Structured Design of Safety Management Systems

by Steve Thomas

Introduction

The adoption of an explicit and systematic approach to safety by Air Traffic Management (ATM) service providers in Europe was motivated to a large extent by the emergence in the late 1990s of a series of safety regulatory requirements from EUROCONTROL, the European Organization for the Safety of Air Navigation. As a result, air navigation service providers were faced with the need to introduce unfamiliar new activities within their organizations via the development and deployment of a formal Safety Management System (SMS). The SMS would have to be fully compliant with regulatory requirements, but also be practicable in terms of the resources it demands during operation, and be compatible with other business processes. This challenge is common to providers of products or services in any regulated, safety-related industry but leaves the question of how an organisation should go about solving it.

The approach in this paper recognises that, although it is non-physical in form, an SMS is nevertheless a system. Its development therefore can, and should, exploit engineering design principles within a structured approach. As a consequence, there is a need to represent SMS requirements and design at different levels of decomposition using suitable notations. In view of similarities between an SMS and real-time software, the paper proposes an established software design notation for this purpose.

Design Hierarchy

The regulatory compliance aspect can be viewed as an information transformation problem. The regulatory requirements must be analysed and interpreted, and then be progressively decomposed and elaborated until a much larger set of interconnected low-level actions has been defined which is suitable for direct execution by the safety 'actors' within the service provider's organization. The safety actors comprise those operational, technical and management personnel who perform the activities defined within the scope of the SMS.

The intermediate representations within this transformation hierarchy may be referred to as 'Safety Policy' and 'Safety Principles' which, like the regulatory requirements, are usually not directly actionable by safety actors. The derived low-level actions, which can be directly traced back to regulatory requirements via the Safety Policy and Safety Principles, are collected together in a set of procedures, with the actors being defined via a safety organization whose members have specific accountabilities and responsibilities. A typical logical model for the Safety Management Manual (SMM), which can be thought of as the operational requirements part of the SMS, is shown in Figure 1 below.

Some of the design choices to be considered when deciding on the structure of this logical model and the contents of each block are:

- The number of transformation levels that are needed, the amount of detail at each level, and the ownership of each block;
- The optimum grouping of the low-level actions within and among the set of procedures to achieve the best 'size' for the corresponding safety processes, and to minimise their interdependencies;

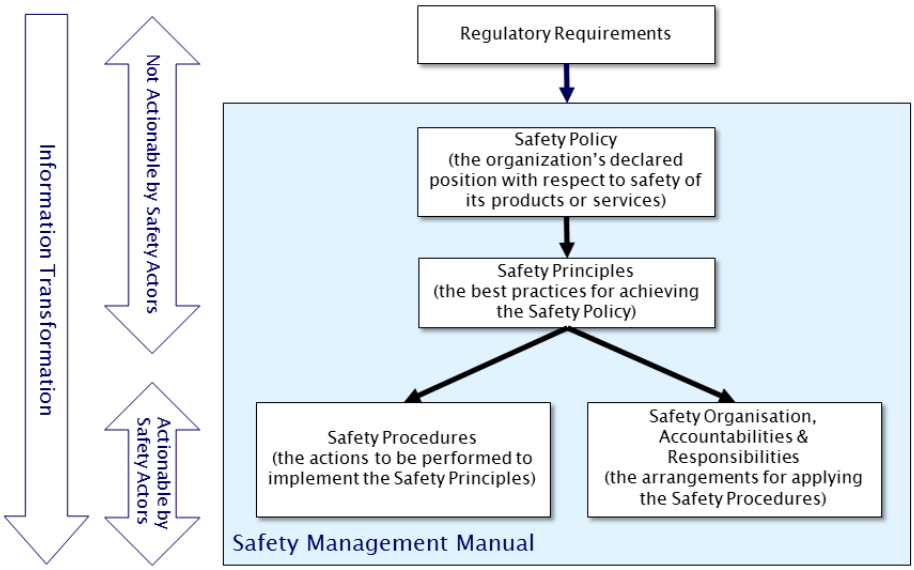


Figure 1: SMM Logical Model

- Balancing prescription (which makes procedures actionable and auditable) against objectives (which permit innovation and flexibility) when defining the safety activities;
- Defining who are the intended safety actors;
- Defining the required actions in a way that is consistent with the capabilities of the intended actors;
- The distribution of information about how to conduct safety activities in and among procedures (mandatory action), guidance (recommended action) and training;
- The integration of pre-existing business processes with the SMS.

