Capturing Emerging Complex Interactions - Safety Analysis in ATM

Massimo Felici
LFCS, School of Informatics, The University of Edinburgh, Edinburgh EH9 3JZ, UK
http://homepages.inf.ed.ac.uk/mfelici/

Abstract: The future development of Air Traffic Management (ATM), set by the ATM 2000+ Strategy, involves a structural revision of ATM processes, a new ATM concept and a systems approach for the ATM network. This requires ATM services to go through significant structural, operational and cultural changes that will contribute towards the ATM 2000+ Strategy. Moreover, from a technology viewpoint, future ATM services will employ new systems forming the emergent ATM architecture underlying and supporting the European Commission's Single European Sky Initiative. Introducing safety relevant systems in ATM contexts requires us to understand the risk involved in order to mitigate the impact of possible failures. This paper is concerned with some limitations of safety analyses with respect to operational aspects of introducing new systems (functionalities).

Keywords: Safety Analysis, ATM, Complex Interactions, System Evolution.

Introduction

The future development of Air Traffic Management (ATM), set by the ATM 2000+ Strategy (EUROCONTROL, 2003), involves a structural revision of ATM processes, a new ATM concept and a systems approach for the ATM network. The overall objective (EUROCONTROL, 2003) is, *for all phases of flight, to enable the safe, economic, expeditious and orderly flow of traffic through the provision of ATM services, which are adaptable and scalable to the requirements of all users and areas of European airspace*. This requires ATM services to go through significant structural, operational and cultural changes that will contribute towards the ATM 2000+ Strategy. Moreover, from a technology viewpoint, future ATM services will employ new systems forming the emergent ATM architecture underlying and supporting the European Commission's Single European Sky Initiative. ATM services, it is foreseen, will need to accommodate an increasing traffic, as many as twice number of flights, by 2020. This challenging target will require the cost-effectively gaining of extra capacity together with the increase of safety levels (Matthews, 2002; Overall, 1995). Enhancing safety levels affects the ability to accommodate increased traffic demand as well as the operational efficiency of ensuring safe separation between aircrafts. Suitable safe conditions shall precede the achievement of increased capacity (in terms of accommodated flights). Therefore, it is necessary to foreseen and mitigate safety issues in aviation where ATM can potentiality deliver safety improvements.

Introducing safety relevant systems in ATM contexts requires us to understand the risk involved in order to mitigate the impact of possible failures. Safety analysis involves the activities (i.e., definition and identification of system(s) under analysis, risk analysis in terms of tolerable severity and frequency, definition of mitigation actions) that allow the systematic identification of hazards, risk assessment and mitigation processes in critical systems (Leveson, 2005; Storey, 1996). Diverse domains (e.g., nuclear, chemical or transportation) adopt safety analyses that originate from a general approach (Leveson, 2005; Storey, 1996). Recent safety requirements, defined by EUROCONTROL (European organization for the safety of air navigation), imply the adoption of a similar safety analysis for the introduction of new systems and their related procedures in the ATM domain (EUROCONTROL, 2001a). Unfortunately, ATM systems and procedures have distinct characteristics (e.g., openness, volatility, etc.) that expose limitations of the approach. In particular, the complete identification of the system under analysis is crucial for its influence on the cost and the effectiveness of the safety analysis. Some safety-critical domains (e.g., nuclear and chemical plants) allow the properly application of conventional safety analyses. Physical design structures constrain system's interactions and stress the separation of safety related components from other system parts. This ensures the independence of failures. In contrast, ATM systems operate in open and dynamic environments where it is difficult completely to identify system interactions.

For instance, there exist complex interactions between aircraft systems and ATM safety relevant systems. Unfortunately, these complex interactions may give rise to catastrophic failures. The accident (1 July 2002) between a BOING B757-200 and a Tupolev TU154M (BFU, 2004), that caused the fatal injuries of 71 persons, provides an instance of unforeseen complex interactions. These interactions triggered a catastrophic failure, although all aircraft systems were functioning properly. Hence, safety analysis has to take into account these complex interaction mechanisms (e.g., failure dependence, reliance in ATM, etc.) in order to guarantee and even increase the overall ATM safety as envisaged by the ATM 2000+ Strategy.

This paper is concerned with some limitations of safety analyses with respect to operational aspects of introducing a new system (functionality). The paper is structured as follows. Firstly, it introduces safety analysis in ATM domain. The EUROCONTROL Safety Regulatory Requirement (EUROCONTROL, 2001a), ESARR4, requires the use of a risk based-approach in ATM when introducing and/or planning changes to any (ground as well as onboard) part of the ATM System. Unfortunately, ATM systems, procedures and interactions expose limitations of safety analyses. This paper proposes a framework for capturing complex interactions. The framework supports the iterative aspects of safety analyses. It, finally, discusses the proposed framework and draws some conclusions.

