Project: COMPASS

Grant Agreement: 287829

Comprehensive Modelling for Advanced Systems of Systems

Traceable Engineering of Fault-Tolerant SoSs

COMPASS White Paper WP08

February 2014

Public Document

http://www.compass-research.eu

Authors:

Zoe Andrews, Claire Ingram, Richard Payne, Alexander Romanovksy, Newcastle University, UK
Jon Holt, Simon Perry, Atego, UK

Abstract:

Systems of systems (SoSs) are characterised by a challenging combination of continuous evolution, emergent behaviour and distributed, autonomous and independent constituents. The development of SoSs that can tolerate faults and harmful events is hampered by these and other complexities. Currently there is little in the way of methods or tools to help SoS developers to design fault-tolerant SoSs. In this white paper we introduce a structured approach for capturing the fault-tolerant aspects of an SoS at the architectural phase of design, the COMPASS fault modelling architectural framework (FMAF).

1. Introduction

Development of systems of systems (SoSs) that are capable of tolerating faults and other potentially harmful events (including failures of constituent systems (CSs)) is a daunting task. The characteristics of SoSs make the need for fault tolerance clear:

- There is a risk of unanticipated failures in independently managed CSs
- The distributed nature of the SoS can introduce communications faults
- There is dynamic change in CSs, and the potential for mismatches between CSs
- There is also a high risk of concurrent errors caused by error propagation or by reliance of several CSs on the same components/infrastructures
- It is not always clear to what extent we could recover CSs when we are recovering from an error in an SoS, or with which CSs the responsibility for recovery lies

Together, these sources of risk suggest that the introduction of fault tolerance into SoS designs requires an architectural model identifying SoS boundaries, CSs and their connectors as well as clearly recording responsibilities for error detection and recovery. There is a lack of methods and tools to help SoS engineers engage with other stakeholders and make explicit, informed and traceable choices about the fault assumptions, the fault tolerance strategies and the redundancy to be employed. For these reasons, we propose an architectural design approach for a fault-tolerant SoS that supports reasoning about faults and error propagation. The approach includes:

- a structured approach to capturing requirements of fault-tolerant SoSs
- a traceable mapping of fault tolerance requirements into SoS architectural designs
- an architectural framework that supports disciplined and reusable development of fault-tolerant architectures

We introduce the approach in this white paper and provide some pointers to more detailed examples published on behalf of the COMPASS project. To ensure a wider industrial acceptance our proposed solutions are developed for the Systems Modelling Language (SysML) (Holt and Perry 2008, OMG 2012) and are supported by industry-strength tools (we use Artisan Studio[1] for developing prototypes and experiments), although our approach is intended to be applicable independently of the selected modelling language. The COMPASS architectural approach will allow a reliability engineer to answer the question: are there faults identified in the requirements that haven't been taken into account in the architectural modelling?

2. Fault Modelling Architectural Framework

The COMPASS Fault Modelling Architectural Framework (FMAF) aims to address the need for a systematic approach to capturing fault tolerance and dependability aspects of SoSs. The FMAF particularly focuses on support for architectural modelling in SoSs, including:

- definition of faults, errors and failures
- identification of the causal chains of dependability threats (faults, errors and failures)
- identification of CSs (and the connections and interfaces between them) needed to tolerate faults
- identification of erroneous behaviour/recovery scenarios and processes
- behaviour description of processes in the presence of faults and recovery processes

[1] www.atego.com/products/artisan-studio/

These requirements for the FMAF are presented in the AF Context Viewpoint (AFCV), given in **Figure 1**. This view is used to define the requirements of an AF (in this case the FMAF) in the context of its stakeholders.

