Model Management for Model-Based Systems Engineering of Complex Systems: A Federation of Presentations

Dominique Blouin, Rakshit Mittal, Yara Hallak, Anish Bhobe, Mohammad-Sajad Kasaei

LTCI Lab

Telecom Paris, Institut Polytechnique de Paris

dominique.blouin@telecom-paris.fr

What is a complex system?

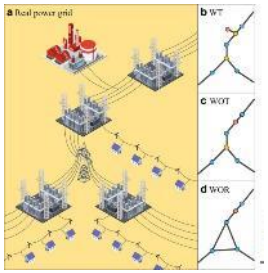
- Composed of **many components** which **interact with each other**.
  - Transportation or communication systems.
  - Complex software and electronic systems.
  - Infrastructures such as power grid.
  - Living organisms
  - And ultimately the entire universe.

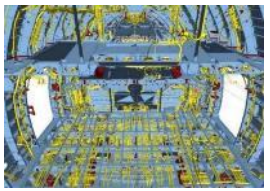
- Behavior is intrinsically **difficult to model** (plan) due to the different types of **interactions** between their parts or their environments.

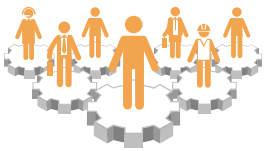
- Properties such as **nonlinearity**, **emergence**, **self organization**, **adaptation**, and **feedback loops**, among others.

- Often consists of Systems of Systems (SoS)

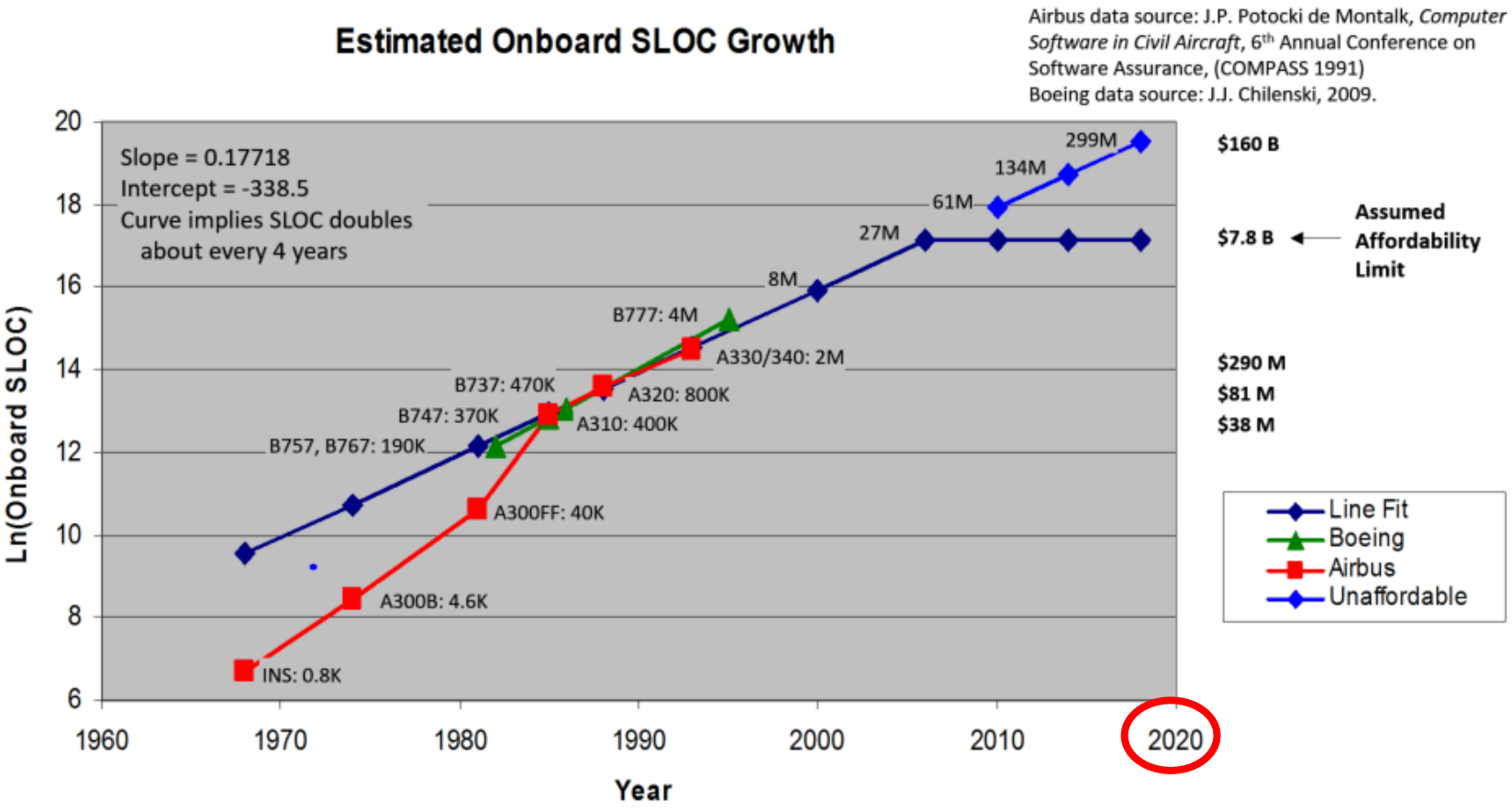







Example Complexity in Engineering Systems



Source: Feiler, Hansson, de Niz and Wrage. “System Architecture Virtual Integration: An Industrial Case Study”, 2009.

Non-Linear Development Efforts

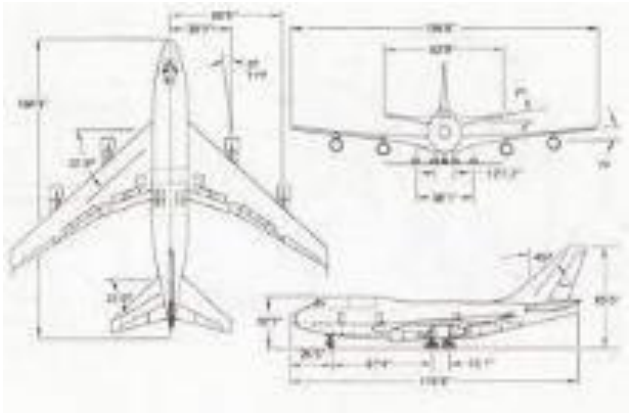
- F35 SLOC / F16 SLOC ~ 175
- F35 Effort / F16 Effort ~ **300**
  - Source: SAVI Project (https://savi.avsi.aero/)

Non-Linear!