Safety Analysis in ATM

ATM services across Europe are constantly changing in order to fulfill the requirements identified by the ATM 2000+ Strategy (EUROCONTROL, 2003). Currently, ATM services are going through a structural revision of processes, systems and underlying ATM concepts. This highlights a systems approach for the ATM network. The delivery and deployment of new systems will let a new ATM architecture to emerge. The EUROCONTROL OATA project (Skyway, 2004) intends to deliver the Concepts of Operation, the Logical Architecture in the form of a description of the interoperable system modules, and the Architecture Evolution Plan. All this will form the basis for common European regulations as part of the Single European Sky.

The increasing integration, automation and complexity of the ATM System requires a systematic and structured approach to risk assessment and mitigation, including hazard identification, as well as the use of predictive and monitoring techniques to assist in these processes. Faults (Laprie et al, 1998) in the design, operation or maintenance of the ATM System or errors in the ATM System could affect the safety margins (e.g., loss of separation) and result in, or contribute to, an increased hazard to aircrafts or a failure (e.g., a loss of separation and an accident in the worst case). Increasingly, the ATM System relies on the reliance (e.g., the ability to recover from failures and accommodate errors) and safety (e.g., the ability to guarantee failure independence) features placed upon all system parts. Moreover, the increased interaction of ATM across State boundaries requires that a consistent and more structured approach be taken to the risk assessment and mitigation of all ATM System elements throughout the ECAC (European Civil Aviation Conference) States (EUROCONTROL, 2001). Although the average trends show a decrease in the number of fatal accidents for Europe, the approach and landing accidents are still the most safety pressing problems facing the aviation industry (Ranter, 2003; Ranter, 2004; van Es, 2001). Many relevant repositories[1] report critical incidents involving the ATM System. Unfortunately, even maintaining the same safety levels across the European airspace would be insufficient to accommodate an increasing traffic without affecting the overall safety of the ATM System (Enders, Dodd, and Fickeisen, 1999).

The introduction of new safety relevant systems in ATM contexts requires us to understand the risk involved in order to mitigate the impact of possible failures. The EUROCONTROL Safety Regulatory Requirement (EUROCONTROL, 2001a), ESARR4, requires the use of a risk based-approach in ATM

[1] Some repositories are: Aviation Safety Reporting Systems - http://asrs.arc.nasa.gov/ -; Aviation Safety Network - http://aviation-safety.net/ -; Flight Safety Foundation: An International Organization for Everyone Concerned With Safety of Flight - http://www.flightsafety.org/ -; Computer-Related Incidents with Commercial Aircraft: A Compendium of Resources, Reports, Research, Discussion and Commentary compiled by Peter B. Ladkin et al. - http://www.rvs.uni-bielefeld.de/publications/Incidents/ -.

when introducing and/or planning changes to any (ground as well as onboard) part of the ATM System. This concerns the human, procedural and equipment (i.e., hardware or software) elements of the ATM System as well as its environment of operations at any stage of the life cycle of the ATM System. The ESARR4 (EUROCONTROL, 2001a) requires that ATM service providers systematically identify any hazard for any change into the ATM System (parts). Moreover, they have to assess any related risk and identify relevant mitigation actions. In order to provide guidelines for and standardize safety analysis EUROCONTROL has developed the EATMP Safety Assessment Methodology (SAM) (EUROCONTROL, 2004) reflecting best practices for safety assessment of Air Navigation Systems.

The SAM methodology provides a means of compliance to ESARR4. The SAM methodology describes a generic process for the safety assessment of Air Navigation Systems. The objective of the methodology is to define the means for providing assurance that an Air Navigation System is safe for operational use. The methodology describes a generic process for the safety assessment of Air Navigation Systems. The process consists of three major steps: *Functional Hazard Assessment (FHA)*, *Preliminary System Safety Assessment (PSSA)* and *System Safety Assessment (SSA)*. Figure 1 shows how the SAM methodology contributes towards system assurance. The process covers the complete life cycle of an Air Navigation System, from initial system definition, through design, implementation, integration, transfer to operations, to operations and maintenance. Moreover, it takes into account three different types of system elements (human, procedure and equipment elements), the interactions between these elements and the interactions between the system and its environment.

