of safety. All safety-related systems (SRS) – by definition - pose a threat to their environment, and we need to specify *safety integrity requirements* for such systems in order to limit this increase in risk. However, there is a subset of SRS - safety protection systems (SPS) - whose raison d'étre is to reduce risk in their environment; IEC 61508 (ref. 5) recognises that the safety of such a system depends as much on **what** it does (its *safety functions*) as on how **reliably** it does it (its *safety integrity requirements*).

The determination of *safety functions* requires initial hazard-analysis to focus on the operational environment in which an SPS functions to provide a service, rather than on the system itself. The recent trend towards objective-based safety regulatory requirements and the resultant setting of numerical target levels of safety (TLS) for SPS providing safety-related services force us to take a broader view of safety than that provided by IEC 61508, by specifying tolerable risk for the operational environment, rather than for the system itself.

To illustrate this, Figure 1 shows a simple model, in which three main levels are defined:

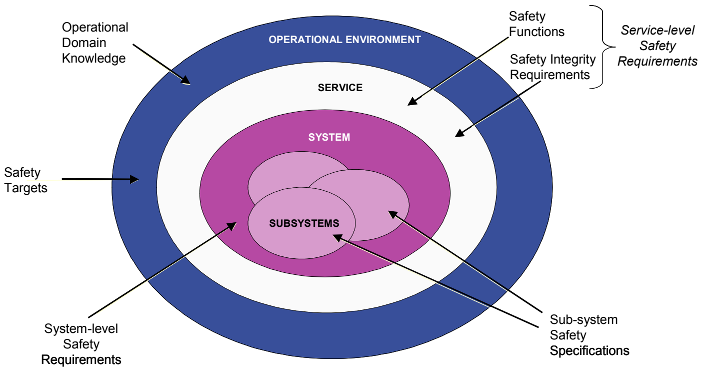
Operational Domain Knowledge
OPERATIONAL ENVIRONMENT
Safety Functions
Safety Integrity Requirements
Service-level Safety Requirements
SERVICE
SYSTEM
Safety Targets
SUBSYSTEMS
System-level Safety Requirements
Sub-system Safety Specifications

Figure 1 – System Context

- *Operational environment* (or domain) into which the SPS provides a (safety-related) service. Pre-existing, *domain hazards* to users of the safety service exist at this level.
- *Service level,* at which the safety properties of the service are defined, in abstract terms – ie independent of the eventual physical system implementation.
- *System level* at *which* the required safety properties of service are implemented in the physical system – comprising, typically, equipment, people and procedures.

The service level can be considered as the interface between the SPS and its operational environment, and hazards emanating from the system can be considered to exist in this interface.

The safety requirements determination process maps on to this model, as follows:

- *Safety targets* are what we want to **make happen** in the operational environment – ie to eliminate or mitigate *domain hazards*. They define, inter alia, what is tolerable in terms of risk.
- *Safety functions* specify **what** the service has to provide to the operational environment – including the level of **performance** required – in order to meet the *safety targets*.
- *Safety integrity requirements* limit the probability that the *safety functions* will not meet their specified function and performance - ie they limit the occurrence of *system hazards*.

- *Operational domain knowledge* covers those pre-existing properties of the operational environment, which are **known**, or have to be **assumed**, to be true; such knowledge is critical to whether the service-level *safety functions* and their *safety integrity requirements* will meet the *safety targets*.
- System-level *safety requirements* are those safety properties required of the **physical** system in order to implement the (service-level) *safety functions* and the *safety integrity requirements*. They are expressed in terms of the functionally, performance and integrity required of each subsystem – ie including people, procedures and equipment. *System domain knowledge* may also be specified at this level.
- Where the design of a subsystem is further decomposed, the term *safety specification* is often used to describe the safety properties of COTS and other pre-existing components.

Safety Requirements Determination

The following framework is a traceable, staged implementation of the model in Figure 1 from operational-level safety targets to implementation in equipment and human-based subsystems.

