Five Theses for Model-Based Systems Engineering and Model-Based Safety Assessment

Prof. Antoine B. Rauzy

Department of Production and Quality Engineering
Norwegian University of Science and Technology
Trondheim, Norway

&

Chair Blériot-Fabre
CentraleSupélec
Paris, France

Agenda

Model-Based Systems Engineering

Taxonomy of Models

Modeling Languages

Model-Based Safety Assessment

Model Synchronization

Agenda

Model-Based Systems Engineering

Taxonomy of Models

Modeling Languages

Model-Based Safety Assessment

Model Synchronization

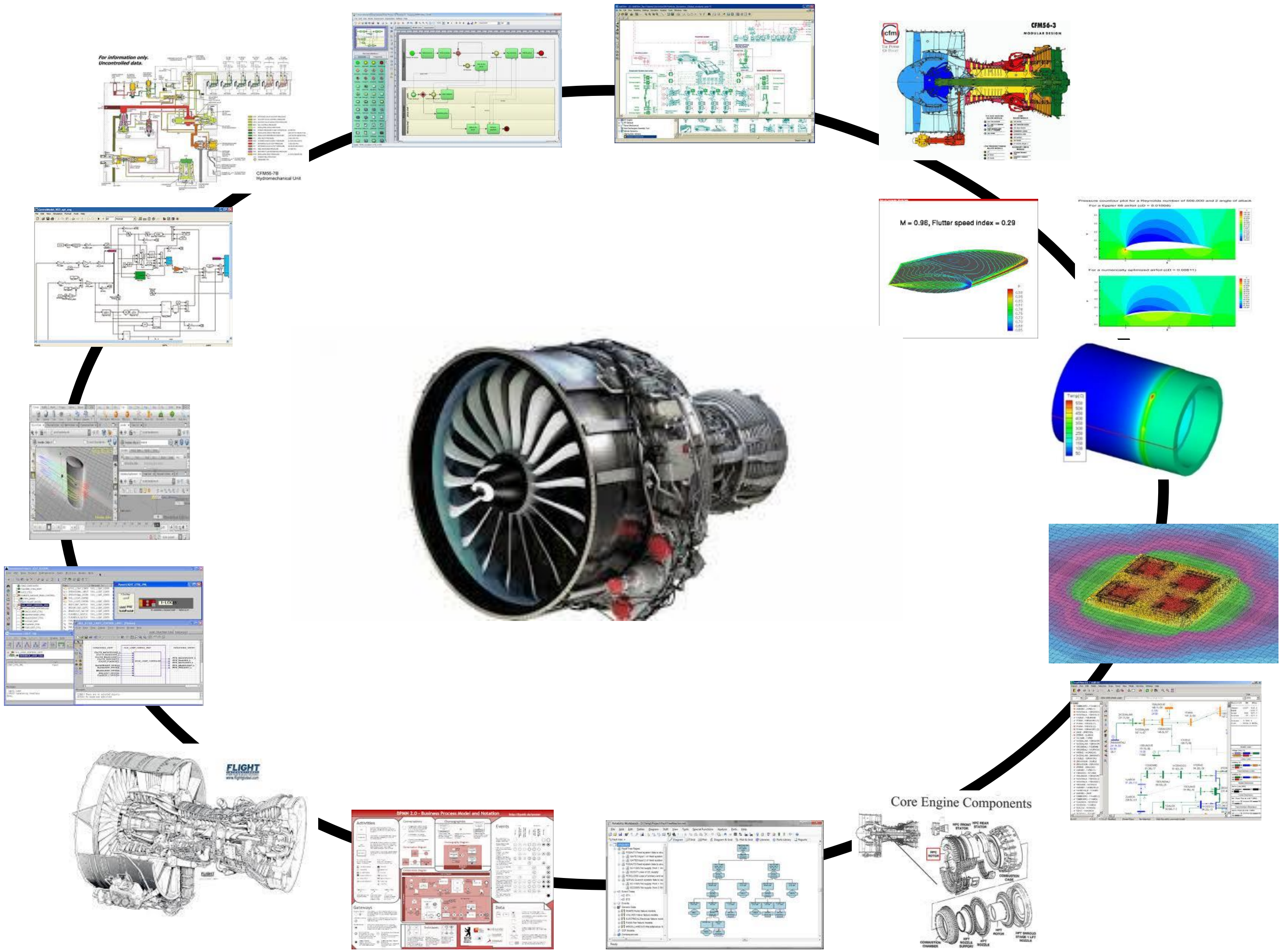
Models Are Everywhere

- The **systems** designed by industry are more and more **complex** and **interconnected**. Not only these **products** are more and more complex but also the **processes** by which they are **designed/produced/operated/decommissioned** and **organizations** that implement these processes are.