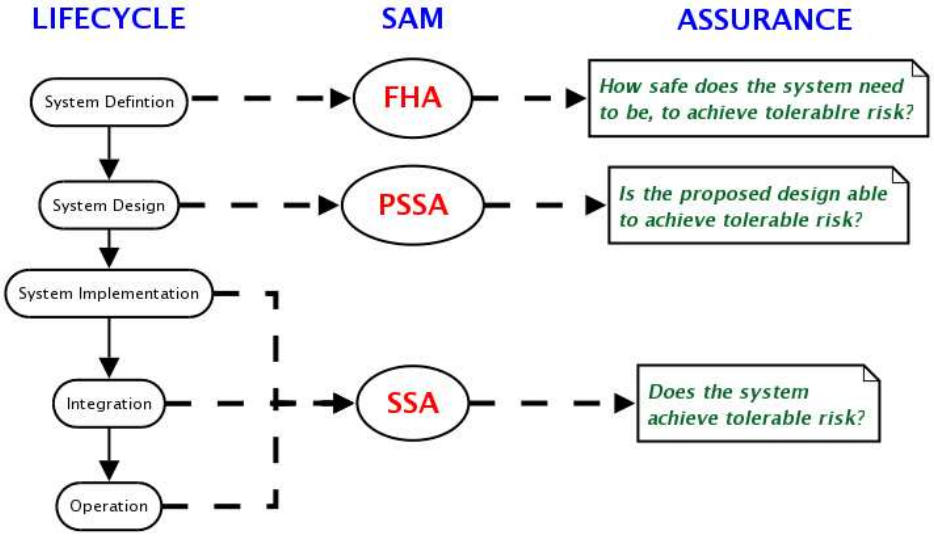
LIFECYCLE
SAM
ASSURANCE
System Defintion
System Design
System Implementation
Integration
Operation
FHA
PSSA
SSA
How safe does the system need to be, to achieve tolerablre risk?
Is the proposed design able to achieve tolerable risk?
Does the system achieve tolerable risk?

Figure 1 - Contribution of the Safety Assessment Methodology towards system assurance

The FHA is a top-down iterative process, initiated at the beginning of the development or modification of an Air Navigation System. The objective of the FHA process is to determine the overall safety requirements of the system (i.e., specifies the safety level to be achieved by the system). The process points out potential functional failures modes and hazards. It assesses the consequences of their occurrences on the safety of operations, including aircraft operations, within a specified operational environment. The FHA process specifies overall Safety Objectives of the system. The PSSA is another top-down iterative process, initiated at the beginning of a new design or modification to an existing design of an Air Navigation System. The objective of performing a PSSA is to demonstrate whether the assessed system architecture can reasonably be expected to achieve the Safety Objectives specified in the FHA. The PSSA process the Safety Objectives into Safety Requirements allocated to the system elements. That is, it identifies the risk level to be achieved by each system element. The SSA is a process initiated at the beginning of the implementation of an Air Navigation System. The objective of performing a SSA is to demonstrate that the implemented system achieves an acceptable (or at least tolerable) risk and consequently satisfies its Safety Objectives specified in the FHA. Moreover, the SSA assesses whether each system element meets its Safety Requirements specified in the PSSA. The SSA process collects evidences and provides assurance throughout the system life cycle (i.e., from implementation to decommissioning).

Although the SAM methodology describes the underlying principles of the safety assessment process, it provides limited information to applying these principles in specific projects. The hazard identification, risk assessment and mitigation processes comprise a determination of the scope, boundaries and interfaces of the constituent part being considered, as well as the identification of the functions that the constituent part is to perform and the environment of operations in which it is intended to operate. This supports the identification and validation of safety requirements on the constituent parts.

Modeling: The definition and identification of the system under analysis is extremely critical in the ATM domain and can have a significant influence on the safety analysis. System Models used during design phases provide limited support to safety as well as risk analysis. This is because existing models defined in the design phases are adapted and reused for safety and risk analysis. Organizational and cost-related reasons often determine this choice, without questioning whether models are suitable for the intended use. The main drawback is that design models are tailored to support the work of system designers. Thus, system models capture characteristics that may be of primary importance for design, but irrelevant for safety analysis. On the contrary, models should be built as working-tools that, depending on their intended use, ease and support specific cognitive operations of users, for instance, by highlighting some aspects and neglecting others. The level of granularity of the model should be adaptable to the safety relevance of the part under analysis. Modeling has attracted a substantial effort from research and practice in system engineering. In spite of quality and effective development processes, many system faults are traced back to high-level requirements. This has motivated the increasing use of modeling in system engineering. The aim of modeling is twofold. On the one hand modeling contributes towards correctness and completeness of system requirements. On the other hand modeling supports validation and verification activities. The overall goal of modeling is mainly to reduce the gap between system requirements and design. The requirements-design gap represents a major source of (requirements) changes. Although this gap is one of the sources of requirements changes, research on (requirements) evolution clearly points out other origins of changes (PROTEUS, 1996). Modeling tackles two main issues. The first is that translations from requirements to design are error-prone. The second is that stakeholders (e.g., system users, system engineers, etc.) have often contradicting understandings about which system. These problems have motivated the blossom of many modeling methodologies and languages, e.g., UML (Rumbaugh, Jacobson, and Booch, 1999), used in practice.