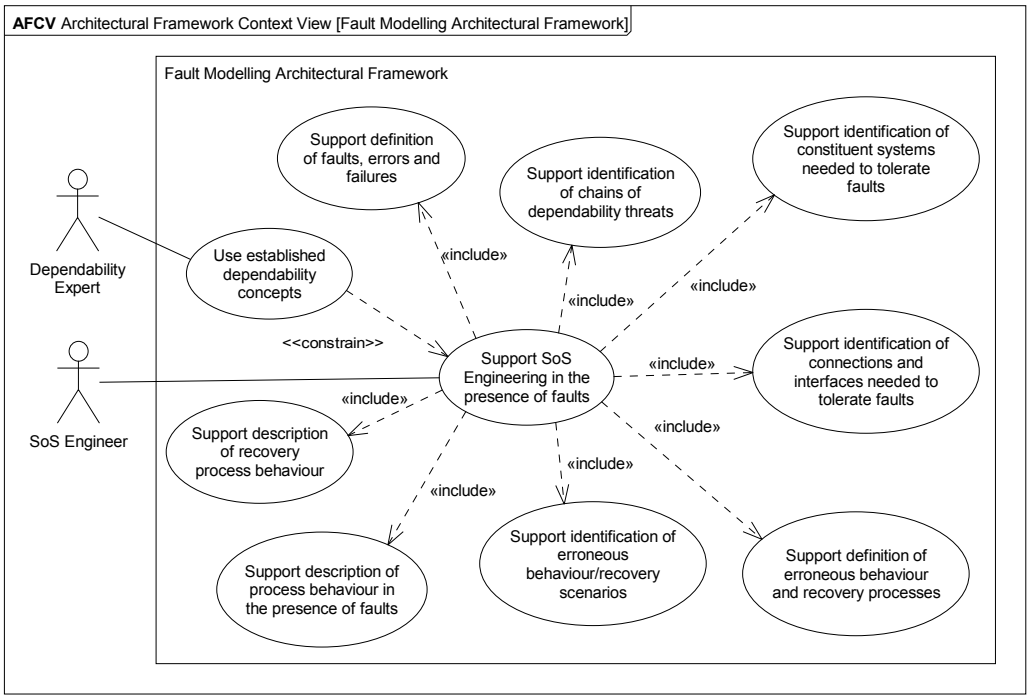

Figure 1 AF Context View showing the requirements for the FMAF

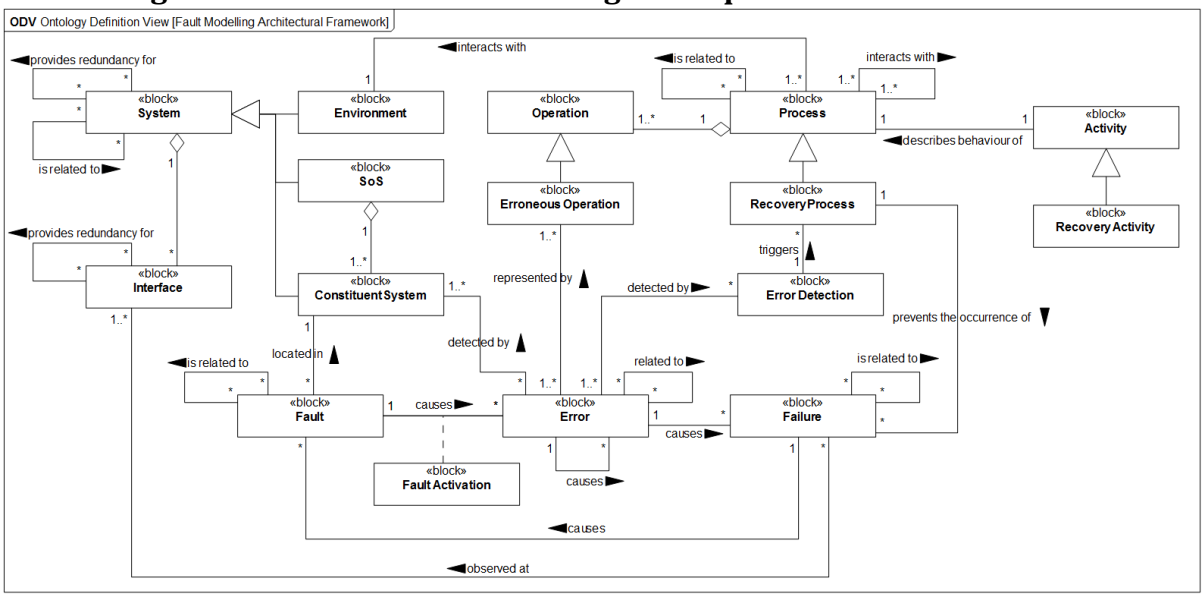

Figure 2 FMAF Ontology Definition View

The AFCV includes a requirement that the FMAF be consistent with established dependability concepts. To satisfy this we define an ontology (shown in Figure 2) for the FMAF (COMPASS D24.2) based on concepts identified in (Avizienis et al. 2004). Under this taxonomy:

- a *failure* is a deviation of the service provided by an SoS (or, at a different level of abstraction, a CS of the SoS) from expected (correct) behaviour; this results in incorrect behaviour visible at the boundary of the SoS (or the CS).

- An *error* is defined as the part of the SoS state that can lead to its subsequent service failure.
- The adjudged or hypothesized cause of an error is called a *fault*.

The focus of our work is on *fault tolerance*, preventing failures from arising in the presence of faults; this is achieved by detecting errors and conducting system *recovery* to remove the erroneous state and, if possible, faults.

