Formal Architecture Description of Trustworthy Systems-of-Systems with SosADL

Flavio Oquendo, Axel Legay

Formal Architecture Description of Trustworthy Systems-of-Systems with SosADL

Flavio Oquendo and Axel Legay

July 2015

Over the last 20 years, considerable research effort has been put into conceiving Architecture Description Languages (ADLs), resulting in the definition of different languages for formal modelling of static and dynamic architectures of single systems. However, none of these ADLs has the expressive power to describe the architecture of a trustworthy System-of-Systems (SoS). SosADL is a novel ADL specifically conceived for describing the architecture of Software-intensive SoSs. It provides a formal language that copes with the challenging requirements of this emergent class of complex systems that is increasingly shaping the future of our software-reliant world.

The importance of developing sound languages and technologies for architecting SoSs is highlighted in several roadmaps targeting year 2020 and beyond, e.g. ROAD2SoS and T-Area-SoS. They show the importance of progressing from the current situation, where SoSs are basically developed in ad-hoc ways, to a rigorous approach for mastering the complexity of Software-intensive SoSs.

Complexity is inevitable in SoSs since missions in SoSs are achieved through emergent behaviour drawn from the interaction among constituent systems. Hence, complexity poses the need for separation of concerns between architecture and engineering: (i) architecture focuses on reasoning about interactions of parts and their emergent properties; (ii) engineering focuses on designing and constructing such parts and integrating them as architected.

A key facet of the design of any software-intensive system or system-of-systems is its architecture, i.e. its fundamental organization embodied in the components, their relationships to each other, and to the environment, and the principles guiding its design and evolution, as defined by the ISO/IEC/IEEE Standard 42010 [2].

Therefore, the research challenge raised by SoSs is fundamentally architectural: it is about how to organize the interactions among the constituent systems to enable the emergence of SoS-wide behaviours/properties derived from local behaviours/properties (by acting only on their interconnections, without being able to act in the constituent systems themselves).

Trustworthiness is thereby a global property directly impacted by emergent behaviours - which may be faulty, resulting in threats to safety or cyber-security.

Various recent projects have addressed this challenge by formulating and

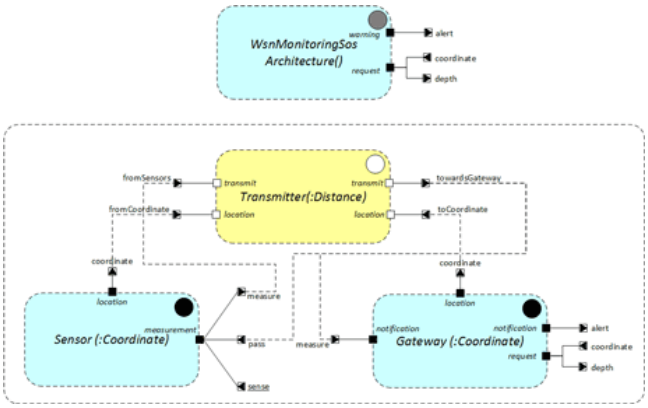

Figure 1: Abstract architecture of a flood monitoring SoS

formalizing the architecture of software-intensive SoSs. A systematic literature review revealed that 75% of all publications addressing the architecture of software-intensive SoSs appeared in the last five years, and approximately 90% in the last 10 years. Much of the published research describes open issues after having experimented with existing systems approaches for architecting or engineering SoSs.

Actually, although different Architecture Description Languages (ADLs) have been defined for formally modelling the architecture of single systems, none has the expressive power to describe the architecture of software-intensive SoSs [3, 1].

To fill this gap, we have defined SosADL, a novel ADL specifically conceived for formally describing the architecture of trustworthy software-intensive SoSs.

Formally defined in terms of the π-calculus with concurrent constraints, SosADL provides architectural concepts and notation for describing SoS architectures. The approach for the design of SosADL is to provide architectural constructs that are formally defined by a generalization of the π-calculus with mediated constraints. Both safety and cyber-security are addressed.

Using SosADL, an SoS is defined by coalitions that constitute temporary alliances for combined action among systems connected via mediators. The coalitions are dynamically formed to fulfil the SoS mission through emergent behaviours under safety and cyber-security properties. The SoS architecture is defined intentionally in abstract terms (Figure 1) and is opportunistically created in concrete terms (Figure 2).