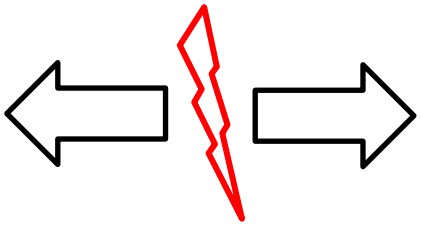
- A400M:
  - Over 10 years delayed (2013)
  - 6.2 billion euros over budget (30% overrun)
  - Source: https://simpleflying.com/airbus-a380-program-software-discrepancies-delay-story/

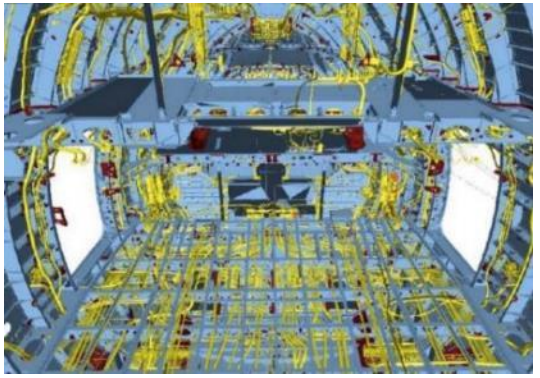
Mismatched Assumptions in Collaborative Engineering: A380

- A380 cables and airframe **inconsistency** problem:
  - Cables **too short** for the airframe.
  - Only discovered at **system integration** time.
  - Wiring had to be **completely redone** from the design stage.
  - About **5 billion Euros** in additional costs!