Service-level Requirements: Figure 2 illustrates the first step of determining service-level *safety functions* and *integrity requirements*. The Safety Target(s) specify the domain hazards to be addressed by the SPS and associated target level of safety. The *safety functions* and performance (eg accuracy, capacity, timeliness etc, but **excluding** integrity), are specified so as to meet the *safety targets* to be met. It is necessary at this stage to carry out a performance-risk assessment in order to show that specified *safety functions* are sufficient to reduce the risk from the domain hazards (R_D) to a level **well below** the target (R_T), since R_T-R_D represents the portion of the *safety target* that can be allocated to (functional) failure. *Safety integrity requirements* are obtained from hazard and risk analysis at this level, and limit the allowable rate of occurrence of each *safety function* failure mode (ie hazard) such that the total risk for the identified hazards (R_S) satisfies $R_S + R_D < R_T$, taking account of any *mitigations* that are identified during the process. All mitigations of the consequences of the identified hazards must be captured as either:

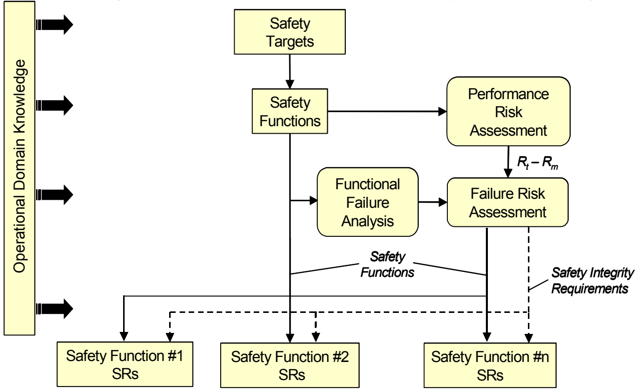

Figure 2 – Service-level Safety Requirements

- Additional *safety functions* and related safety integrity requirements, for the provision of "deliberate" mitigations.
- *Operational domain knowledge* regarding any *assumptions* about "circumstantial" mitigations (ie those arising as a matter of pure chance).

At this point a *satisfaction argument* (ref. 2) needs to show that the (service-level) *safety functions* and *safety integrity requirements* would meet the *safety targets*, given the *domain knowledge*.

System-level Requirements: The specification of *safety requirements* follows from an architectural design of the system, as illustrated in Figure 3. The process is similar to that for the service level, as described above. The **primary** *safety requirements* stem from an allocation of the service-level *safety functions* to the subsystem(s) on which they are to be implemented. The example illustration in Figure 3 shows typical ATM equipment sub-systems (air-ground communications, radar data processing, flight data processing, and display) and human-based subsystems (executive and planner controllers). The hazards and risks associated with failure of each subsystem are assessed, any mitigations are identified and allocated (as *domain knowledge* or additional *safety functions*, as appropriate), and the *safety integrity requirements* for each subsystem determined – these safety properties being known collectively as **derived** *safety requirements.* The outputs from this stage are therefore:

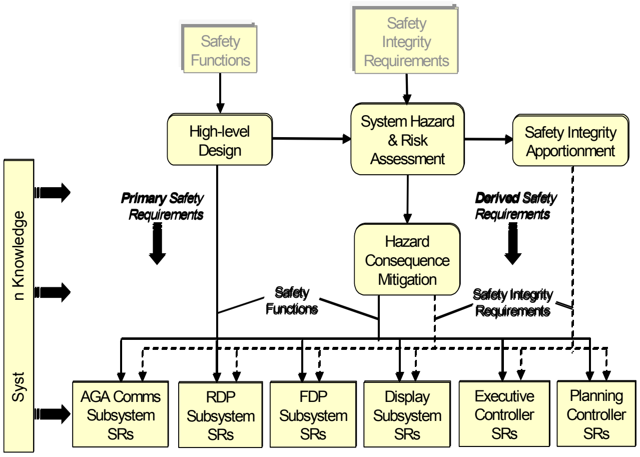
Safety Functions
Safety Integrity Requirements
High-level Design
System Hazard & Risk Assessment
Safety Integrity Apportionment
Primary Safety Requirements
Derived Safety Requirements
Hazard Consequence Mitigation
Safety Functions
Safety Integrity Requirements
Syst
n Knowledge
AGA Comms Subsystem SRs
RDP Subsystem SRs
FDP Subsystem SRs
Display Subsystem SRs
Executive Controller SRs
Planning Controller SRs