Based on the context and requirements identified, a number of viewpoints have been defined that constitute the FMAF approach. The viewpoints are described informally in (Andrews, Fitzgerald et al. 2013) and are summarised in Table 1. These include structural viewpoints to define the faults, errors, failures in an SoS and their causal relationships as well as behavioural viewpoints that identify the behaviour of the SoS in the presence of identified faults. We include viewpoints to make clear the boundaries between CSs and their environment and causal chains through the SoS, as well as viewpoints modelling functional and failure states. We also include model elements to clearly identify where responsibility lies for detecting and recovering from errors. A complete set of models is presented in (COMPASS D24.2), in SysML; we present here a short selection of models.

Table 1 - Informal description of the FMAF viewpoints

	Name	**Description**
Structural Viewpoints	Fault/Error/Failure Definition	Define faults, errors and failures of the SoS. Faults, errors or failures may be generalised into abstract categories.
	Threats Chain	Identifies causal chains of faults, errors and failures, and relationships between these threats and the CSs.
	Fault Tolerance Structure	Shows the composition of the SoS with the required redundancy to tolerate a given fault.
	Fault Tolerance Connections	Shows connections and interfaces between CSs of the SoS with the required redundancy to tolerate a given fault. Includes all the CSs given in the respective *Fault Tolerance Structure View*.
	Name	**Description**
Behavioural Viewpoints	Erroneous/Recovery Processes	Identifies the processes of the SoS, including erroneous behaviour and any required recovery processes.
	Erroneous/Recovery Scenarios	Models behaviour in the presence of errors (with and without recovery) as scenarios. Shows erroneous behaviour propagation and recovery procedure triggers.
	Fault Activation	Defines process behaviour and identifies when faults may be activated, what happens after activation and where the error may be detected.
	Recovery	Defines the behaviour of the recovery procedures that are triggered once an error has been detected.

The FMAF, with its collection of viewpoints, aims to be the first step for thinking about faults in SoSs and designing dependability into the SoS. It necessarily makes abstractions to manage the complexity of the task. For example, fault activation is modelled as an event without explicitly defining how the fault is activated, and likewise the details of how errors are detected are also abstracted away by a similar usage of error detection events. This means that the same modelling approach is applicable to a wide variety of fault activation scenarios.

These details can of course be included in a modelling approach that applies the FMAF, but the way to do so has not (yet) been prescribed by the FMAF.

As an example, let us consider one viewpoint of the FMAF in more depth – the Threats Chain Viewpoint (TCV). The TCV supports the identification of causal chains of faults, errors and failures (see the Viewpoint Context Viewpoint (VCV) in Figure 3, which represents the requirements in context for a particular viewpoint). This includes the definition of possible causal chains between faults, errors and failures as well as identifying the location of faults, the CSs that can detect the errors, and the interfaces of the SoS at which failures may be observed.

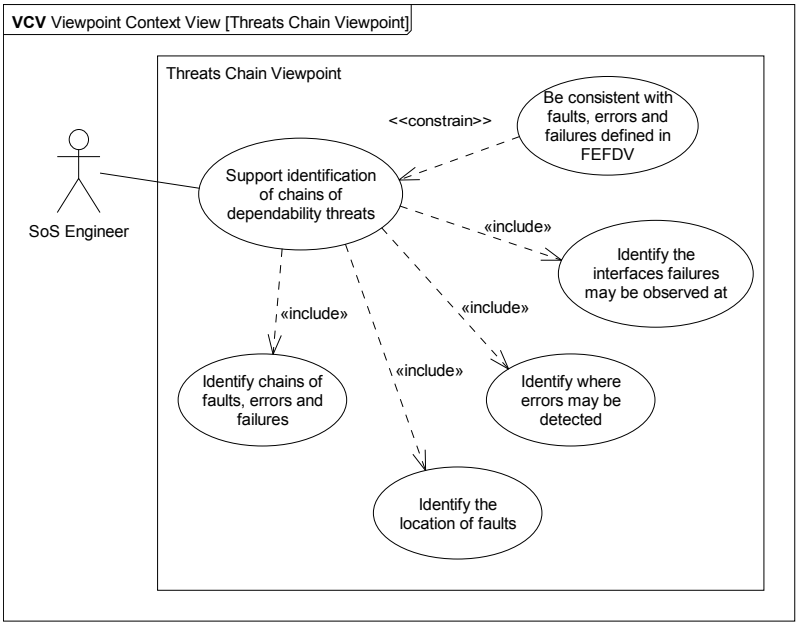

Figure 3 Viewpoint Context View for the FMAF Threats Chain Viewpoint

All faults, errors and failures shown in a TCV must be defined, i.e. included in a Fault/Error/Failure Definition View. A Viewpoint Definition Viewpoint (VDV) is given in Figure 4, showing the viewpoint elements of a TCV and their relationships. A single view in a model may implement more than one FMAF viewpoint. For example, recovery processes can and do fail, therefore fault activation and error detection may be modelled within a Recovery View, which would mean that the view is also a Fault Activation View.