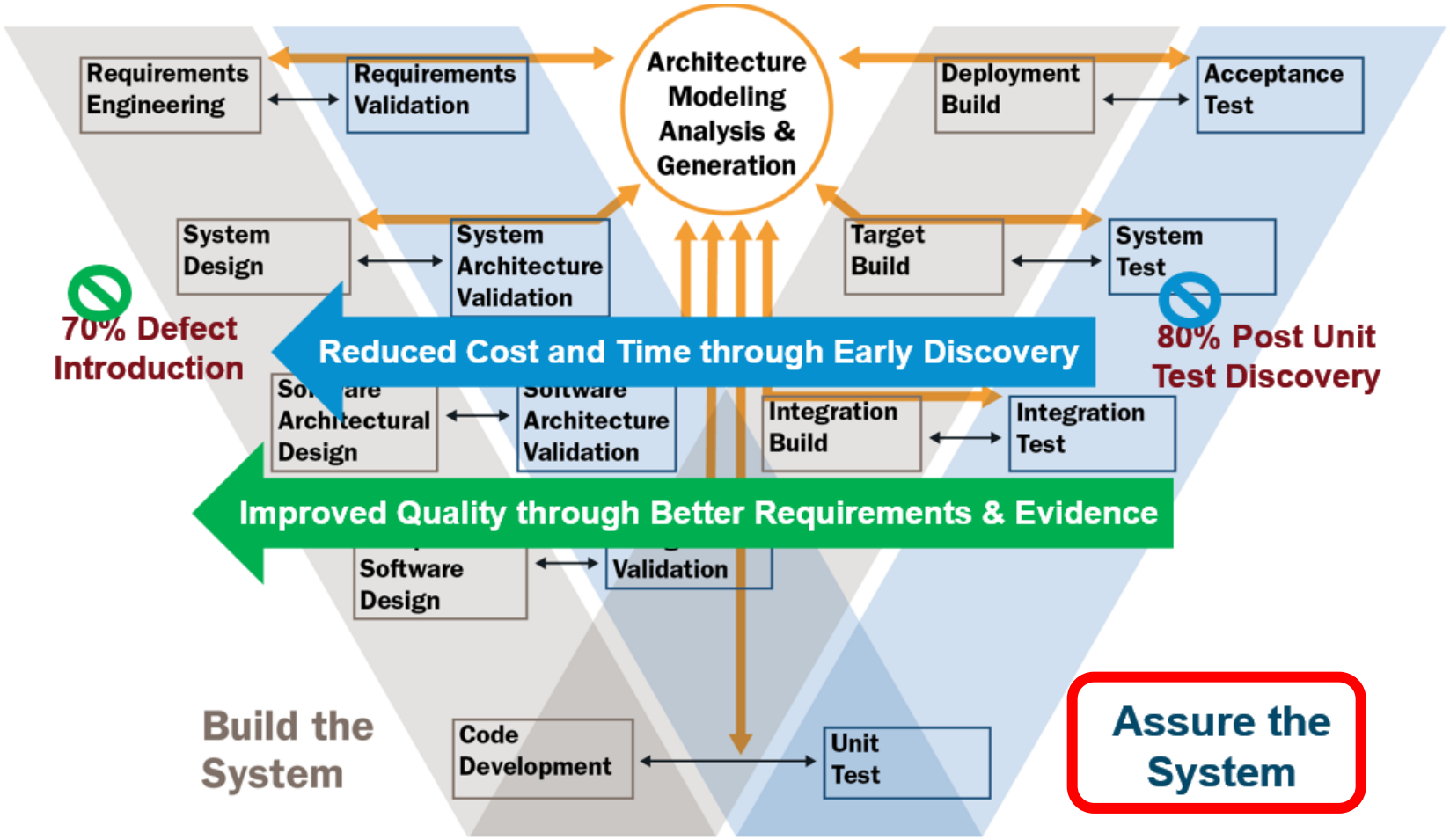
Toulouse, France





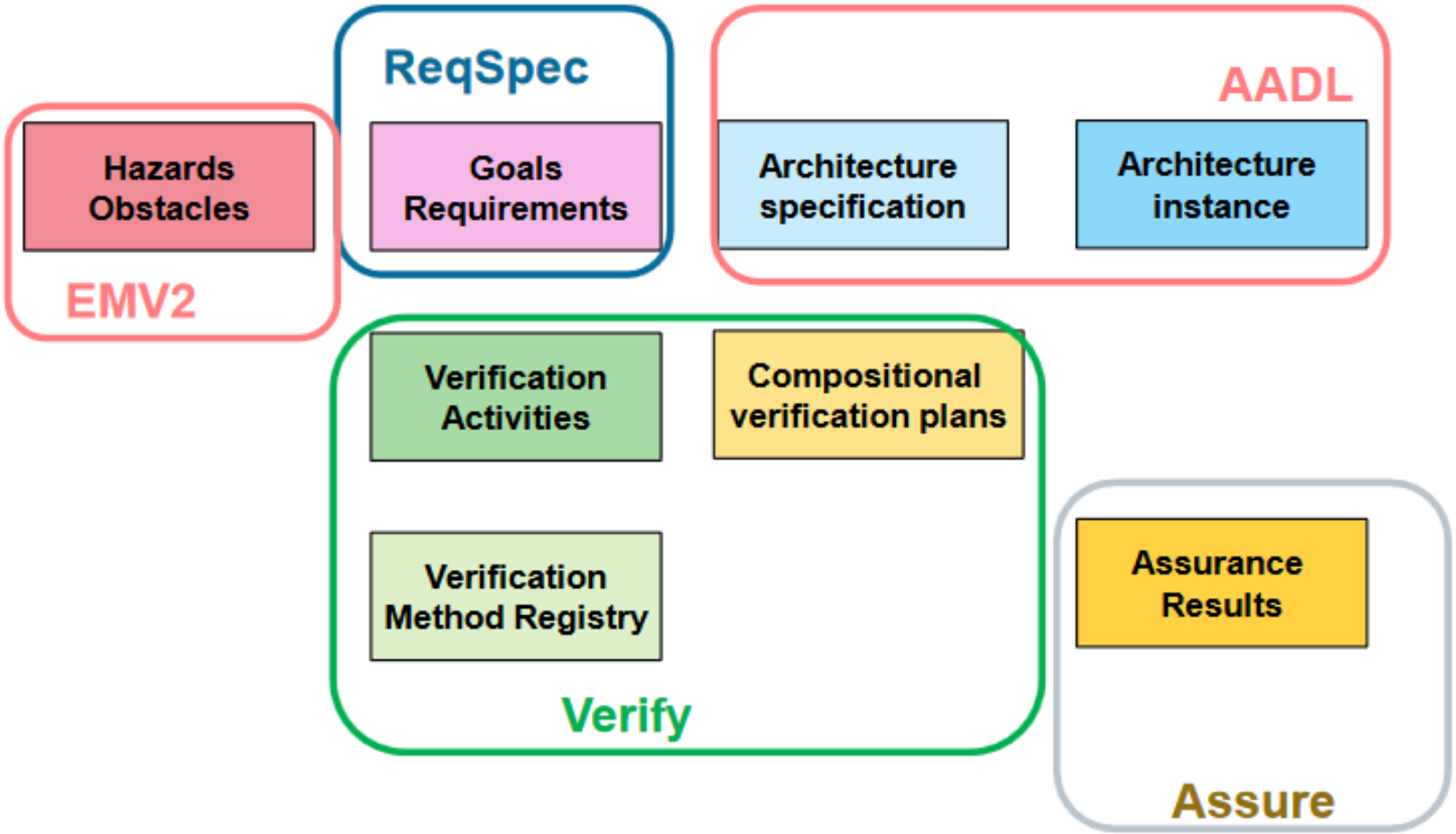
Hamburg, Germany

MBSE as Solution: ACVIP (Architecture-Centric Virtual Integration Process)



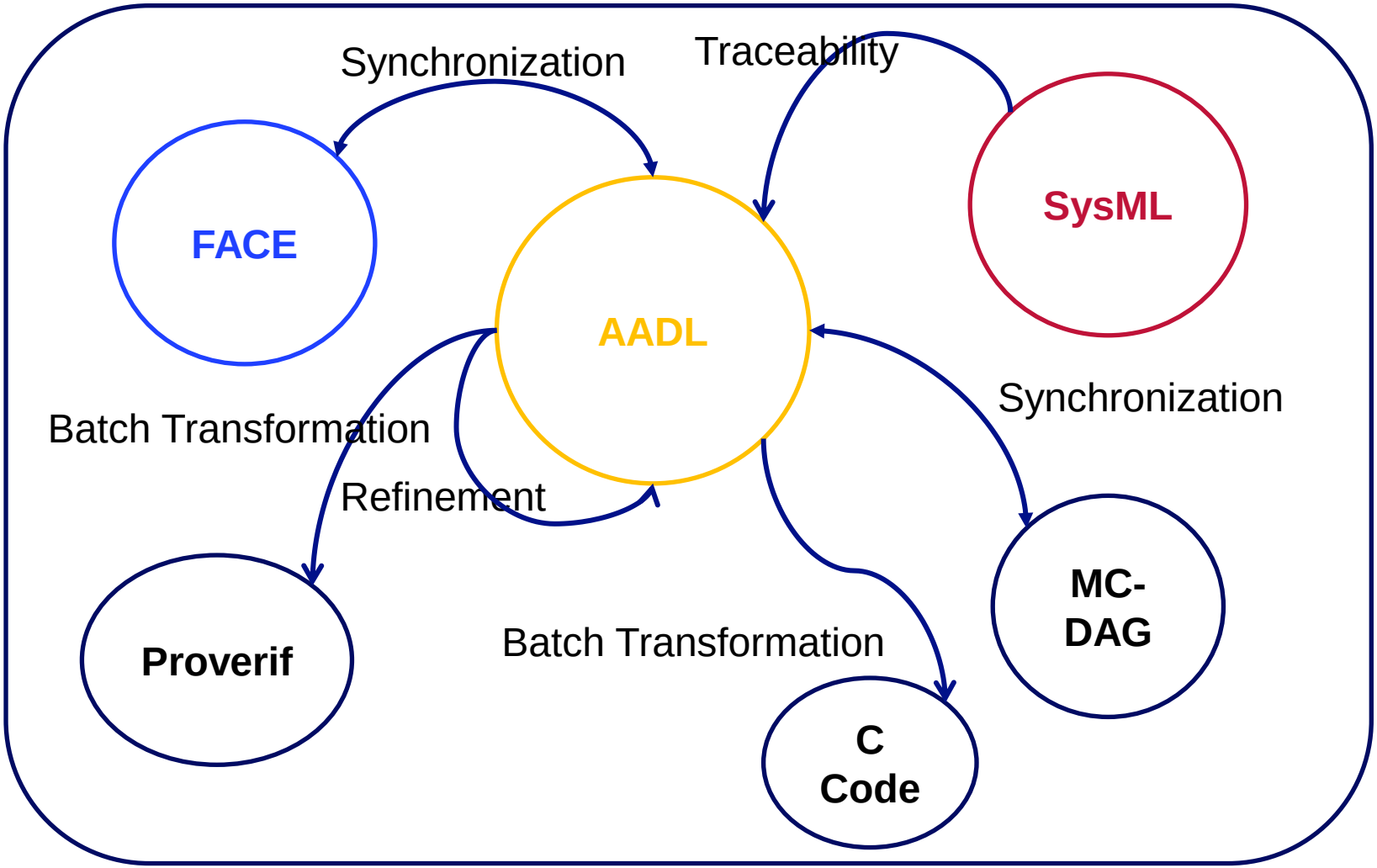
Source: J. McGregor, P. Gluch and P. Feiler, “Analysis and Design of Safety-critical, Cyber-physical Systems”, 2017.