Figure 3 – System-Level Safety Requirements

- *Safety functions*, to be *implemented* by each subsystems, and the performance required of them.
- Specification of the interactions and interfaces between the subsystems.
- *Safety integrity requirements* for each subsystem.

Finally assurance is required that each service-level *safety function*, with its corresponding *safety integrity requirements*, is implemented by the subsystem *safety requirements,* given the *domain knowledge,* and taking account of any new hazards that emerge at *subsystem* level.

The remainder of the paper examines how the above generic approach can be developed in the case of human-based subsystems, using Human Factors techniques.

Human Factors Considerations

Human Factors (HF) broadly covers the ergonomic, organizational and social aspects of a system in its context of use. HF analyses address the need to match technology with humans operating within a particular environment, in order to meet the higher (service) level safety requirements. HF seeks to promote appropriate job and task design, suitable physical environments and workspaces, human-machine interfaces and the appropriate selection, training and motivation of the humans involved. At the detailed level, HF analyses must examine how the design of human-computer interactions can foster the efficient transmission of information between the human and machine, in a form suitable for the task demands and human physical and cognitive capabilities.

Design is an iterative process and a high-level architectural design will normally be produced by the allocation of functionality to subsystems as shown in Figure 3. If the human factors are taken into consideration (though sometimes they are not!), then the initial high-level design decisions will take human capabilities and limitations into consideration when allocating functions to man and/or machine. Task Analysis (TA) is the term applied to the process that identifies and examines tasks performed by humans as they interact with systems. Essentially, the design of a human subsystem will typically be expressed as the output from the TA (ref. 6).

Figure 1 has shown that the scope of HF analyses must address the system, service and operational environment. This vast scope presents a challenge for the systems engineer who needs to consider the safety-related aspects of the system and even then to focus the often limited resources available on the most critical system functions. Figure 3 has shown a generic process for deriving the system-level safety requirements from a high-level architectural design. However, the processes for determining **primary** safety requirements and producing **derived** safety requirements will necessarily be based upon on different analysis techniques when dealing with human rather than technical subsystems.

A pragmatic method is required to focus HF analysis on the safety-related aspects of the system using suitable HF techniques. The method should support the determination of human subsystem safety requirements using HF techniques that are integrated into the systems engineering life-cycle. One such method to achieve this is outlined in the remainder of the paper.

Human Subsystems Safety in CONTEXT

CONTEXT is a method for integrating the use of appropriate HF analysis techniques within the systems engineering lifecycle for the systematic determination and realisation of HF safety requirements. As discussed in the previous section, the key to safety assurance is to ensure that each causal factor (people, procedures, equipment) must be considered within and between each system level (Figure 1) to a depth of analysis commensurate with the integrity required of the human subsystem. Although the analysis begins with the core-system/subsystem level, CONTEXT takes into account the human risks and mitigations at the service and operational levels. Although these techniques are described below in relation to the human subsystems, the method also provides for the analysis of human risks and mitigations at the service and operational levels.

