Kairos team

Model based Systems Engineering, from co-modeling to co-simulation

Julien Deantoni
Universite Cote d'Azur,
CNRS I3S, INRIA KAIROS
Julien.deantoni@polytech.unice.fr
The term "system" comes from the Latin word systēma, in turn from Greek σύστημα systēma: "whole concept made of several parts or members, system", literary "composition".

Many stakeholders are needed to develop such systems.
Many stakeholders are needed to develop such systems.
System Engineering

Several concerns for a single system

Avionics
Aerodynamics
Propulsion System
Communications
Navigation
Mechanical Structure
Airlines
Human-Machine Interaction
Environmental Impact
Safety Regulations
Control
Several concerns for a single system

At least one expert by concern
Several concerns for a single system

At least one expert by concern

At least one “model” by concern (expressed in a DSML)
System Engineering

Several concerns for a single system

At least one expert by concern

At least one “model” by concern (expressed in a DSML)

A model is an abstraction of a specific Concern/View of a system for a given purpose

A system is represented by a set of functional and non-functional heterogeneous models
Several concerns for a single system
At least one expert by concern
At least one “model” by concern (expressed in a DSML)
Model Based System Engineering specifies in a model the correspondences between models from the different concerns, all along the product life cycle.
Model Based System Engineering specifies in a model the correspondences between models from the different concerns, all along the product life cycle.
Model Based System Engineering specifies in a model the correspondences between models from the different concerns, all along the product life cycle.
Model Based System Engineering specifies in a model the correspondences between models from the different concerns, all along the product life cycle.

Lots of interesting challenges here...
Model Based System Engineering specifies in a model the correspondences between models from the different concerns, all along the product life cycle.
MBSE challenges

A Kairos point of view

1) Make model executable: models are mostly syntactic and behavioral semantics is « elsewhere » (sometimes not accessible) → forbids the understanding of the behavior
   • We consider models that can be interpreted according to their (concurrent and timed) operational semantics
   • We do not want to implement all the tooling for each new language
MBSE challenges

A Kairos point of view

1) Make model executable: models are mostly syntactic and behavioral semantics is « elsewhere » (sometimes not accessible) → forbids the understanding of the behavior
   • We consider models that can be interpreted according to their (concurrent and timed) operational semantics
   • We do not want to implement all the tooling for each new language
MBSE challenges

*A Kairos point of view*

1) **Make model executable:** models are mostly syntactic and behavioral semantics is « elsewhere » (sometimes not accessible) → forbids the understanding of the behavior
   • We consider models that can be interpreted according to their (concurrent and timed) operational semantics
   • We do not want to implement all the tooling for each new language

```
What is a Syntax
  Syntax
  A Language
  A model
  \(\text{conformsTo}\)

What is an operational semantics
  Operational semantics
  (partially) implements

An interpreter

Tools the meta language to avoid tooling each language
```

**Syntax**

**Operational semantics**

**A Language**

**A model**

**Tooling the meta language to avoid tooling each language**

**Executable DSML**

**Executable model**

**Runtime**
MBSE challenges

A Kairos point of view

1) Make model executable: models are mostly syntactic and behavioral semantics is « elsewhere » (sometimes not accessible) → forbids the understanding of the behavior
   • We consider models that can be interpreted according to their (concurrent and timed) operational semantics
   • We do not want to implement all the tooling for each new language

Meta-languages

Executable DSML

Executable model

Runtime

What is a Syntax

What is an operational semantics

Tooling the meta language to avoid tooling each language

SubChallenges:
• How to formally specify an operational semantics? (Expressiveness, analyzability, etc)
• How to specify and tool the language that describes the operational semantics?
• How to automatically generate dedicated interpreters?
The GEMOC Initiative  On the Globalization of Modeling Languages

GEMOC is an open and international initiative that aims to coordinate and disseminate the research results regarding the support of the coordinated use of various modeling languages that will lead to the concept of the globalization of modeling languages.

http://gemoc.org
Modeling workbench: [http://gemoc.org/studio/](http://gemoc.org/studio/)

Making the Capella language executable....
A generic execution engine
Only configured by the semantics!

Making the Capella language executable....
Modeling workbench: [http://gemoc.org/studio/](http://gemoc.org/studio/)

A generic scheduling state space explorer

Only configured by the semantics!
The VerCors platform

A specification and verification platform for distributed applications [FASE’16]
- Eclipse-based graphical editors for distributed component architecture (Fractal/GCM), interfaces, state-machines
- Bridges to the CADP verification and model-checking toolset.

[FASE’16] Integrated environment for verifying and running distributed components. Ludovic Henrio, Oleksandra Kulankhina, Siqi Li, Eric Madelaine. FASE’16
MBSE challenges

A Kairos point of view

2) **Ensure models consistency**: Correspondences between models are left implicit or informal → forbids understanding of emerging behaviors. We must be able to execute conjointly heterogeneous modeling languages.
MBSE challenges

A Kairos point of view

2) **Ensure models consistency**: Correspondences between models are left implicit or informal → forbids understanding of emerging behaviors. We must be able to execute conjointly heterogeneous modeling languages.

For both new and existing languages, white or black box, all this trying to stick the industrial standards.
The BCOoL approach has been defined for cyber model coordination.

Recently a thesis has been started with Safran to drive co-simulation of Cyber-Physical systems.
Conclusion
Conclusion