Modeling incorporates design concepts and formalities into system specifications. This enhances our ability to assess safety requirements. For instance, *Software Cost Reduction* (SCR) consists of a set of techniques for designing software systems (Heitmeyer, 2002; Hoffman and Weiss, 2001). The SCR techniques support the construction and evaluation of requirements. The SCR techniques use formal design techniques, like tabular notation and information hiding, in order to specify and verify requirements. According to information hiding principles, separate system modules have to implement those system features that are likely to change. Although module decomposition reduces the cost of system development and maintenance, it provides limited support for system evolution. *Intent Specifications* provide another example of modeling that further supports the analysis and design of evolving systems (Leveson, 2000). Intent Specifications extend over three dimensions. The vertical dimension represents the intent and consists of five hierarchical levels[2]. Along the horizontal dimension, the Intent

[2] Level 1, system purpose; Level 2, system principles; Level 3, blackbox behavior; Level 4, design representation; Level 5, physical representation or code. Note that a recent version of Intent Specifications introduces two additional levels: Level 0 and Level 6. Level 0, the management level, provides a bridge from the contractual obligations and the management planning needs to the high-level engineering design plans. Level 6, the system operations level, includes information produced during the actual operation of the system.

Specifications decompose the whole system in heterogeneous parts: Environment, Operator, System and Components. The third dimension, Refinement, further breaks down both the Intent and Decomposition dimensions into details. Each level provides rationale (i.e., the intent or "why") about the level below. Each level has mappings that relate the appropriate parts to the levels above and below it. These mappings provide traceability of high-level system requirements and constraints down to physical representation level (or code) and vice versa. In general, the mappings between Intent levels are many-to-many relationships. In accordance with the notion of semantic coupling, Intent Specifications support strategies (e.g., eliminating tightly coupled, many-to-many, mappings or minimizing loosely coupled, one-to-many, mappings) to reduce the cascade effect of changes. Although these strategies support the analysis and design of evolving systems, they provide limited support to understand the evolution of high-level system requirements[3]. The better is our understanding of system evolution; the more effective are design strategies. That is, understanding system evolution enhances our ability to inform and drive design strategies. Hence, evolution-informed strategies enhance our ability to design evolving systems.

Modeling methodologies and languages advocate different design strategies. Although these strategies support different aspects of software development, they originate in a common *Systems Approach*[4] to solving complex problems and managing complex systems. In spite of common grounds, modeling methodologies and languages usually differ in the way they interpret the relationships among heterogeneous system parts (e.g., hardware components, software components, organizational components, etc.). A common aspect is that models identify the relations between the different system parts. On the one hand these relations constrain the system behavior (e.g., by defining environmental dependencies). System (architectural) design partially captures these relations. On the other hand they are very important for system management and design. Among the different relations over heterogeneous system parts and hierarchical levels is *Traceability*. Although traceability supports management, traceability often faces many issues in practice. In particular, traceability faces evolution.

Research and practice in system engineering highlight critical issues. Among these issues evolution affects many aspects of the system life cycle. Unfortunately, most methodologies provide limited support to capture and understand system evolution. This is often because the underlying hypotheses are often unable to capture system evolution. Although requirements serve as basis for system production, development activities (e.g., system design, testing, safety analysis, deployment, etc.) and system usage feed back system requirements. Thus system production as a whole consists of cycles of discoveries and exploitations. The different development processes (e.g., V model, Spiral model, etc.) diversely capture these discover-exploitation cycles, although development processes constrain any exploratory approach that investigates system evolution. Thus system-engineering methodologies mainly support strategies that consider changes from a management viewpoint. In contrast, system changes, like the ones occurring in the ATM System, are emerging behaviors of combinations of development processes, products and organizational aspects.