A major impetus behind developing formal languages for SoS architecture description is that their formality renders them suitable to be manipulated by software tools. The usefulness of an ADL is thereby directly related to the kinds of tools it provides to support architecture description, but also analysis and evolution, in particular in the case of SoSs.

We have developed an SoS architecture toolset for supporting architecture-

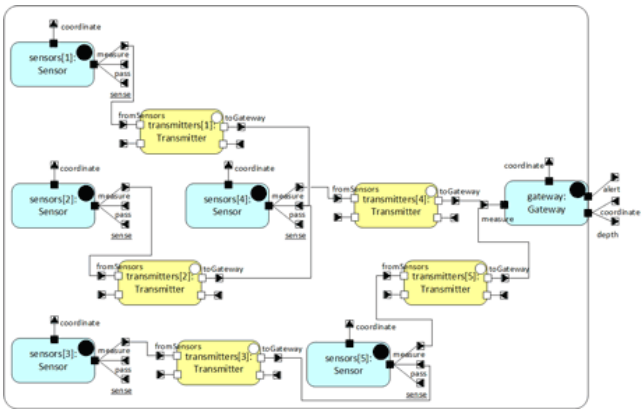

Figure 2: Concrete architecture of a flood monitoring SoS.

centric formal development of SoSs using SosADL. This toolset, SoSmart, is constructed as plugins in Eclipse Luna. It provides a Model-Driven Architecture software environment where the SosADL meta-model is transformed to different meta-models and converted to input languages of external tools, of which we have selected: UPPAAL for model checking, PLASMA-Lab for statistical model checking, DEVS and FMI (Functional Mockup Interface)/FMU (Functional Mockup Unit) for simulation.

In our approach for co-engineering safety and cyber-security supported by SoSmart, we are extending techniques applied for safety analysis to address cyber-security evaluation. This promising approach tackles different open issues, largely due to fundamental differences between the accidental nature of the faults appearing in safety analysis, and the intentional, human nature of cyber-attacks.

SosADL, supported by its SoSmart toolset, has been applied in various case studies and pilot projects for architecting SoSs, including a pilot project of a real SoS for architecting a novel flood monitoring and emergency response SoS to be deployed in the Monjolinho River. This SoS is based on different kinds of constituent systems: sensor nodes (for measuring river level depth via pressure physical sensing), a gateway and base station (for analyzing variations of river level depths and warning inhabitants of the risk of flash flood), UAVs (Unmanned Aerial Vehicles for minimizing the problem of false-positives), and VANETs (Vehicular Ad-hoc Networks embedded in vehicles of rescuers). In addition to the deployment in the field, this SoS (via the gateway system) has access to web services providing weather forecasting used as input of the computation of the risk of flash flood.

In the context of this pilot project, the SosADL met the requirements for describing trustworthy SoS architectures. As expected, a key identified benefit of using SosADL was the ability, by its formal foundation, to validate and verify the studied SoS architectures very early in the SoS lifecycle with respect

to trustworthiness, including analysis of uncertainties in the framework of safety and cyber-security.

Future work will address the application of SosADL in industrial-scale pilot projects, feeding back the research work on the ADL. This will include joint work with DCNS for applying SosADL to architect naval SoSs, and IBM in which SosADL will be used to architect smart-farms in cooperative settings.

Link:`http://www-archware.irisa.fr/`

References

[1] Milena Guessi, Valdemar Vicente Graciano Neto, Thiago Bianchi, Katia Romero Felizardo, Flávio Oquendo, and Elisa Yumi Nakagawa. A systematic literature review on the description of software architectures for systems of systems. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing, Salamanca, Spain, April 13-17, 2015*, pages 1433–1440, 2015.

[2] ISO/IEC/IEEE. Systems and software engineering architecture description. December 2011.

[3] Ivano Malavolta, Patricia Lago, Henry Muccini, Patrizio Pelliccione, and Antony Tang. What industry needs from architectural languages: A survey. *IEEE Trans. Software Eng.*, 39(6):869–891, 2013.