All of these are variables influencing the SMS design. It follows that arriving at the 'best' SMS design is an optimization problem as for any technical system design exercise. However, since the scope of an SMS can encompass existing (or modified) business processes in addition to many new safety processes, in practice it will inevitably comprise a large set of processes and interactions even when optimised. Therefore, a way of representing SMS processes, their boundaries, interactions and their relationship to actors is needed in order to facilitate investigation of the different SMS design options around process granularity, interfaces and integration.

It should be mentioned at this point that, in addition to the need for regulatory compliance, there will usually exist other external influences on the initial and ongoing development of an SMS. These can include the desire to satisfy safety standards, and incremental developments as a result of feedback from safety actors, audits, and *ad hoc* suggestions for safety improvement. The SMS should contain specific procedures for capturing this information in order to trigger and inform any necessary SMS design changes. Equally, it should contain a procedure for assessing the impact of any changes to the regulatory requirements. Consequently, a complete SMS will contain procedures which can effect changes to the SMS design itself.

SMS Characteristics

To implement these design concepts there must be, as a minimum, a means to trace and justify the relationship between the low-level safety actions and the regulatory requirements in order to verify compliance. This implies the use of an information hierarchy in which the different levels of definition, as described previously, are mutually consistent. Furthermore, the low-level actions in the procedures need to be articulated as requirements to be enacted by the (human) safety actors.

It is also apparent that an SMS is a real-time system, albeit consisting of human activity (even when the SMS is supported by tools). Such activity has associated inputs, outputs, triggers and time constraints. Moreover, since the SMS encompasses all the diverse aspects of safety management, it should comprise processes that are sufficiently decoupled to allow them to run concurrently and asynchronously.

Therefore, it can be seen that there are some parallels between an SMS and a software-based system, and between the means of their development. For example, there is a need for progressively elaborated representations in the requirements/design part of the development lifecycle (from user requirements down to source code in the case of software-based systems), and a need to model their behaviour (including real-time aspects). Once these similarities are recognised, one can go on to explore whether established methods used in software engineering could be beneficially exploited for the development of an SMS.

Structured Design

This paper proposes the use of the structured design technique from software engineering, which can potentially aid SMS development by enabling the visualisation, manipulation, and optimization of its data processing elements. One established technique that can readily be applied is the use of data flow diagrams (DFD) with real-time extensions [1], in which processes and their interactions can be diagrammatically represented at different levels of decomposition using the basic symbols shown in Figure 2 below.

Although the original DFD notation has been in existence since the 1970s, hitherto it has been used for representing human activities very rarely [2][3], and there is no published work on its use for development of a complete SMS in any industry sector. Therefore, while recognising that a number of other process modelling notations could have been used, it was decided to adopt the technique for the creation of an SMS for an Air Traffic Control centre in order to judge its merits. In particular, it seemed to offer the means to better manage the growth of the SMS and retain traceability of requirements as its development progressed. It was applied in a pragmatic way, without tool support, by personnel who were not experts in software engineering. The result was a fully-coherent SMS design, depicted as a DFD process model, in which all process inputs, outputs and triggers were represented and traceable.