ALISA (Architecture-Led Incremental System Assurance) Unified Concepts



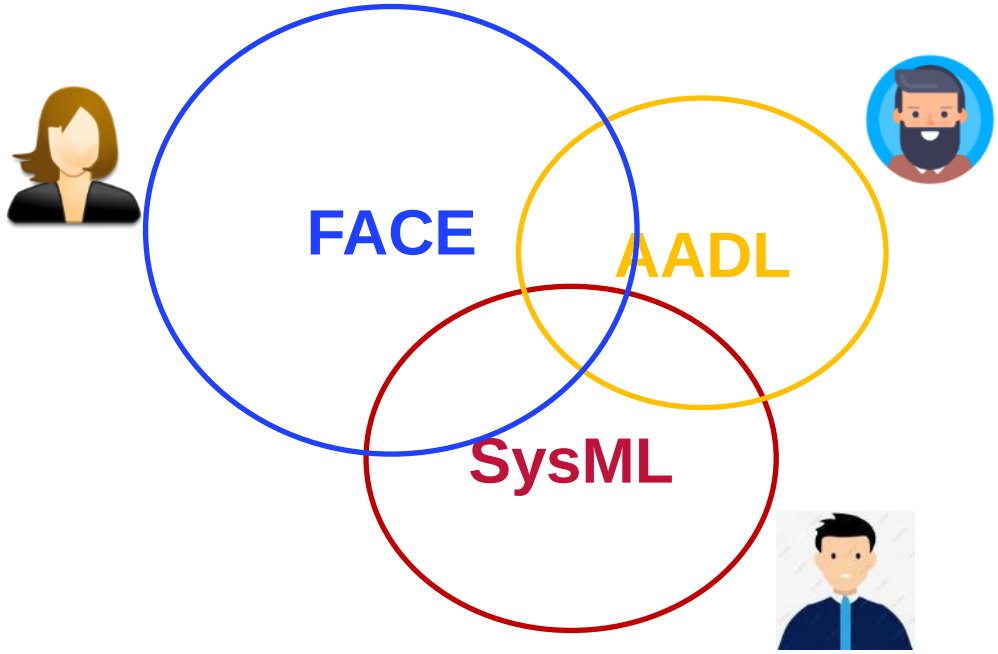
Source: Peter Feiler, *ALISA Tutorial*, 2018.