[3] Leveson in (Leveson, 2000) reports the problem caused by Reversals in TCAS (Traffic Alert and Collision Avoidance System): *"About four years later the original TCAS specification was written, experts discovered that it did not adequately cover requirements involving the case where the pilot of an intruder aircraft does not follow his or her TCAS advisory and thus TCAS must change the advisory to its own pilot. This change in basic requirements caused extensive changes in the TCAS design, some of which introduced additional subtle problems and errors that took years to discover and rectify."*

[4] *"Practitioners and proponents embrace a holistic vision. They focus on the interconnections among subsystems and components, taking special note of the interfaces among various parts. What is significant is that system builders include heterogeneous components, such as mechanical, electrical, and organizational parts, in a single system. Organizational parts might be managerial structures, such as a military command, or political entities, such as a government bureau. Organizational components not only interact with technical ones but often reflect their characteristics. For instance, a management organization for presiding over the development of an intercontinental missile system might be divided into divisions that mirror the parts of the missile being designed."*, INTRODUCTION, p. 3, (Hughes and Hughes, 2000).

Limitations: Conventional safety analyses are deemed acceptable in domains such as the nuclear or the chemical sector. Nuclear or chemical plants are well-confined entities with limited predictable interactions with the surroundings. In nuclear and chemical plants design stresses the separation of safety related components from other plant systems. This ensures the independence of failures. Therefore, in these application domains it is possible to identify acceptable tradeoffs between completeness and manageability during the definition and identification of the system under analysis. In contrast, ATM systems operate in open and dynamic environments. Hence, it is difficult to identify the full picture of system interactions in ATM contexts. In particular:

- There is a complex interaction between aircrafts' controls and ATM safety functions. Unfortunately, this complex interaction may give rise to catastrophic failures. Hence, failure separation (i.e., understanding the mechanisms to enhance failure independence) would increase the overall ATM safety.
- Humans (Flight Safety Foundation, 2003; Pasquini and Pozzi, 2004) using complex language and procedures mediate this interaction. Moreover, most of the final decisions are still demanded to humans whose behavior is less predictable than that of automated systems. It is necessary further to understand how humans use external artifacts (e.g., tools) to mediate this interaction. Moreover, this will allow the understanding of how humans adopt technological artifacts and adapt their behaviors in order to accommodate ATM technological evolution. Unfortunately, the evolution of technological systems often corresponds to a decrease in technology trust affecting work practice.
- Work practice and systems evolve rapidly in response to demand and a culture of continuous improvements. A comprehensive account of ATM systems, moreover, will allow the modeling of the mechanisms of evolution. This will enhance strategies for deploying new system configurations or major system upgrades. On the one hand modeling and understanding system evolution support the engineering of (evolving) ATM systems. On the other hand modeling and understating system evolution allow the communication of changes across different organizational levels. This would enhance visibility of system evolution as well as trust in transition to operations.

Capturing Emerging Complex Interactions

Heterogeneous engineering[5] provides a different perspective that further explains the complex interaction between system (specification) and environment. Heterogeneous engineering considers system production as a whole. It provides a comprehensive account that stresses a holistic viewpoint, which allows us to understand the underlying mechanisms of evolution of socio-technical systems. Heterogeneous engineering involves both the systems approach (Hughes and Hughes, 2000) as well as the social shaping of technology (MacKenzie and Wajcman, 1999). On the one hand system engineering devises systems in terms of components and structures. On the other hand engineering processes involve social interactions that shape socio-technical systems. Hence, stakeholder interactions shape socio-technical systems. Heterogeneous engineering is therefore convenient further to understand engineering processes.

The most common understanding in system engineering considers requirements as goals to be discovered and design solutions as separate technical elements. Hence system engineering is reduced to be an activity where technical solutions are documented for given goals or problems. Differently according to heterogeneous engineering, system requirements specify mappings between problem and solution spaces. Both spaces are socially constructed and negotiated through sequences of mappings between solution spaces and problem spaces (Bergman, King, and Lyytinen, 2002; 2002a). Therefore, system requirements emerge as a set of consecutive solution spaces justified by a problem space of concerns to stakeholders. Requirements, as mappings between socio-technical solutions and problems, represent an account of the history of socio-technical issues arising and being solved within industrial settings.

[5] *"People had to be engineered, too - persuaded to suspend their doubts, induced to provide resources, trained and motivated to play their parts in a production process unprecedented in its demands. Successfully inventing the technology, turned out to be heterogeneous engineering, the engineering of the social as well as the physical world."*, p. 28, (MacKenzie, 1990).