It should be noted at this point that due to the non-physical nature of an SMS, there are some important differences between the use of DFDs for design of an SMS, and their use for design of software-based systems. One is that safety actors are internal to the SMS processes being represented rather than being part of the environment external to the system, as in the case of equipment. Secondly, since the documented inputs and outputs from safety processes likewise are considered to be an integral part of the SMS, DFD terminators can appear in and among the DFDs, whereas for software they only appear in the top-level DFD (aka

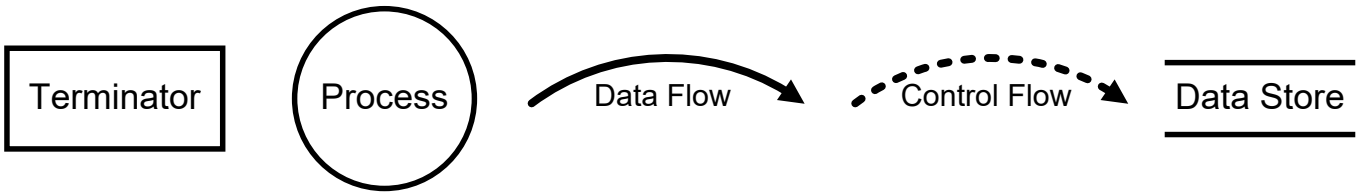


Figure 2: Data Flow Diagram Notation

the context diagram). Therefore, the DFDs herein are an adaptation of the pure notation found necessary to fit the specific application.

Safety Procedures

The generic relationship between the SMS procedures and the DFDs is shown in Figure 3. This shows how elements of the DFD process model are captured in the corresponding procedure. Depending on the SMS design, the procedure might capture the input and output objects (Object_1 to Object_3 and Database_X in the example) instead of the associated data flows (Data_1 to Data_6). For example, it may suffice for a procedure to mandate the document types to be used as inputs and outputs, rather than the individual items of data contained therein.

For the ATM application, the group of SMS processes which govern the reporting and assessment of safety occurrences is depicted in Figure 4. This represents a solution to the regulatory requirement known as ESARR2 [2], which mandates that all accidents and incidents in the airspace are reported and investigated. Investigation and reporting of operational and technical incidents affecting ATM service provision, but which have not led to an accident or incident in the airspace, is also required. These events are termed 'ATM-specific Occurrences'.

Figure 4 depicts the corresponding part of the SMS design in which the airspace accident/ incident investigation process is shown as 'OCC', with separate processes covering technical ATM-specific occurrences, non-technical ATM-specific occurrences, and external reporting. Visualisation has been enhanced by means of an outline colour for a DFD element denoting the source regulatory requirement document, and a fill colour denoting its ownership within the