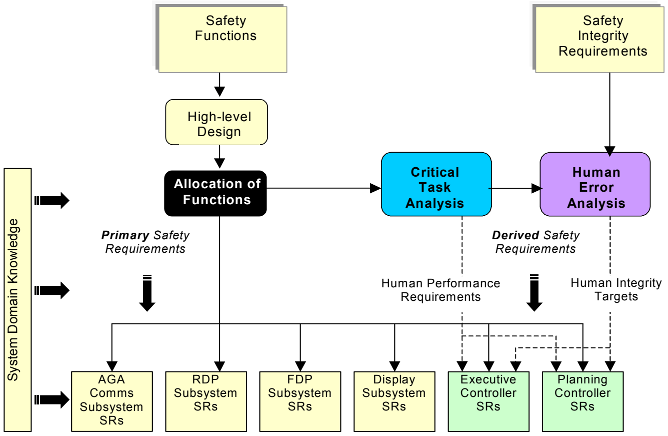

Figure 4 – Human Subsystem Safety Requirements

Figure 4 depicts a specific process for deriving the human subsystem safety requirements from a high-level architectural design.

An important feature of Figure 4 is that the high-level design must take into consideration the human factors in the initial Allocation of Functions. Too often, this decision is based upon technical capability and the human is allocated whatever functionality can't be implemented in hardware or software, regardless of the suitability of the human to undertake the resultant tasks.

Specific safety-related HF activities comprise Critical Task Analysis (CTA) and Human Error Analysis (HEA). These safety-specific activities are planned to ensure that there is no overlap with the wider, system-level HF programme while taking maximum advantage of System Hazard and Risk Assessment analyses for the non-human subsystems (depicted in Figure 3). CTA and HEA activities are tightly coupled and are based upon, and integrated with, the findings of other typical safety analyses such as Fault Tree Analysis (FTA).

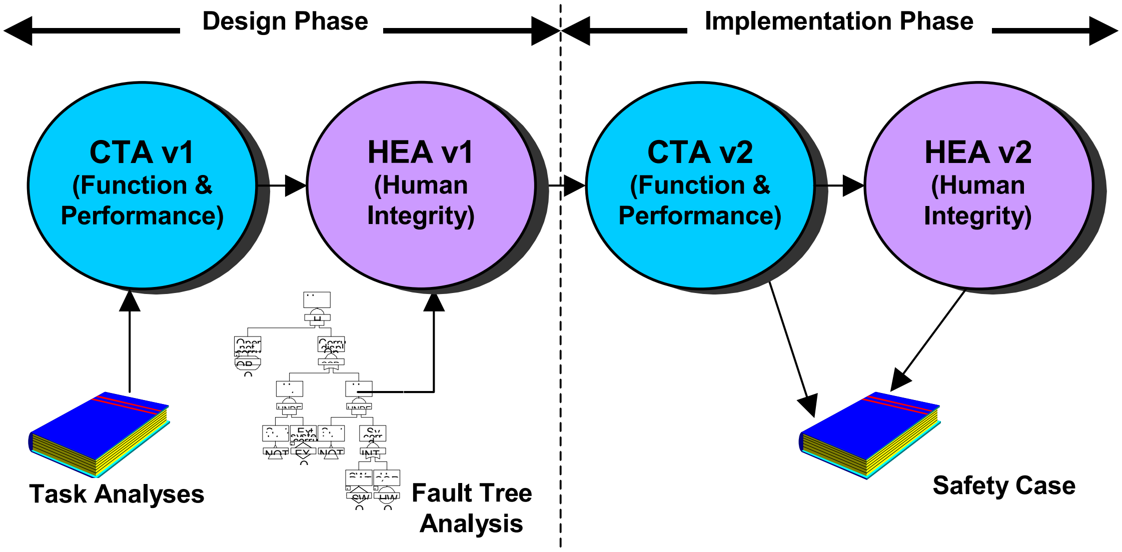

Figure 5: CONTEXT Analyses

Typically, two iterations of each technique would be undertaken and, as the analyses become more focused, the results will inform each other as shown in Figure 5. These activities are complementary as CTA and HEA are bottom-up and top-down analysis techniques respectively (from a hazard to human event perspective). This combination of top-down and bottom-up analyses significantly increases the probability of identifying inconsistencies in the individual techniques and thus enhances safety assurance. A detailed examination of the application of these analyses follows.