3. Fault Tolerant Architectural Modelling Case Study

The COMPASS FMAF has been employed to model fault-tolerant aspects of different case studies. For a worked example of the FMAF using an emergency response SoS as a case study see (COMPASS D24.2).

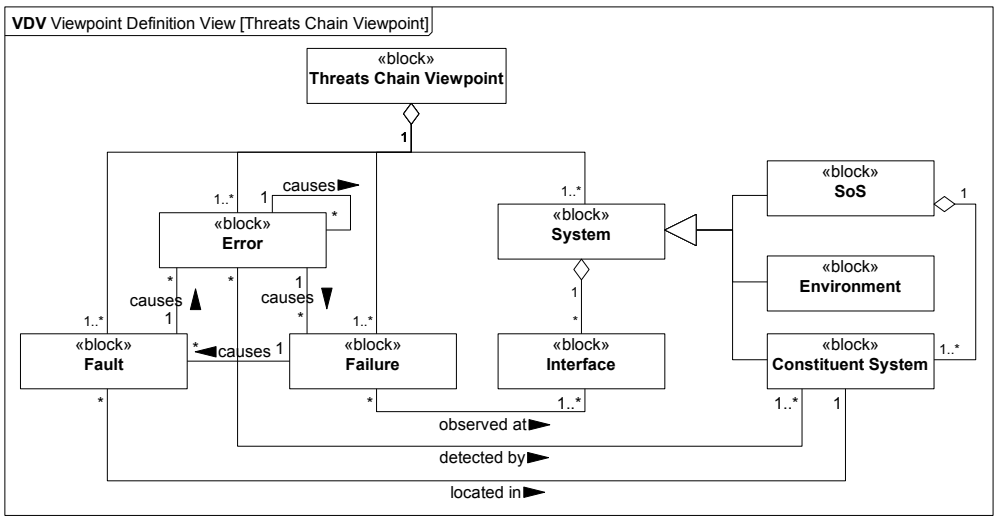

Figure 4. Viewpoint Definition View for the FMAF Threats Chain Viewpoint

4. Conclusion

In this white paper, we have provided a rationale for a structured approach to capturing the fault-tolerant aspects of an SoS at the architectural phase of design and introduced the FMAF, an architectural framework that supports disciplined and reusable development of fault-tolerant architectures. The FMAF describes a range of viewpoints to encompass faults, errors and failures in an SoS architectural model. These have been demonstrated with worked examples based on COMPASS case studies – for a worked example of a case study, see (COMPASS D24.2).

A SysML profile has also been developed (COMPASS D24.2), which specialises FMAF views and view elements to include additional information through the use of stereotypes (of underlying SysML model elements) and associated tags. This profile aims to provide support for ensuring model consistency and translation to external analysis tools and may be used in any SysML toolset with profiling support.

The work described here forms part of a larger COMPASS approach that also tackles fault analysis, including techniques for translating from SysML (as presented here) into HiP-HOPS[2].

References

Andrews, Z.; Fitzgerald, J.; Payne, R. and Romanovsky, A. 2013. "Fault Modelling for Systems of Systems" *in* Proceedings of the 11th International Symposium on Autonomous Decentralised Systems (ISADS 2013), pp. 59--66.

Avizienis, A.; Laprie, J.-C.; Randell, B. and Landwehr, C. 2004. "Basic Concepts and Taxonomy of Dependable and Secure Computing", *IEEE Transactions on Dependable and Secure Computing* 1, 11-33.

COMPASS D24.2. "Report on Timed Fault Tree Analysis – Fault modelling". COMPASS Deliverable D24.2. Technical Report, http://www.compass-research.eu/deliverables.html, 2013.

[2] A tool that supports safety analysis such as Fault Tree Analysis and Failure Modes and Effects Analysis of annotated architectural system models, see http://www.hip-hops.eu/.

Holt, J. D. and Perry, S. A. 2008. *SysML for Systems Engineering*. IET.

OMG 2012. "OMG Systems Modelling Language Version 1.3." Available: http://www.omg.org/spec/SysML/1.3 (Accessed June 2013).