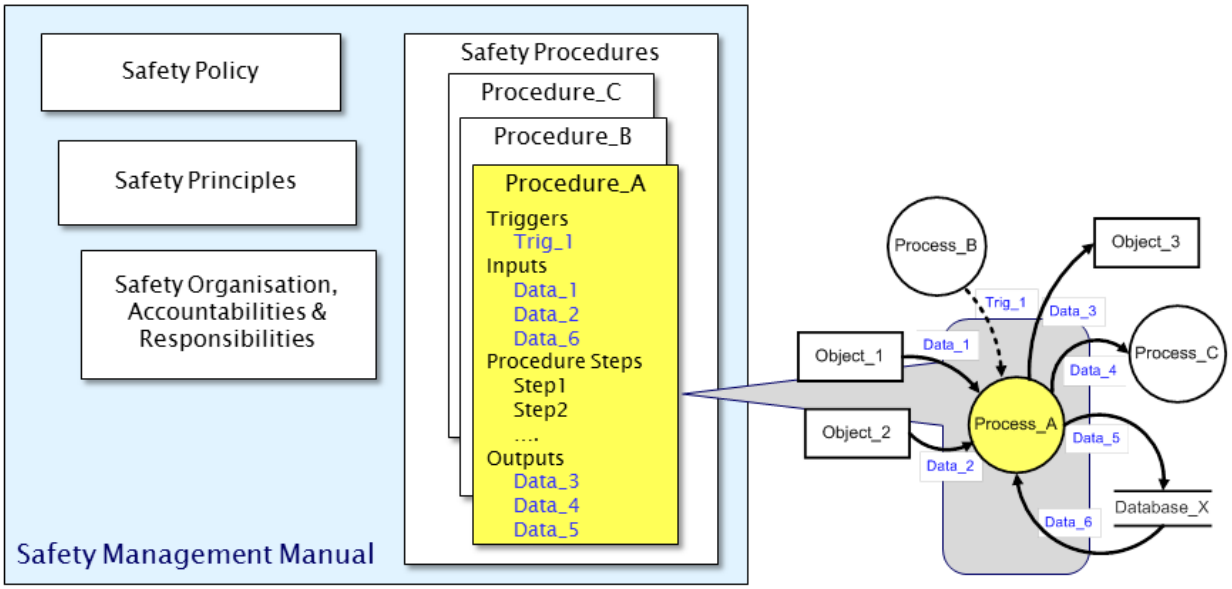


Figure 3: DFD Relationship to Procedures

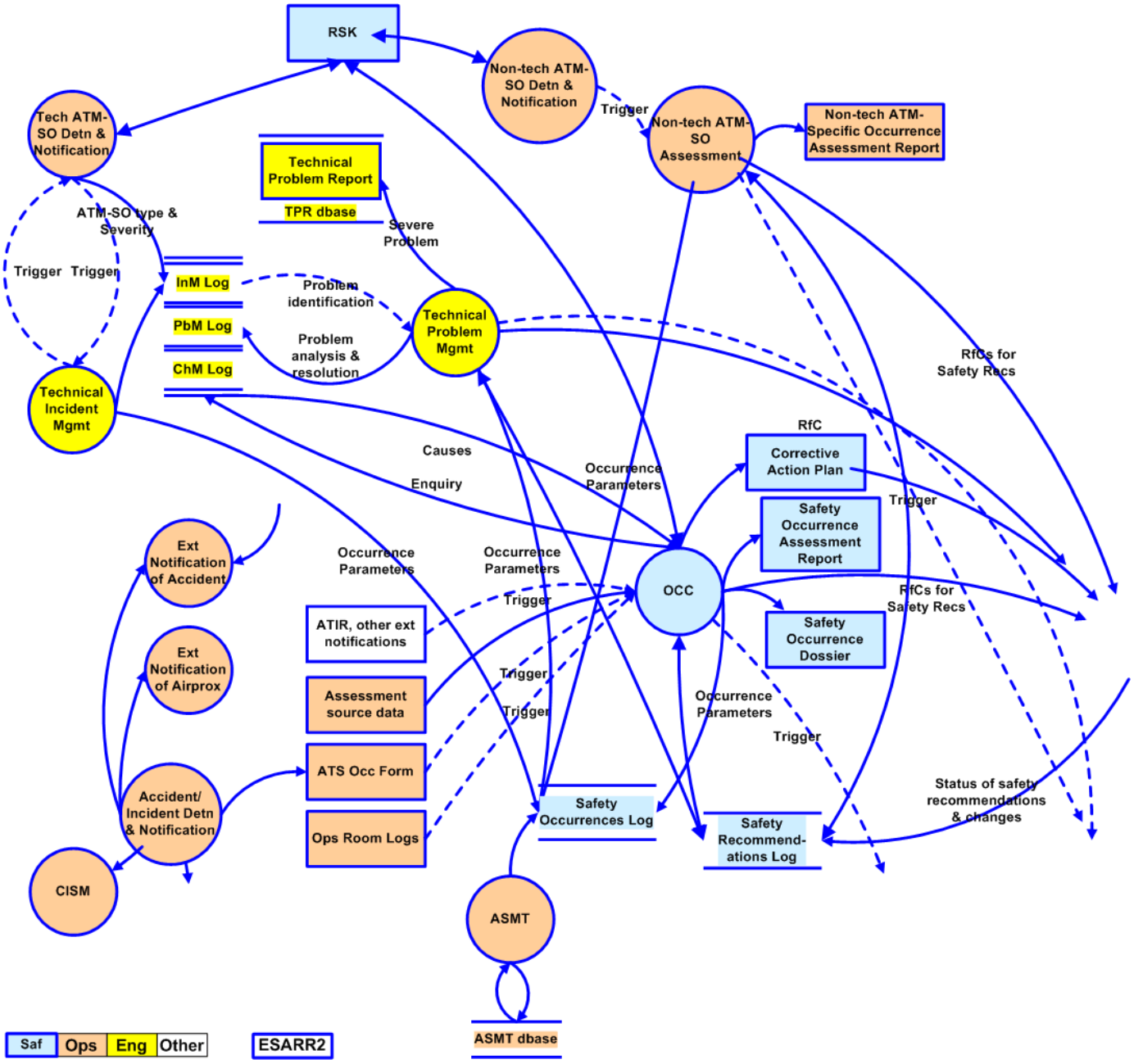


Figure 4: Safety Occurrences Process Group

organization (Safety, Operations, Engineering, Other). Flows which do not appear to terminate actually connect with other groups of processes, which have been masked out from Figure 4 for the sake of clarity.