Application of CONTEXT Analyses

Initial Allocation of Function: The production of a high-level architectural design, as shown in Figure 4, requires initial decisions to be made on the allocation of functions to human or equipment sub-systems, in full knowledge of the safety risks involved. Functional allocation decisions need to be informed by good human factors principles and yet the allocation of function is still considered predominantly an ergonomics problem in many design communities. The early work of Fitts (ref. 7) is often used to derive MABA-MABA (Men Are Better At - Machines Are Better At) lists that are typically restricted to considerations of **either** the human or the machine performing each individual function. However, since Fitts' early work, it has become apparent that many functions in complex systems require apportionment of the function between **both** human and machine. An extensive discussion on functional allocation is beyond the scope of this paper; however, for a detailed review of function allocation techniques see Older *et. al.* (ref. 8).

Human Safety Functions: A CTA can be undertaken to identify and analyse the potential for human performance errors in critical operational tasks. CTA concentrates on the HF aspects of the Human Machine Interface (HMI). This analysis is a bottom-up technique used broadly to analyse the relationships between system hazards (identified by the System Hazard Assessment in Figure 3) and operational tasks (identified by TA) and the HMI design. The analysis work in a bottom-up fashion from operational tasks, related to base events, to identified service-level hazards.

A CTA can concentrate initially on the identification and analysis of the relationships between system hazards and safety-related operational tasks. This analysis will enable both the PHA and TAs to be checked for consistency, providing confidence in subsequent safety assurance claims. Any deficiencies - such as hazards with no related operational tasks or operational tasks (deemed as safety-related by subject matter experts) with no relationship to identified hazards - can be highlighted.

The analysis will also look for opportunities for hazard mitigation through removal of human error potential and improved information presentation by comparing the TA with HMI design guidelines from appropriate sectors (for example Federal Aviation Authority ATM HMI design guidelines (ref. 9)).

In summary, the CTA will enable the systems engineer to:

- Define the allocated *safety functions* in terms of human operator tasks, including potential mitigations to be provided by the Operator in the event of failure of other subsystems.
- Capture the interactions and interfaces between the human and equipment subsystems.
- Determine task skills, knowledge and procedure requirements and record these as additional functional *safety requirements*.
- Confirm feasibility regarding human capabilities performance and reallocate inappropriate tasks to equipment (ie tools, automation etc) as functional *safety requirements*.
- Identify training requirements and record these as functional *safety requirements*.
- Determine human information requirements and human-machine interaction requirements and record these as functional *safety requirements*.

Human Integrity Targets: For highly interactive systems situated in dynamic environments, the quality of the information acquired through the interface can contribute significantly to system failure, and the design of the human-computer interface can have a profound effect on operator performance and system safety. It is imperative that qualitative and, where appropriate, quantitative safety arguments are made for each critical human failure linked to service-level hazards identified during the System Hazard Assessment. The depth of analysis required to make a compelling safety argument for each critical human event must be determined by these derived human integrity requirements. Analyses should also identify opportunities for hazard mitigation through removal of human error potential and improved information presentation.

In safety-related systems, the derivation of quantitative human integrity targets is difficult. Human Reliability Analysis techniques have attempted to address this issue (ref. 10);, however, much of this research has been dominated by assumptions that apply to technical systems and arguably these do not translate well to human systems. A pragmatic method of addressing this issue is to undertake a Human Error Analysis (HEA) focused specifically on the basic human events identified in the system Fault Trees.

The HEA analysis is a top-down technique used broadly to model the relationship between critical human failures and service-level hazards, and the mitigating aspects of the system design. For systems, which typically have a high degree of operator interaction, many of the FTA basic events will be identified as human interactions. An example fragment of an FTA is shown in Figure 6 with the basic human event OP NOT DET.