Many kinds of Relations between Models
MoM is about that…

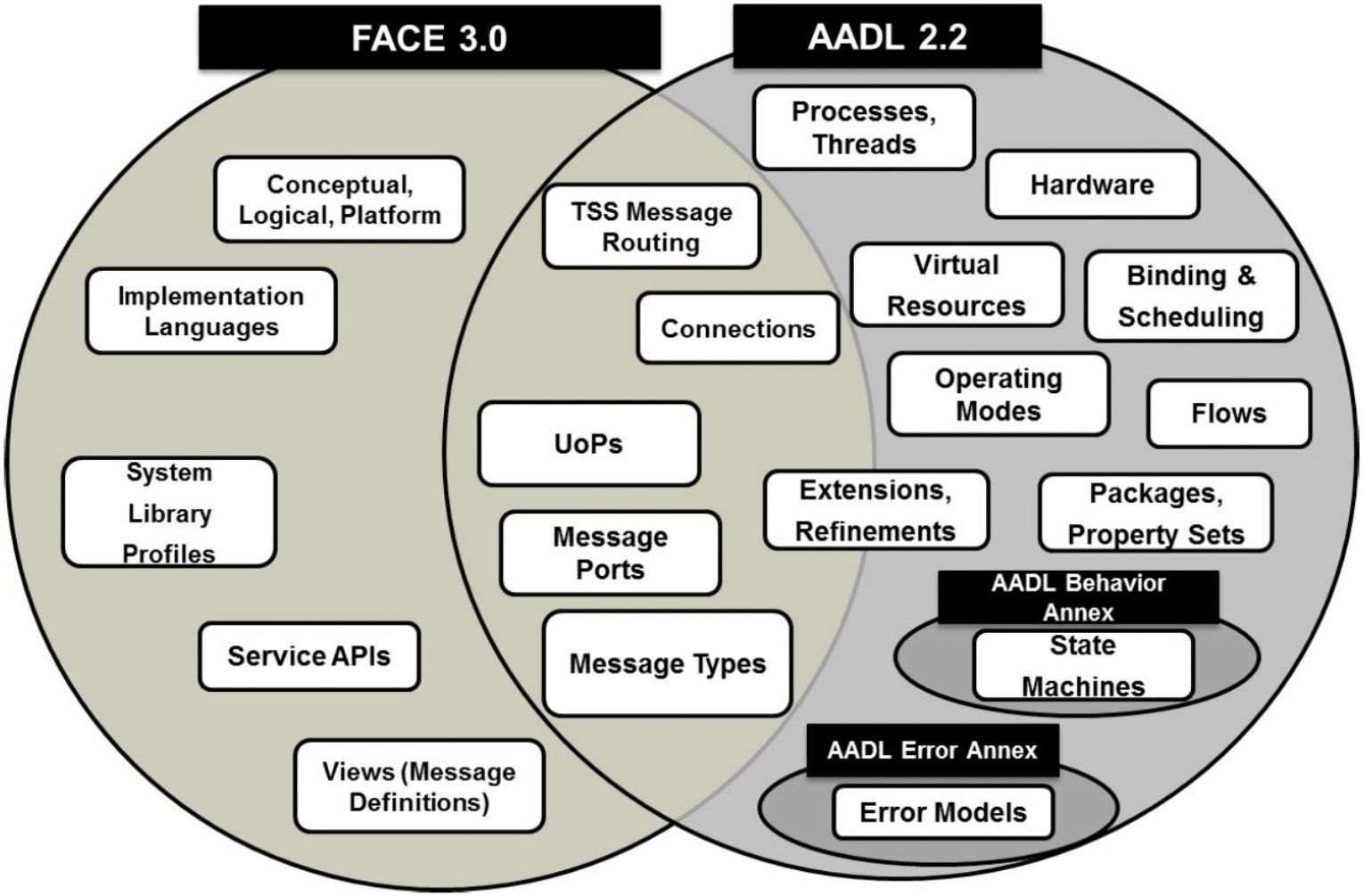


Model Changes Problems

- Models must be synchronized.
- Who can modify what and when?
- What is the Single Source of Truth?
  - Actually, not single; Authoritative Source of Truth (ASoT)



FACE and AADL Information Overlap



Anish Bhobe's PhD: Change Policies for Model Change Management

- Some information is inherently of higher importance:
  - Data-sheets for components should not be changeable.
  - Same for physical constants
  - Requirements drive design (and conversely)
  - Etc.
- Some information can have higher importance for certain users:
  - Constraints given by systems engineers to domains engineers.
  - Ownership of certain components by teams.
- Some synchronization tools may lead to information losses under some circumstances
  - E.g.: TGGs

Senate: Change Policies Specification Language

- Framework (language and interpreter) to describe policies and their triggering based on:
  - Role of the person executing.
  - Action (Create, Update, Delete)
  - Elements or graph patterns on which the action occurs, and
  - Instance level Annotations (markers)

- Policy can:
  - Revert models to previous revision.
  - Call an arbitrary operation, or
  - Permit a change to occur.

Policy Chains

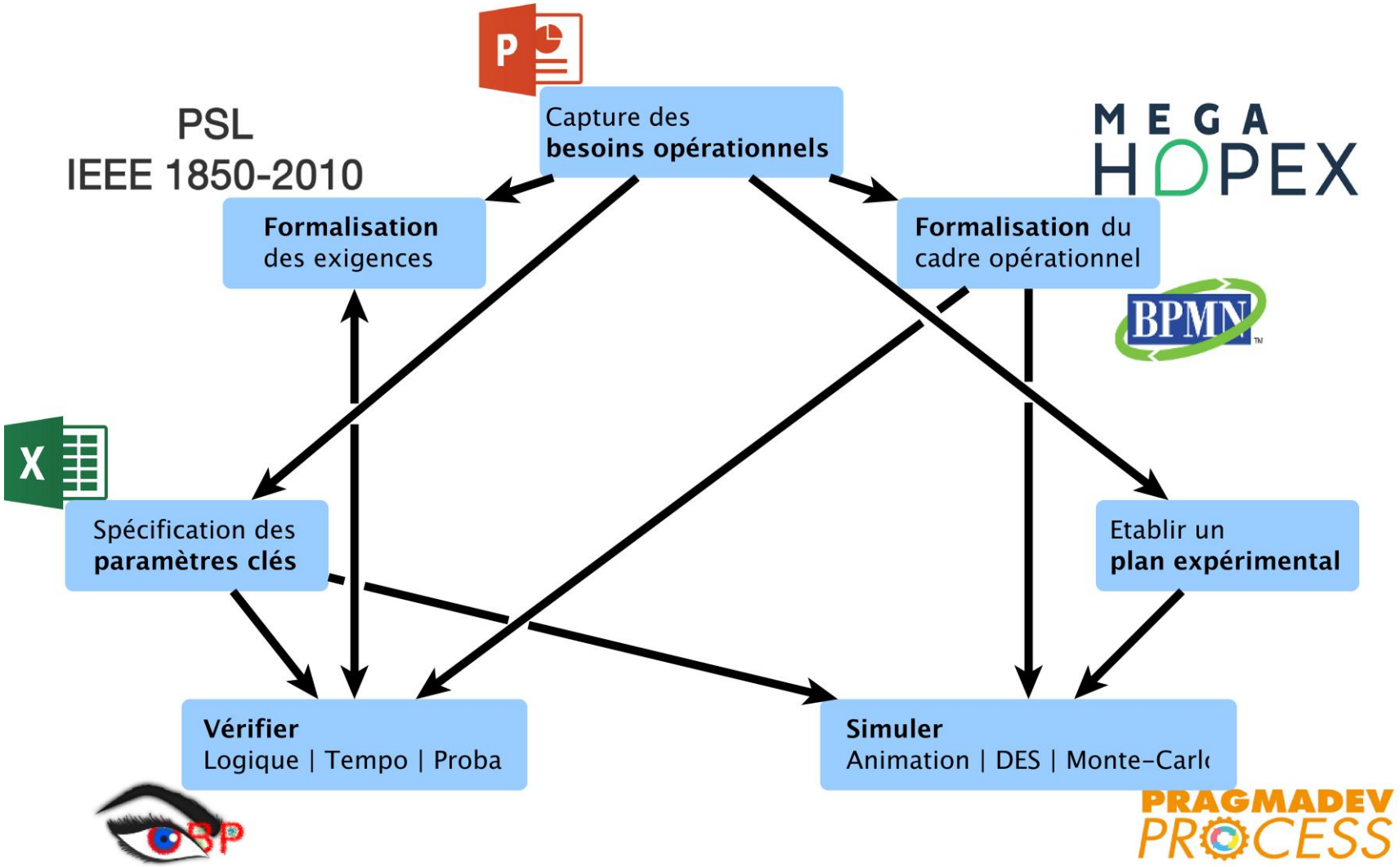
- Changes in source model will synchronize with the linked model(s).
  - A change reverted by one of the target models' policy is notified to the source model.
  - The policy developer can choose to revert change on source, notify other policies in the chain, or to warn the **MoM** framework/user of an inconsistency.