The formal extension of these mappings (or solution space transformations) identifies a framework to model and capture evolutionary system features (e.g., requirements evolution, evolutionary dependencies, etc.) (Felici, 2004). The resulting framework is sufficient to interpret system changes. Therefore, the formal framework captures how design solutions evolve through subsequent releases. Hence, it is possible to define system evolution in terms of sequential solution space transformations. Moreover, it is possible to capture evolution at different abstraction levels with diverse models. This defines evolutionary cycles of iterations in the form: solutions, problems and solutions. This implies that engineering processes consist of solutions searching for problems, rather than the other way around (that is, problems searching for solutions). This holistic viewpoint of systems allows us to understand the underlying mechanisms of evolution of socio-technical systems, like the ATM System.

Capturing cycles of discoveries and exploitations during system design involves the identification of mappings between socio-technical solutions and problems. The proposed framework exploits these mappings in order to construct an evolutionary model that will inform safety analyses of ATM systems. Figure 2 shows the proposed framework, which captures these evolutionary cycles at different levels of abstraction and on diverse models. The framework consists of three different hierarchical layers: *System Modeling Transformation (SMT)*, *Safety Analysis Modeling Transformation (SAMT)* and *Operational Modeling Transformation (OMT)*.

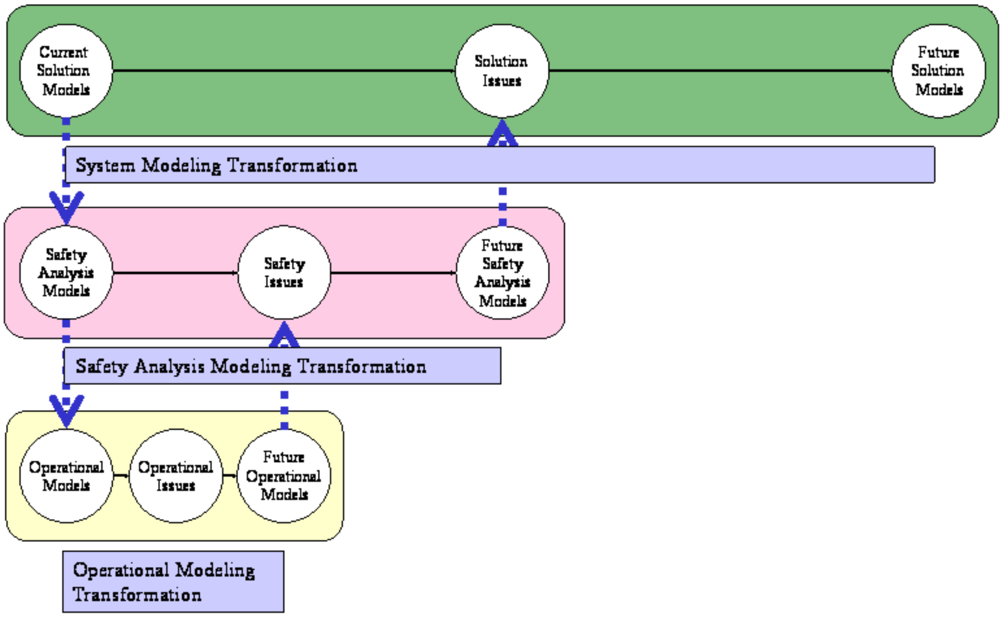
Current Solution Models
Solution Issues
Future Solution Models
System Modeling Transformation
Safety Analysis Models
Safety Issues
Future Safety Analysis Models
Safety Analysis Modeling Transformation
Operational Models
Operational Issues
Future Operational Models
Operational Modeling Transformation

Figure 2 - A framework for modeling evolutionary safety analyses

The SMT layer captures how solution models evolve in order to accommodate design issues or evolving requirements. Therefore, an SMT captures system requirements as mappings between socio-technical solutions and problems. This allows the gathering of changes into design solutions. That is, it is possible to identify how changes affect design solution. Moreover, this enables sensitivity analyses of design changes. In particular, this allows the revision of safety requirements and the identification of hazards due to the introduction of a new system. Therefore, the SMT supports the gathering of safety requirements for evolving systems. That is, it supports the main activities occurring during the top-down iterative process FHA in the SAM methodology (EUROCONTROL, 2004). The FHA in the SAM methodology then initiates another top-down iterative approach, i.e., the PSSA. Similarly, the framework considers design solutions and safety objectives as input to Safety Analyses. Safety analyses assess whether the proposed design solution satisfies the identified safety objectives. This phase involves different methodologies (e.g., Fault Tree Analysis, HAZOP, etc.) that produce diverse (system) models. System usage or operational trials may give rise to unforeseen safety issues that invalidate (part of) safety models. In order to take into account these issues, it is necessary to modify safety analyses. Therefore, safety analysis models evolve too. SAMT, the second layer of the framework, captures how safety analysis models evolve in order to accommodate raising safety issues. Although design models serve as a basis for safety models, they provide limited supports to capture unforeseen system interactions. Therefore, SAMT supports those activities involved in the PSSA process of the SAM methodology (EUROCONTROL, 2004). Note that although the SAM methodology stresses that both FHA and PSSA are iterative process, it provides little supports to manage process iterations as well as system evolution in terms of design solution and safety requirements. The framework supports these evolutionary processes.