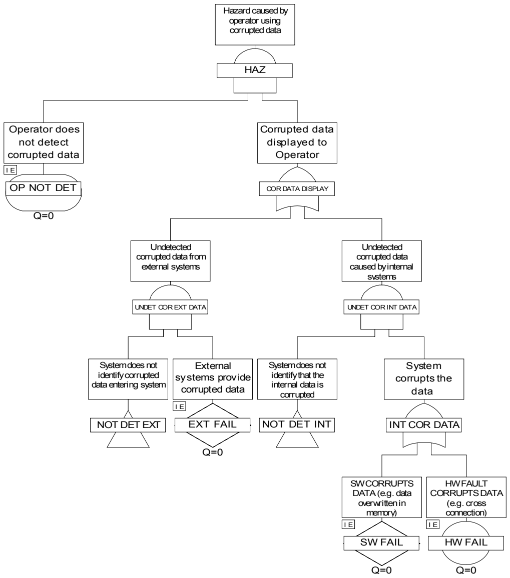
Hazard caused by operator using corrupted data
HAZ
Operator does not detect corrupted data
I E
OP NOT DET
Q=0
Corrupted data displayed to Operator
COR DATA DISPLAY
Undetected corrupted data from external systems
UNDET COR EXT DATA
Undetected corrupted data caused by internal systems
UNDET COR INT DATA
System does not identify corrupted data entering system
NOT DET EXT
External systems provide corrupted data
I E
EXT FAIL
Q=0
System does not identify that the internal data is corrupted
NOT DET INT
System corrupts the data
INT COR DATA
SW CORRUPTS DATA (e.g. data overwritten in memory)
I E
SW FAIL
Q=0
HW FAULT CORRUPTS DATA (e.g. cross connection)
I E
HW FAIL
Q=0

Figure 6 – Fault Tree Example

Once each fault tree is modelled, predictive, quantitative failure data can be input at the bottom from Availability and Reliability data for all hardware and software base events. By subtracting these values from the associated hazard target, quantitative human integrity targets (HITs) can then be calculated for each critical human event. It should be understood that these basic human events originate from both the system and service levels taking the operational context into account.

The HEA would then focus on developing specific safety arguments for each basic human event to provide evidence that the HITs can be achieved. For critical areas, where the HEA reveals that the HITs are unrealistic, mitigations can be re-assessed and recommendations developed for further action. In this way, no predictions are being made about the human error rates; rather, the HITs are derived from the remaining integrity requirements once the hardware and software failure data is input and an analysis is undertaken to ascertain if the remaining human integrity requirements are realistic.

Summary

The CONTEXT method described here has been used within the development of large Command and Control systems by integrating the use of the CTA and HEA analysis techniques within the systems engineering lifecycle and addressing all system levels. Importantly, and entirely consistent with the design requirements process previously described, the outputs from these analyses are safety requirements:

- Safety functions, and performance from the CTA.
- Human integrity targets from the HEA.

Given the questionable nature of any quantitative human error rates calculated using current HRA techniques, it is suggested here that CONTEXT is a pragmatic method for ensuring the systematic determination and realisation of human subsystems safety requirements by integrating HF analysis techniques into a systems development lifecycle and making maximum use of typical systems analyses.

Conclusions

This paper examined the problems associated with the determination and realisation of functional safety requirements for the human elements of a system for which a target level of safety is specified at the service level. The paper challenged the prevalent view that safety is largely a matter of system reliability and an argument was made to show that safety requirements exist at all system levels (service and core system) and these need to take into consideration functionality and its associated performance and integrity. The high-level allocation of functions to hardware, software or humans must be done by taking human performance and limitations into account and a generic approach was presented for the determination of both service-level and system-level safety requirements down to the allocation of functions and safety requirements to subsystems. The determination of Human subsystem safety requirements are no different to software or hardware and the paper examined issues relating to the consideration of human subsystem safety and outlined the scope and activities necessary for a comprehensive human factors safety analysis. CONTEXT was introduced as a pragmatic method for the application of Human Factors techniques to the realisation of safety for human subsystems. In summary, this paper has shown how Human Factors can be an integrated systems engineering discipline and the use of appropriate specialist techniques and methods can be brought to bear for the determination and realisation of human subsystem safety requirements.