Example Policy

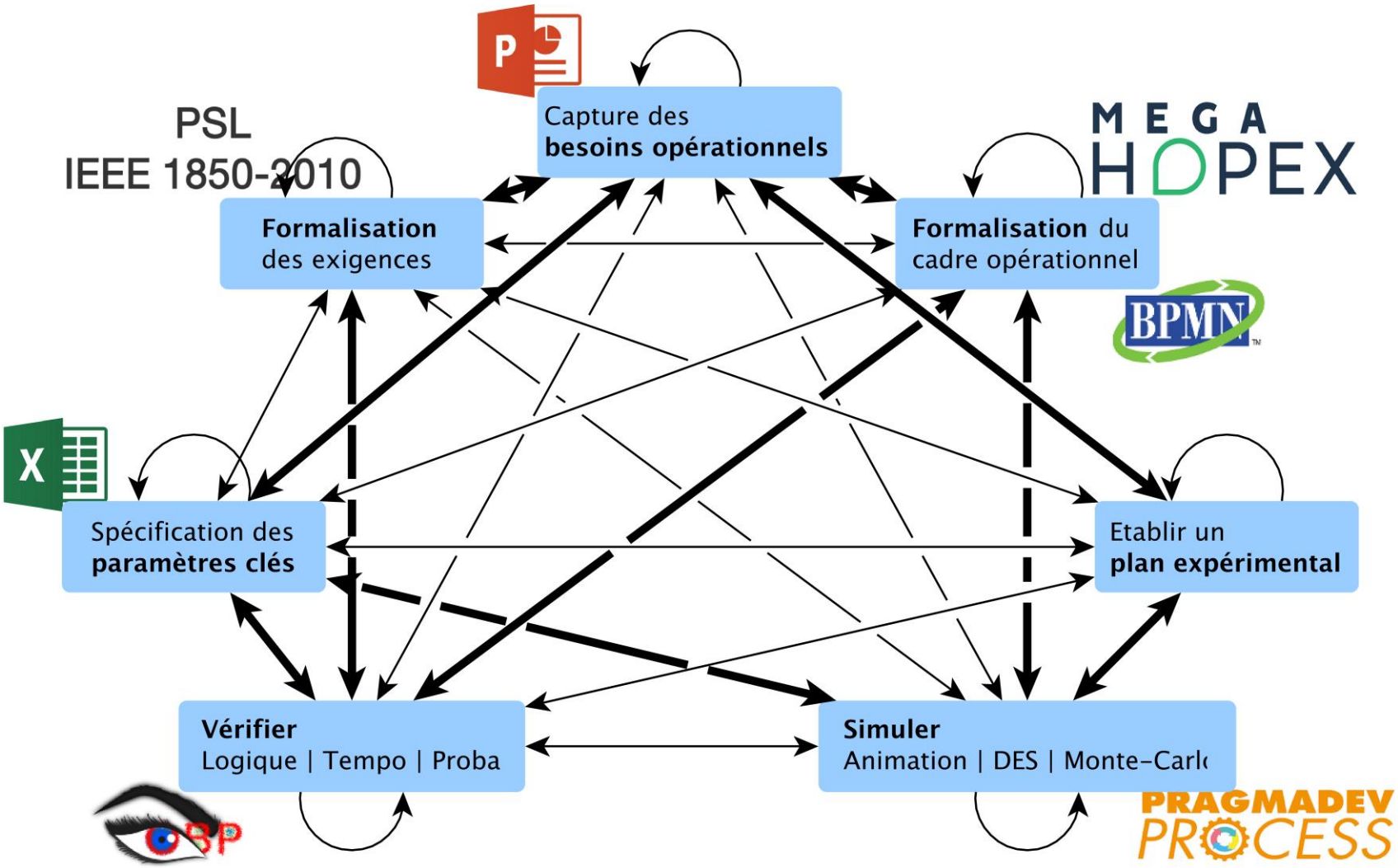
```
role AADL_Dev, FACE_Dev

policy DenyDeletion {
  applies to all
  let {
    Changes = $patterns.where[.name = AaxlProcesses]
    Removeable = $markers.where[.action = Delete]
    Unremoveable = exclude(Changes, Removable)
  } in
  postcondition {
    on Delete
    by FACE_Dev except AADL_Dev
    over any of Unremoveable
  }
  then revert
}
```

Ciprian's Thought Experiment



Actually, it is more like this!



Need to Manage Model Evolutions

- How to manage the evolution of the many, many, many models when they are:
  - Modified by many engineering teams,
  - in parallel (concurrent engineering)
  - and distributed.

- Sajed’s work on model versioning and merging.

Simulation Challenges: Example at Renault

- **Hundreds** of simulation platforms composed of **hundreds** of simulation models.
- Building **correct** simulation platforms is **difficult**!

Management of Model Validity Challenge

- How to manage experiments that are performed to:
  - Build models.
  - Validate model.

- Rakshit Mittal's PhD on a Model-Driven Reasoning Framework for the Design of Simulation Experiments.