Finally, operational models (e.g., structured scenarios, patterns of interactions, structured procedures, workflows, etc.) capture heterogeneous system dynamics. Unfortunately, operational profiles often change with system usage. For instance, system users often refine procedures in order to integrate different functionalities or to accommodate system failures. OMT, the bottom-layer of the framework, captures how operational models change in order to accommodate issues arising. The evolution of operation models informs safety analyses of new hazards. Therefore, OMT supports the activities involved in the SSA process of the SAM methodology.

Discussion and Conclusions

The proposed framework addresses three main points in order effectively to support evolutionary safety analyses. Firstly, the model questions the system boundaries and the required level of details. These aspects considerably vary from design models to risk analysis models, since system parts that need to be specified in details for the design may be much less relevant from a safety point of view. The typical drawback experienced in most cases is that resources for risk analysis may be consumed in investigating detailed aspects of every system part, instead of trying to identify unknown risks that may be related to elements not central in the design model. Furthermore it is often the case that system boundaries can be more neatly defined in respect to the design objectives, whilst risk analysis often requires the adoption of a larger focus. All the recent major incidents occurred in the civil aviation domain proved to stem from unexpected interactions from a large variety of elements, differently located in space and time. Those elements were often judged as outside of the system boundaries (or outside of normal operating conditions) when safety analysis has been conducted. For instance, the investigation report (BFU, 2004) of the accident between two aircrafts highlights that although individual ATM systems and procedures work properly, the ATM socio-technical interactions may, unfortunately, result in a catastrophic event.

The second point directly addresses these unexpected interactions between system elements as main source of incidents. Best practices and standards in safety analysis prescribe that mutual impact between different risks be analyzed. A system model is a key support to perform this task effectively, but the possible interactions need to be represented explicitly. On the contrary, models defined for design purposes usually outline the relationship between system elements by a functional (or physical) decomposition. In all the cases when design models are exploited for the safety analysis, the functional decomposition principle many unduly provide the structure for the analysis of incident causal dynamics (Johnson, 2003; Leveson, 2004), thus failing to acknowledge their different underlying nature. Furthermore, a correct model should not only ensure that interactions and mutual impact between different risks be analyzed, but also outline interactions between everyday productive processes in “normal operating conditions”, since risk factors are likely to interact along these lines.

The third characteristic of the model refers to the possibility of effective re-use of (part of) the model to inform other safety analyses. This would ensure that part of the safety feedback and experience related to a system can be beneficial when introducing major changes to the current system or when developing new similar systems. In the same way, the effective reuse of the model would result in safety analyses that have better means to achieve a good balance between exhaustiveness and costs, as findings of closely related analysis could be easily considered.

In order realistically and cost-effectively to realize the ATM 2000+ Strategy, systems from different suppliers will be interconnected to form a complete functional and operational environment, covering ground segments and aerospace. Industry will be involved as early as possible in the life cycle of ATM projects. EUROCONTROL manages the processes that involve the definition and validation of new ATM solutions using Industry capabilities (e.g., SMEs). In practice, safety analyses adapt and reuse system design models (produced by third parties). Technical, organizational and cost-related reasons often determine this choice, although design models are unfit for safety analysis. Design models provide limited support to safety analysis, because they are tailored for system designers. The definition of an adequate model and of an underlying methodology for its construction will be highly beneficial for whom is performing safety analyses. As stated before, currently the model definition phase cannot be properly addressed as an integral part of safety analysis, mostly because of limited costs and resources. This paper is concerned with problems in modeling ATM systems for safety analysis. The main objective is to highlight a model specifically targeted to support safety analysis of ATM systems. Moreover, the systematic production of safety analysis (models) will decrease the cost of conducting safety analyses by supporting reuse in future ATM projects.

Acknowledgments: I would like to thank Alberto Pasquini and Simone Pozzi for their information about the ATM domain. This work has been supported by the UK EPSRC Interdisciplinary Research Collaboration in Dependability, DIRC - http://www.dirc.org.uk - grant GR/N13999.