References

1. Sandom C: *Human Factors Considerations for System Safety*, in Components of System Safety, Redmill F and Anderson T (Eds.), proceedings of 10th Safety Critical Systems Symposium, 5th-7th February 2002 Southampton, Springer-Verlag, UK, Feb 02.
2. Fowler D, Sandom S, Simpson A J, *Challenging Safety Regulation – a Wake-up Call*, Proceedings of the 20th International System Safety Conference, Denver, USA Aug 02
3. Fowler D, *Application of IEC 61508 to Air Traffic Management and Similar, Complex, Critical Systems* – Proceedings of the 8th Safety-Critical Systems Symposium, UK, Feb 00.
4. Nancy G Leveson, *The Role of Software in Recent Aerospace Accidents*, Proceedings of the 19th International System Safety Conference, Huntsville, Alabama, USA Sep 01.
5. International Electrotechnical Commission, *IEC 61508, Functional Safety of Electrical/ Electronic/ Programmable Electronic Safety Related Systems*, 65A/254/FDIS, IEC:2000.
6. Kirwan B and Ainsworth L K (Eds.), *A Guide to Task Analysis*, Taylor and Francis, 1992.
7. Fitts P.M., *Human Engineering for an Effective Air Navigation and Traffic Control System*, National Research Council, Washington D.C., 1951.
8. Older M.T., Waterson P.E. and Clegg C.W., 1997, *A Critical Assessment of Task Allocation Methods and their Applicability*. Ergonomics, 40(2), pp 151 – 171, 1997.
9. Human Factors Design Guide Update (DOT/FAA/CT-96/01): A Revision to Chapter 8- Computer Human Interface Guidelines, National Technical Information Service, April 2001.
10. Kirwan B: *A Guide to Practical Human Reliability Assessment*, Taylor & Francis, 1994.

Biographies

Carl Sandom and Derek Fowler are both Principal Consultants with Praxis Critical Systems. Between them, they have over 50 years practical experience covering all aspects of a typical system life-cycle in high-integrity and safety-related systems procurement, development and maintenance environments.

Carl specialises in systems safety assurance and human factors and has developed and implemented a variety of system safety management programmes within the UK and overseas. Prior to entering consultancy, he spent over 20 years as engineer and manager in the Aerospace and Defence sectors both as an officer in the Royal Air Force and later within industry. He has published and presented numerous human factors and safety papers and has contributed to books on the subject. He is a Chartered Engineer and a Registered Member of the Ergonomics Society.

Derek began his career with the Royal Air Force as an engineering officer, specialising in avionics and guided weapons systems, and then worked in industry on major avionics development projects. He has specialised mainly in air traffic management development since 1990 and for the past 5 years has worked as a safety consultant in that field, playing a key role in several new ATM development programmes. He has also played a major role in the development of safety assurance standards and practical solutions for complying with quantitative target levels of safety. He published and presented numerous papers on system safety issues.

Dr. Carl Sandom CEng MErgS
direct: +44 (0) 1225 823773
mobile : +44 (0) 7967 672560
carl.sandom@praxis-cs.co.uk

Praxis Critical Systems Ltd
20 Manvers St, Bath, BA1 1PX, UK
t: 44 (0)1225 466991
f: 44 (0)1225 469006
www.praxis-cs.co.uk

Derek Fowler CEng
direct: +44 (0) 1491 411952
mobile : +44 (0) 7730 487705
derek.fowler@praxis-cs.co.uk

Praxis Critical Systems Ltd
20 Manvers St, Bath, BA1 1PX, UK
t: 44 (0)1225 466991
f: 44 (0)1225 469006
www.praxis-cs.co.uk