- To face this complexity, the different engineering disciplines (mechanics, thermic, electric and electronic, software, architecture…) virtualize their contents to a large extent, i.e. they are designing **models**. We entered the era of:

Model-Based Systems Engineering

- Each system comes with dozens of models. More and more of these models are **embedded** into systems and used for their operation.


For information only.
Uncontrolled data.
CFM56-7B
Hydromechanical Unit
CFM56-3
M = 0.98, Flutter speed index = 0.29
FLIGHT
Core Engine Components

The Science and Engineering of Models

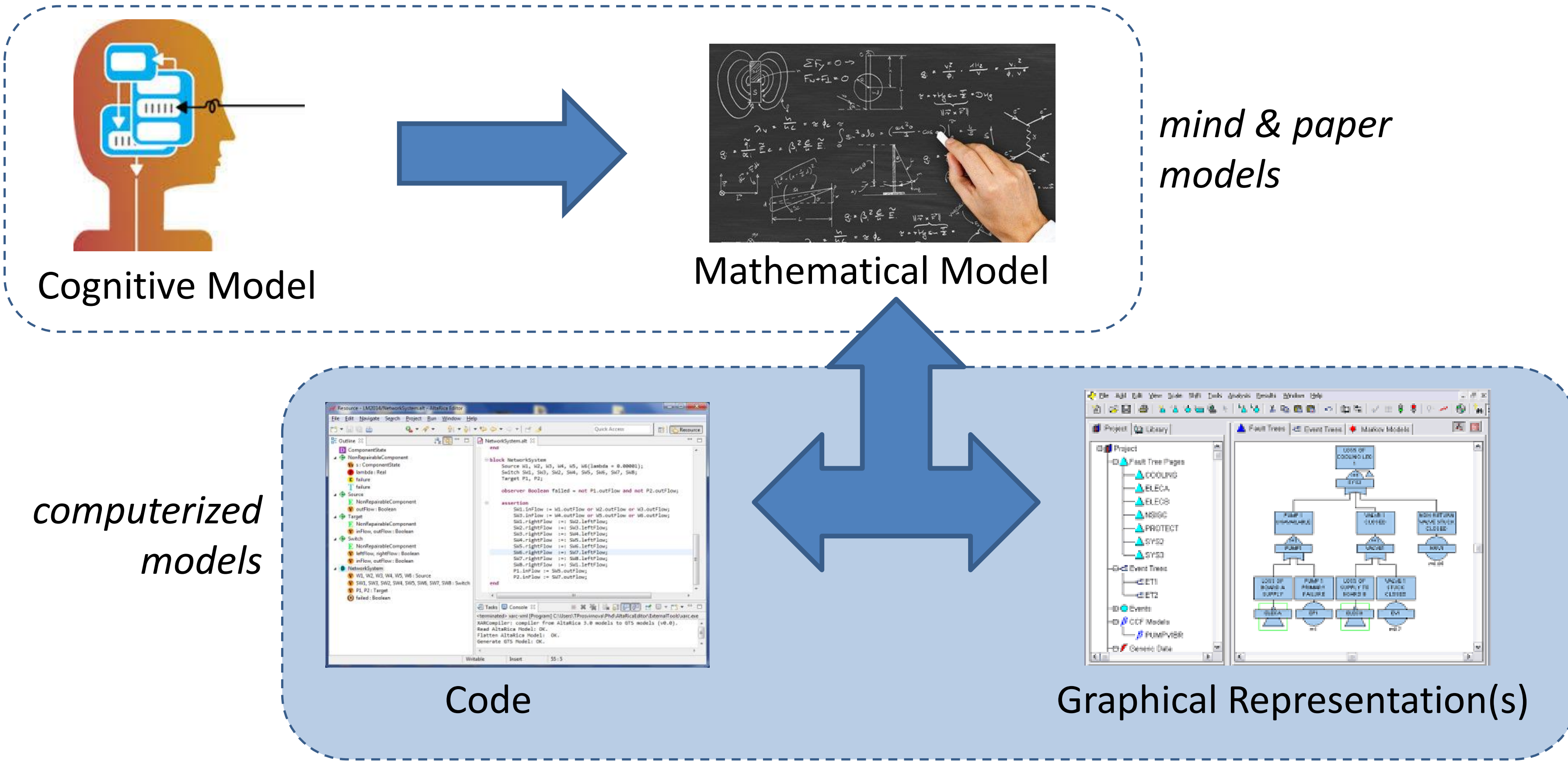
Models must be taken seriously and considered as **first class citizens**. This raises a number of challenges:

- Better understand the **nature** of models and their **roles** in industrial processes.
- Develop the "**Art of Modeling**"(*) in each and every engineering discipline.
- **Manage** models throughout the **life-cycle** of systems.
- Design tools and methods to support the **integration** of engineering disciplines/processes through the integration of models they produce.
- **Teach** and **give taste** of modeling to (future) engineers.
- …

The emerging science of complex systems is the science of models

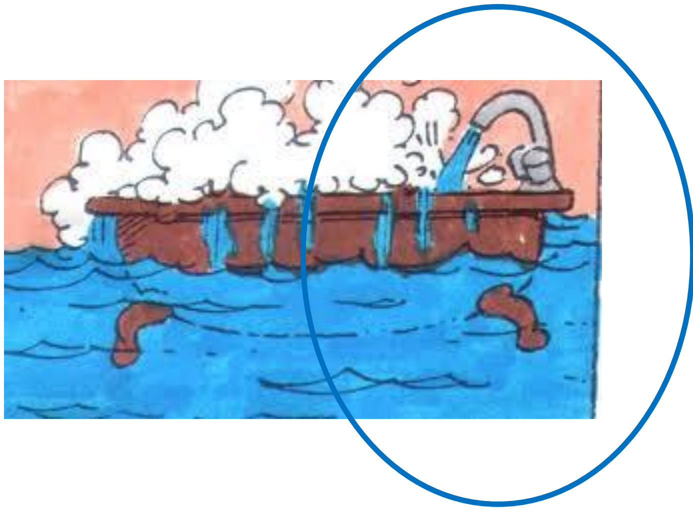
(*) In reference to Knuth's famous series of books about "The Art of Programming"

Models in Systems Engineering



Models are **working tools**, not (platonic) ideals the system should comply to.

Specific Purposes, Specific Models

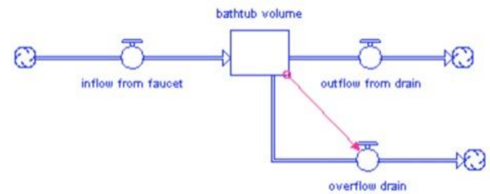


The **content** and the **level of abstraction** of a model depends on what is to be observed, i.e. on the **virtual experiments** to be performed on that model.

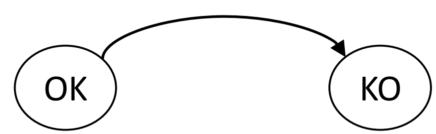
Fluid mechanics

$$\frac{\partial \vec{v}}{\partial t} + \left(\vec{v} \cdot \vec{\nabla}\right) \vec{v} = -\frac{1}{\rho}\vec{\nabla} p + \nu \nabla^2 \vec{v} + \vec{f}$$

Multiphysics simulation



Safety analyses



Insurance

Region	Premium price (Winter 2013/14)	Premium price (Winter 2014/15)	Percentage decrease year on year
London North West (NW)	£182.04	£149.05	-18.1%
Hereford (HR)	£113.42	£97.57	-14%
London West (W)	£157.24	£135.68	-13.7%
Enfield (EN)	£154.31	£133.61	-13.4%
Manchester (M)	£137.45	£121.22	-11.8%
Cambridge (CB)	£118.16	£104.45	-11.6%
Liverpool (L)	£138.86	£123.41	-11.1%
Southend-on-sea (SS)	£150.17	£133.64	-11%
Harrogate (HG)	£122.63	£109.32	-10.9%
Huddersfield (HD)	£128.56	£114.65	-10.8%

Thesis 1

The diversity of models is irreducible

Meaning and practical consequences:

- It is not possible to design all of the models of a system into a unified framework.
- Models are **not compositional**: the set of models of a system is not a model.
- Models designed for **system architecture*** are **not different** with that respect than models designed in other engineering disciplines.
- There **cannot be** such a thing as a **unique model** or even a **master model** of a complex system.