Diagrams like this are indispensable for communication between SMS developers because they reveal exactly how each process fits into the overall picture and how its information flows are used. Safety data that is produced by a process but not consumed elsewhere can be easily identified, along with other structural inconsistencies. As implied by Figure 3 above, the DFD process model is documented outside of the SMM and is therefore invisible to the safety actors who need only base their actions on conventional, structured free-text procedures and guidance.

As well as safety activities conventionally described within procedures, it should also be recognised that meetings are processes in their own right. Therefore, they can explicitly be included in the SMS design

whether triggered by time (periodic meetings) or by events (ad hoc). Capturing meetings and their inputs/outputs in a DFD forces SMS developers and managers to consider their precise purpose and how they should connect to the rest of the SMS. In turn this can help in formulating their Terms of Reference, memberships and agendas. It will also reveal the precise means used for capturing important decisions from a meeting such as an 'approval to operate', and which other SMS process(es) subsequently use these as an input.

To illustrate these points, the ATM example in Figure 5 shows some of the important interfaces between a periodic Safety Committee meeting, the Safety Monitoring (MON) process, the Safety Surveys (SUR) process, and the regulator. This group of processes is part of the SMS design solution to a regulatory requirement known as ESARR3 [3], which defines the required components of an ATM service provider's SMS (broadly at the same level of abstraction as an SMS Safety Policy and Safety Principles). It can be seen that the Safety Committee makes use of a MON database, part of which comprises the Safety Occurrences Log established to satisfy ESARR2.

Safety Responsibilities

As well as process definitions, the SMS needs to allocate safety actions to recognisable entities within the organization. The actions should be meaningfully defined. Therefore, whereas a safety responsibility such as "the Head of Engineering shall ensure that the Safety Policy is implemented in the Engineering Department" is a worthy statement of intent and can easily be documented, it is not directly actionable (or auditable) because it is defined at too high a level (note 1). In fact, any responsibility statement phrased in the form "xxx shall ensure that yyy" could conceivably be discharged in a number of ways by the responsible person. This is because a number of subsidiary actions arising from the verb 'ensure' (such as 'plan', 'perform', 'verify', 'monitor') are implied. Collectively, these might satisfy the concept of 'ensuring', but have not been prescribed explicitly.

It is also conceivable for SMS developers to include less tangible responsibilities such as "an Air Traffic Controller shall have a pro-active attitude to Safety Management'" Although a desirable consequence of having an effectively operating SMS, again it is not necessarily directly actionable by the

