Architecting Systems of Systems

Module 2: *The AMADEOS SysML profile to support SoS conceptual modeling*

P. Lollini – Univ. of Florence
A. Babu – ResilTech srl

Tutorial session – July 16th, 2016
Module 2 - outline

• Intro to MDE and SysML basic modeling constructs
• The AMADEOS SysML profile: intro, overview, role in MDE
• Examples of profile’s applications, demos, exercises
INTRO TO MDE AND SYSML
BASIC MODELING CONSTRUCTS
THE AMADEOS SYSML PROFILE: INTRO, OVERVIEW, ROLE IN MDE
Basic concepts and relationships

• Definition of **basic** SoS concepts
  – Not domain specific
  – Applicable to any SoS

• Definition of **relationships** among basic SoS concepts
From Conceptual to SysML modeling

SoS Basic concepts

High-level representation

Semi-formalization
High-level representation

• Explicates relations among basic SoS concepts per viewpoint
• Supports their graphical visualization
• Semantic
  – **Boxes** represent concepts
  – **Arrows** represents semantic relationships among concepts (labels in natural language)
Relating basic SoS concepts in Structure view

- **“System-of-Systems (SoS):** An SoS is an integration of a finite number of constituent systems (CS) which are independent and operable, and which are networked together for a period of time to achieve a certain higher goal.”

- **“Constituent System (CS):** An autonomous subsystem of an SoS, consisting of computer systems and possibly of a controlled objects and/or human role players that interact to provide a given service.”
Dynamicity / Evolution

- "Dynamicity: The property of an entity that is constantly changing in terms of offered services, built-in structure and interactions with other entities."

- "Managed SoS evolution: Process of modifying the SoS to keep it relevant in face of an ever-changing environment."

- "Business value: Overarching concept to denote the performance, impact, usefulness, etc. of the functioning of the SoS"
Dependability / Security / Multi-criticality

- “Critical service: A critical service is the service of a system that requires a specific criticality level.”
- “Criticality level: The criticality level is the level of assurance against failure.”
Time / Emergence

- **Clock**: A (digital) clock is an autonomous system that consists of an oscillator and a register. Whenever the oscillator completes a period, an event is generated that increments the register.

- **Reference clock**: A hypothetical clock of a granularity smaller than any duration of interest and whose state is in agreement with TAI.

- **Emergence**: A phenomenon of a whole at the macro-level is emergent if and only if it is new with respect to the non-relational phenomena of any of its proper parts at the micro level.

- **Explained emergence**: ...if a trans-ordinal law that explains the occurrence of the emergent phenomenon at the macro level out of the properties and interactions of the parts at the adjacent micro level is known.

- **Unexplained Emergence**: ...if, after a careful analysis of the emergent phenomenon, no trans-ordinal law that explains.... is known (at least at present).
Towards a semi-formal representation

– Describe AMADEOS viewpoints via a SysML profile

• UML-like representation
  – Improve the understanding by using different levels of abstraction and different views
  – Foster information sharing and reuse among SoS stakeholders
  – Enable analysis and experimentation at early stage of the SoS lifecycle process
AMADEOS SoS Profile: Rationale

• A SysML profile to implement SoS basic concepts and their relationships
  – Strong focus on conceptual modelling
  – Capturing both static structure and dynamic behavior

• Organized in viewpoints driven-components

• Profile implementation
  – Open source Eclipse integrated development environment along with Papyrus plug-in
AMADEOS profile components

Structure viewpoint

SoSDependability
SoSMulti-criticality
SoSSecurity
SoSTime

SoSInterface
SoSArchitecture
SoSCommunication

SoSDynamicity
SoSEvolution
SoSEmergence
Mapping structure view into AMADEOS SoS profile

• Starting from the high-level graphical representation of Structure…

• ….we implemented three SysML profile components:
  – *SoSArchitecture* component
  – *SoSCommunication* component
  – *SoSInterface* component

Tutorial – P. Lollini, A. Babu
SoSArchitecture package
Dynamicity / Evolution
Time / Emergence
Role in MDE

• SysML SoS profile as a Platform Independent Model
  – It provides a set of concepts involving SoS without losing the platform independent characteristics.

• Platform-Specific Models
  – can be defined and used at different stages of system design and validation
The profile can be integrated into existing SysML MDE tool-chain platforms, to perform e.g.

- System analysis
  - E.g. Hazard Analysis (HA), Failure Mode and Effect Analysis (FMEA), Fault Tree Analysis (FTA)
  - Interface compatibility checking
  - Causal loop detection for detrimental emergence prevention
- Source code generation
- System simulation
- System testing
Causal loops detection

- **Causal loops**
  - are *formed by the information transfer via cyber-channels and stigmergic channels*
  - can be the cause for the appearance of *emergent phenomena*.

- **SysML profile** captures the dependences between CSs at the interface level
  - to *automatically detect causal loops*
  - Thus discovering the potential for *emergent phenomena at design time*.
Traffic Flow

Traffic flows if section of road in front of a car is empty
Traffic Jam

In case of a disturbance, a car does not free the current section of road as fast as expected => Following cars cannot free their sections either
A causal loop exists between the environment (road) and each vehicle:

- A car causes a road section to be blocked
- A blocked road section causes the next car to block the subsequent section
Abstraction of SoS

The abstract description of this SoS highlights the **cyclic dependence** between the vehicle and its environment.
Conclusion

• Viewpoint-driven approach to design SoSs by adopting a SysML profile
• Implementing the profile in the Eclipse integrated development environment jointly with Papyrus
• Applicability of the profile for SoS description and analysis
References

AMADEOS – Public deliverables (http://amadeos-project.eu/documents/public-deliverables/)
• D2.1 - Basic SoS concepts, glossary and preliminary conceptual model
• D2.2 - AMADEOS Conceptual Model.
• D2.3 AMADEOS conceptual model – Revised (to appear – end of 2016)

A. Babu

EXAMPLES OF PROFILE’S APPLICATIONS, DEMOS, EXERCISES