(*) We refer here to the meaning D. Krob gives to this term through the so-called "CESAMES approach"

Agenda

Model-Based Systems Engineering

Taxonomy of Models

Modeling Languages

Model-Based Safety Assessment

Model Synchronization

Taxonomy of Engineering Models

Models are designed at different level of abstraction, for different purposes and in different **modeling formalisms**.

Models to calculate performance indicators

Models to communicate amongst stakeholders

Models to generate artefacts (via code generation) or physical components (via additive manufacturing)

Informal models, even thought they are written in *standardized notations*, sometimes called *semi-formal*

Formal models, that essentially encode and organize (a given type of) mathematical equations

Thesis 2

There is an epistemic gap between informal and formal models

Meaning and practical consequences:

- Informal models and formals models have radically different natures and purposes.
- **Both types** of models are **useful**.
- **Passing from informal** models **to formal** ones requires an **engineering process**. This process **cannot be automated**.
- As **informal models** are **computerized**, we can design tools to **process** them, but from a **syntactic perspective** (i.e. we can work on their form) as opposed to a semantic perspective (i.e. working on their meaning).

Agenda

Model-Based Systems Engineering

Taxonomy of Models

Modeling Languages

Model-Based Safety Assessment

Model Synchronization

Teaching Models Engineering

Conclusion & Perspectives

Models Engineering

Fact 1: To design a model, we need a **modeling language** (would it be purely graphical), just as to design a program, we need a programming language.

Fact 2: Models of a complex system cannot be simple, otherwise they cannot capture the complexity of the system* (information loss). Therefore, they need to be structured, documented, managed… in a word, we need an **engineering of models**.

Questions:

- What is a good modeling language?
- What is a good palette of modeling languages?
- How to manage versions and configurations of models through the life-cycle of systems?
- …

(*) Models of complex systems are simplex, in the sense of A. Berthoz.

Thesis 3

Behaviors + Structures = Models*

Meaning and practical consequences:

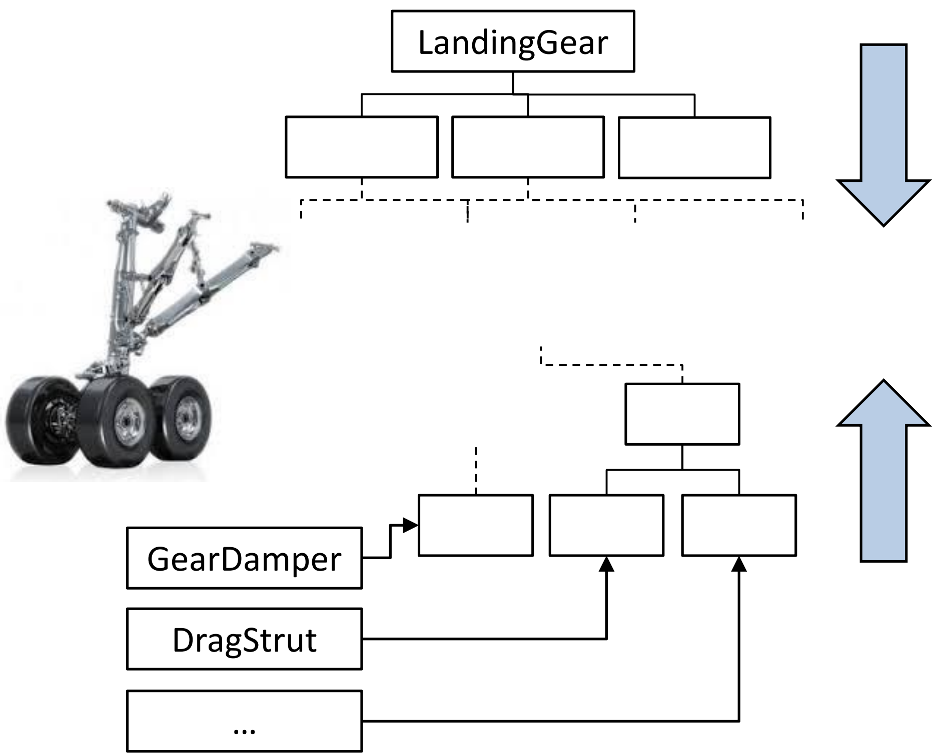
- Any modeling language is the combination of a **mathematical framework** to describe the behavior of the system under study and a **structuring paradigm** to organize the model.
- The choice of the **appropriate mathematical framework** for a model depends on which aspect of the system we want to study
- **Structuring paradigms** are to a very large extent **independent** of the chosen mathematical framework. They can be studied on their own.