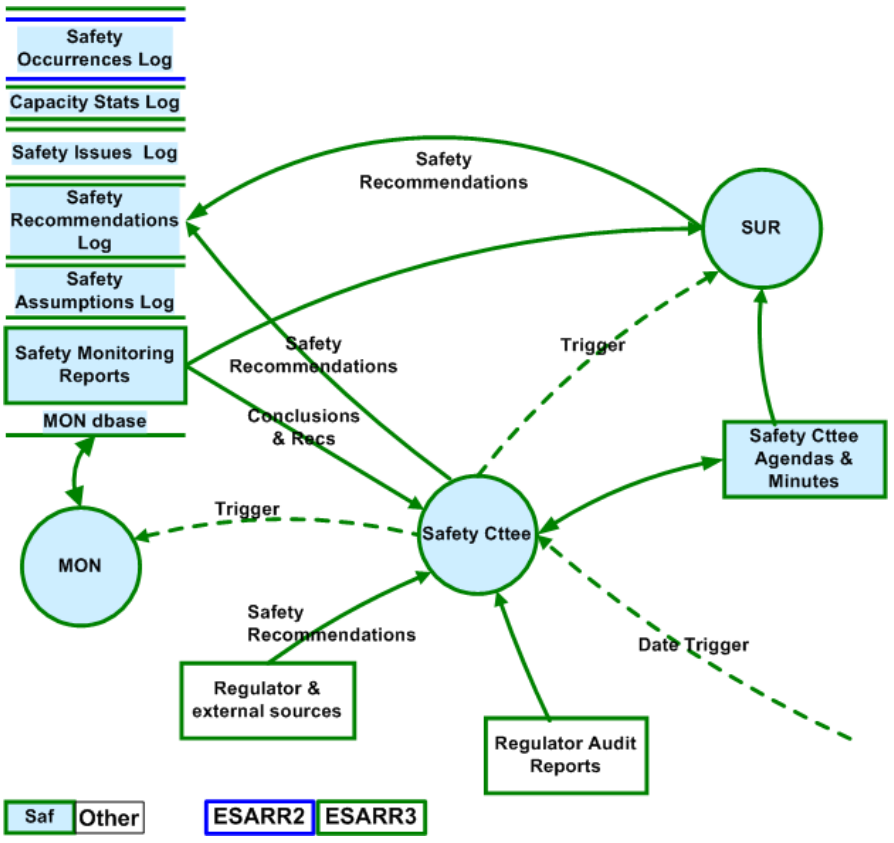


Figure 5: Safety Committee Process Group

responsible person, and may be difficult to verify by objective testing.

In order to have unambiguous, actionable, repeatable implementation of safety responsibilities, they should instead be stated ar a suitably low level of decomposition. The responsibilities, like the safety actions, therefore need to be elaborated (and usually delegated) in a way which satisfies these attributes.

Safety responsibilitles are in fact formal requirements for action. As such, they should as far as possible employ a common syntax, such as <actor><action><target> where all three components are fully traceable. This would provide the responsible person with clear pointers to the relevant SMS process or process input/output with which the responsibilities are implicated. A number of standard syntaxes can be employed to cover the range of human activity implied by the process model. As with the procedural elements in the SMM, it then becomes possible to derive (or at least to trace) the safety responsibilities from the process model, as illustrated in Figure 6.

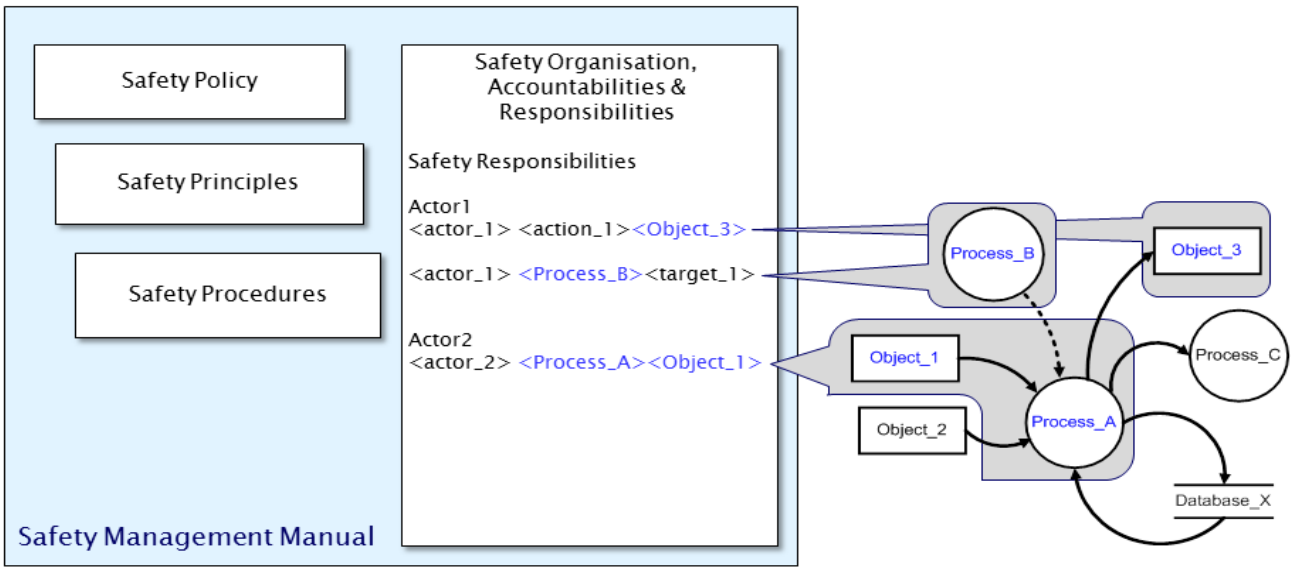
Examples of safety responsibilities using this approach are forms like: *<The Safety Engineer> <shall perform Functional Hazard Assessment of> <operational changes>* and *<The Head of Operations> <shall approve> <Safety Objectives for ATM System changes>* (see note 2 below).