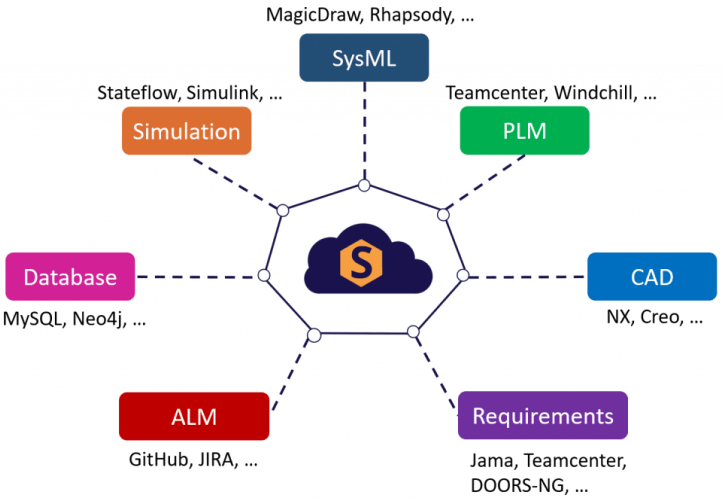
How to help Building Complex Simulation Platforms?

- Yara Hallak's PhD on an approach to automatically select models for building simulation platforms based on requirements and representativity.

MoM: An Emerging Research Field ?

- For sure it is **strongly needed by industry**.
- Many **ad-hoc** approaches are emerging…

openCAESAR



OpenMBEE

Model Management System

Ansys ModelCenter

Connecting system requirements to engineering analysis

- Towards a **theoretical foundation of MOM**

Our Approach: ACMoM
Architecture-Centric Model Management

- Foundations :

Hasso-Plattner-Institut für Softwaresystemtechnik GmbH
Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam
Fachgebiet für Systemanalyse und Modellierung

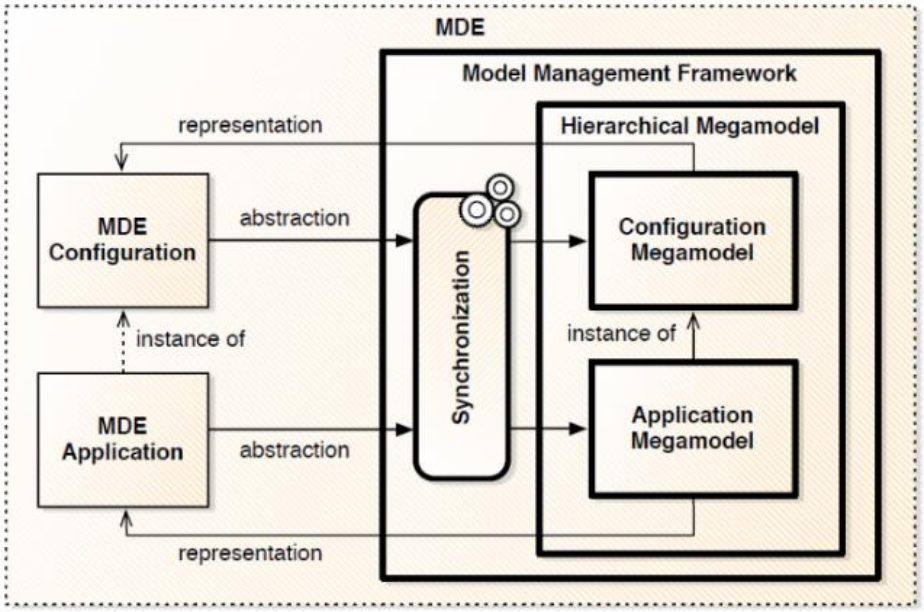
Traceability and Model Management with Executable and Dynamic Hierarchical Megamodels

PhD Thesis of Andreas Siebel (2012)

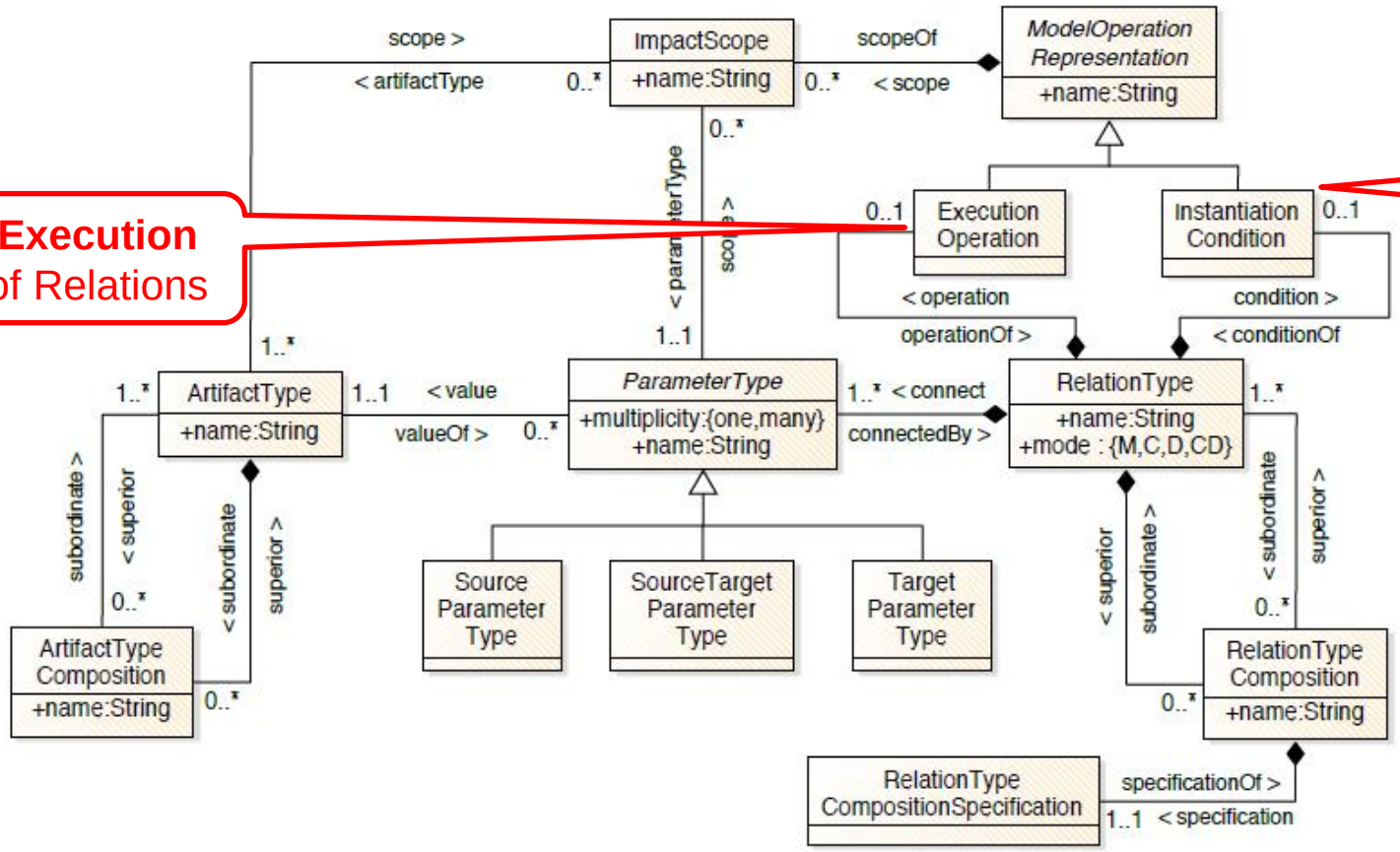
- Megamodeling (modeling in the **large scale**):
  - "A megamodel is a model with other models as elements". "A megamodel contains relationships between models." (Bézivin, 2003 / 2007)
  - "... the idea behind a megamodel is to define the set of entities and relations that are necessary to model some aspect about MDE". (Favre 2004 / 2005)
- **Modeling modeling** by creating a mega**modeling modeling** language.
- **Interpret** these megamodels to ensure consistency is preserved. In case a model change occurs.