References

Bergman, M., King, J.L., and Lyytinen, K. (2002). Large-scale requirements analysis as heterogeneous engineering. Social Thinking - Software Practice, pages 357-386.

Bergman, M., King, J.L., and Lyytinen, K. (2002a). Large-scale requirements analysis revisited: The need for understanding the political ecology of requirements engineering. Requirements Engineering, 7(3):152-171.

BFU (2004). Investigation Report, AX001-1-2/02.

Enders, J.H., Dodd, R.S., and Fickeisen, F. (1999). Continuing airworthiness risk evaluation (CARE): An exploratory study. Flight Safety Digest, 18(9-10):1-51.

EUROCONTROL (2001). EUROCONTROL Airspace Strategy for the ECAC States, ASM.ET1.ST03.4000-EAS-01-00, 1.0 edition.

EUROCONTROL (2001a). EUROCONTROL Safety Regulatory Requirements (ESARR). ESARR 4 - Risk Assessment and Mitigation in ATM, 1.0 edition.

EUROCONTROL (2003). EUROCONTROL Air Traffic Management Strategy for the years 2000+.

EUROCONTROL (2004). EUROCONTROL Air Navigation System Safety Assessment Methodology, 2.0 edition.

Felici, M. (2004). Observational Models of Requirements Evolution. PhD thesis, Laboratory for Foundations of Computer Science, School of Informatics, The University of Edinburgh.

Flight Safety Foundation (2003). The Human Factors Implication for Flight Safety of Recent Developments in the Airline Industry, (22)3-4 in Flight Safety Digest.

Heitmeyer, C.L. (2002). Software cost reduction. In John J. Marciniak, editor, Encyclopedia of Software Engineering. John Waley & Sons, 2nd edition.

Hoffman, D.M. and Weiss, D.M., editors (2001). Software Fundamentals: Collected Papers by David L. Parnas. Addison-Wesley.

Hughes, A.C. and Hughes, T.P., editors (2000). Systems, Experts, and Computers: The Systems Approach in Management and Engineering, World War II and After. The MIT Press.

Johnson, C.W. (2003). Failure in Safety-Critical Systems: A Handbook of Accident and Incident Reporting. University of Glasgow Press, Glasgow, Scotland.

Laprie, J.-C. et al (1998). Dependability handbook. Technical Report LAAS Report no 98-346, LIS LAAS-CNRS.

Leveson, N. (2004). A new accident model for engineering safer systems. Safety Science, 42(4):237-270.

Leveson, N.G. (2000). Intent specifications: An approach to building human-centered specifications. IEEE Transactions on Software Engineering, 26(1):15-35.

Leveson, N.G. (2005). SAFEWARE: System Safety and Computers. Addison-Wesley.

MacKenzie, D.A. (1990). Inventing Accuracy: A Historical Sociology of Nuclear Missile Guidance. The MIT Press.

MacKenzie, D.A. and Wajcman, J., editors (1999). The Social Shaping of Technology. Open University Press, 2nd edition.

Matthews, S. (2002). Future developments and challenges in aviation safety. Flight Safety Digest, 21(11):1-12.

Overall, M. (1995). New pressures on aviation safety challenge safety management systems. Flight Safety Digest, 14(3):1-6.

Pasquini, A. and Pozzi, S. (2004). Evaluation of air traffic management procedures - safety assessment in an experimental environment. Reliability Engineering & System Safety, 2004.

PROTEUS (1996). Meeting the challenge of changing requirements. Deliverable 1.3, Centre for Software Reliability, University of Newcastle upon Tyne.

Ranter, H. (2003). Airliner accident statistics 2002: Statistical summary of fatal multi-engine airliner accidents in 2002. Technical report, Aviation Safety Network.

Ranter, H. (2004). Airliner accident statistics 2003: Statistical summary of fatal multi-engine airliner accidents in 2003. Technical report, Aviation Safety Network.

Skyway (2004). Working towards a fully interoperable system: The EUROCONTROL overall ATM/CNS target architecture project (OATA). Skyway, 32:46-47.

Rumbaugh, J., Jacobson, I., and Booch, G. (1999). The Unified Modeling Language Reference Manual. Addison-Wesley.

Storey, N. (1996). Safety-Critical Computer Systems. Addison-Wesley.

van Es, G.W.H. (2001). A review of civil aviation accidents - air traffic management related accident: 1980-1999. Proceedings of the 4th International Air Traffic Management R&D Seminar, New-Mexico.