Being able to identify explicit relationships between responsibilities and processes in this way allows the SMS developers to verify that there are no gaps or redundancies in either the process model or the responsibilities. or example, it might expose the fact that there is no process (or measurable outputs) defined in the SMS for the Air Traffic Controller's responsibilities to "have a proactive attitude to Safety Management".

Conclusion

This paper promotes the use of structured design

Figure 6: DFD Relationship to Responsibilities



concepts taken from software engineering for aiding the development of an SMS. Specifically, the use of DFDs ensures that the SMS comprises a properly connected set of processes, and the use of standard syntax for the formulation of safety responsibilities ensures proper connectivity between the safety actors and their required actions. This is illustrated using practical examples from ATM. The approach has the potential to yield the following benefits at various points in the lifecycle of an SMS within any industrial sector:

- Initial design of the SMS is represented in a visual, detailed way that allows interested parties to understand and comment on its structure.
- All levels of design can be visualized and their consistency verified because the complete hierarchy of 'process levels' from regulatory requirements down to the internal structure of individual SMS processes can be represented.
- Development of the SMS can be more easily managed. By representing and managing process interfaces, new safety processes and existing quality management processes can be correctly integrated.
- Validation that a developed SMS is complete and coherent can be performed retrospectively by capturing the design from the suite of SMS documentation, thus providing a means of independently assessing the SMS structure.
- Maintainability is enhanced because the implications on the rest of the SMS of modifying a given procedure can clearly be seen.

Notes

1. In effect, this requires the responsible person to develop their own Safety Principles and procedures.

2. The <> symbols are used here to delimit the three syntactic components and would not be documented in the SMM.

References

[1] Hatley, D. J. and Pirbai, I. A., *Strategies for Real-Time System Specification.* Dorset House Publishing, New York, 1987.

[2] Smith, I. C., 'The Maintenance of Computer Based Safety Systems' *IFAC Synposium on the Safety of Computer Control Systems (SAFECOMP'88): Safety Related Computers in an Expanding Market, 9-11 November 1988,* pp. 27-33.

[3] Riaz, Z., Edwards, D. J., Holt, G. D. and Thorpe, T., 'Data Flow Analysis of Plant and Equipment Health and Safety Management', *Journal of Engineering, Design and Technology* vol 9 no 2, July 2011, pp. 178-203.

[4] EUROCONTROL, ESARR 2, *Reporting and Assessment of Safety Occurrences in ATM,* Edition 3.0, 2 December 2009.

[5] EUROCONTROL, ESARR 3, *Use of Safety Management Systems by ATM Service Providers*, Edition 1.0, 17 July 2000.

Dr Steve Thomas is an Independent Consultant at Entity Systems Ltd. He can be contacted by email at steve@entitysystems.eu

Assessing Safety Culture - Where do I start?

by Vicky Gliddon

Safety culture is an important topic in safety-critical industries, where it has received a great deal of coverage. However, for many it can seem daunting to know what to do and where to start to understand the safety culture of their own organisation. Here, we provide advice from Greenstreet Berman Ltd's practical experience for an effective approach to conducting safety culture assessments, as well as tips for implementing targeted improvement following the assessment.