Hierarchical Megamodels

- Megamodel: A model of the employed **models** and their **relations**.
  - E.g., represent languages and their transformations to translate models onto each others
- Interpret the megamodel during development to **manage** the models.



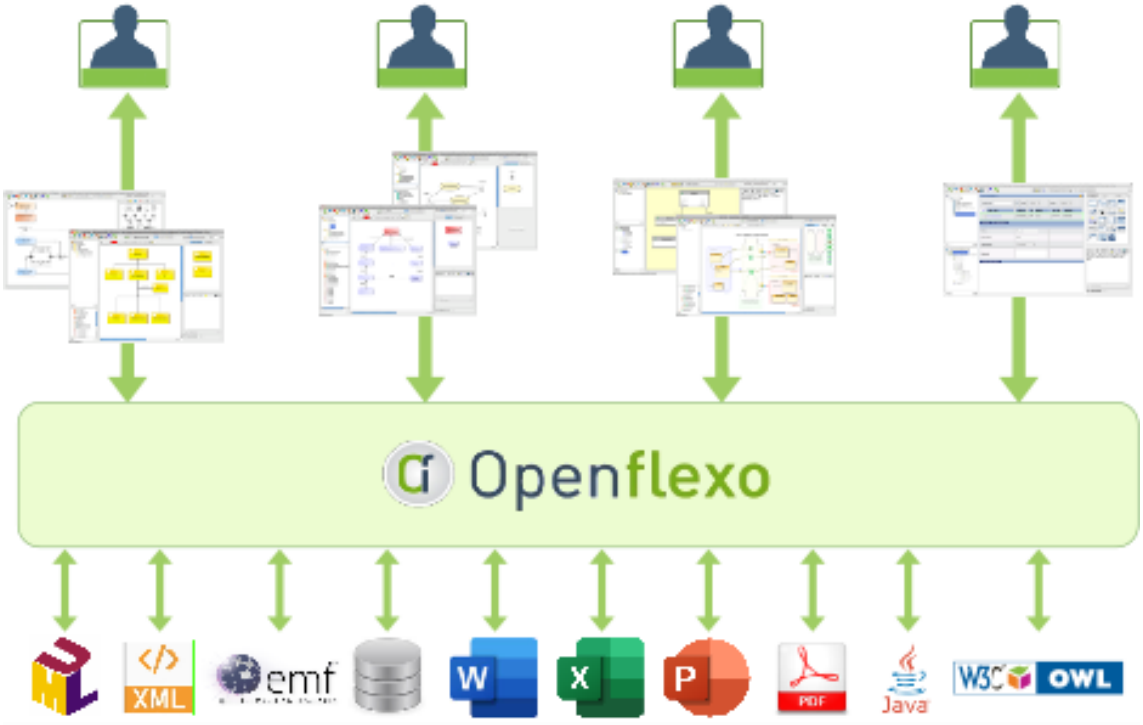
Execution of Hierarchical Megamodels



Execution of Relations

Instantiation of Relations

Sylvain's Approach: Openflexo for Model Federation



- Define a **virtual metamodel** (Flexoconcept).
- **Read** / **write** the data from the **underlying models** via a tool adapter.
  - **Synchronize** models.
- Provide **semantics** to **unformalized / unformatted data**.

AUGMENT Research Project

- Dominique Blouin (Télécom Paris), Ciprian Theodorov (ENSTA **Paris**), Sylvain Guérin (IMT Atlantique),

- Approche Unifiée de Gestion de Modèles pour l'IngéNierie des SysTèmes.
  - Unified Model Management Approach for Systems Engineering.

- Objective 1: Based on a deep state of the art on MoM, establish a theoretical foundation and prototype for the field of model management.
  - Consistency, heterogeneity of technical spaces, views, workflows.

State of the Art on MoM

- A Survey of Federative Approaches for Model Management in MBSE.
  - Published at MoM 2024.

- Ongoing: Extending the survey to cover other **approaches** and **functions** of MoM:
  - Unification, integration, besides federation.
  - Workflows, changes, evolution, validity, etc.

- Also compare the existing MoM tools.
  - Propose a half day challenge along with MoM workshop at MODELS 2025.

Other Objectives of AUGMENT

- Extend the **core foundation and prototype** for:
- Change and evolution management (1 PhD)
  - Based on existing work (e.g. presented PhD work)
- Validity management (1 PhD)
  - Based on existing work (e.g. presented PhD work)
- Individual PhD contributions are standalone but would need to be integrated in a MoM framework
- One Postdoc and one engineer.