(*) In reference to Wirth's seminal book "Algorithms + Data Structures = Programs"

S2ML

S2ML: **System Structure Modeling Language**

- A **structuring paradigm** that unifies the two dominant structuring paradigms for modeling languages, i.e. **object-orientation** and **prototype-orientation**.
- A **modeling language** on its own, dedicated to architecture description.



- Top-down model design
- System level
- Reuse of modeling patterns
- Prototype-Orientation

system architecture

safety

- Bottom-up model design
- Component level
- Reuse of modeling components
- Object-Orientation

Multiphysics simulation

Agenda

Model-Based Systems Engineering

Taxonomy of Models

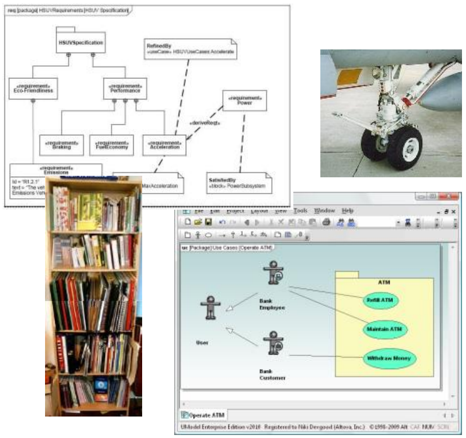
Modeling Languages

Model-Based Safety Assessment

Model Synchronization

Issues with "Classical" Safety Models

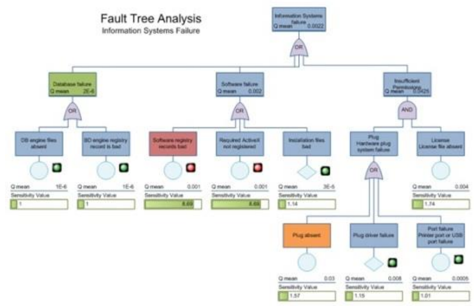
Systems Specifications



Modeling

Requirements, Certification

Models



FMEA, Fault Trees, Markov Chains, Stochastic Petri Nets…

Virtual Experiments

- Failure Scenarii
- Failure Probabilities

Classical modeling formalisms used for safety analyses lack of expressive power and/or are very close to mathematical equations (lack of structure).

→ **Distance** between **systems specifications** and **models**;

→ Models are **hard to design** and even **harder** to **share with stakeholders** and to **maintain** throughout the **life-cycle** of systems.

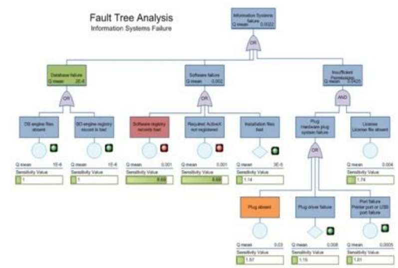
→ Very **conservative** approximations

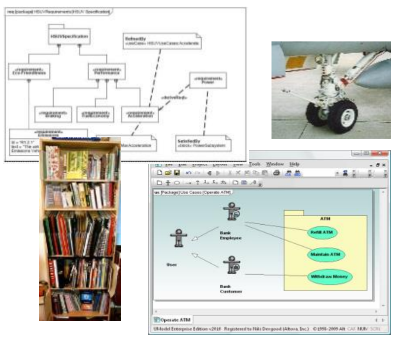
The Promise of MBSA

Modeling systems at **higher level** so to reduce the distance between systems specifications and models (without increasing the complexity of calculations).

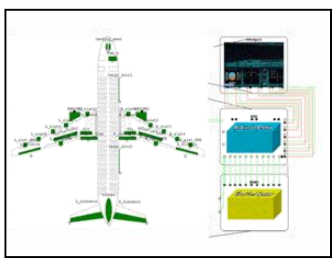
- Ability to **animate/simulate** models: to ease **model validation** and **discussions with stakeholders**;
- One model, several safety goals: to ease **versioning**, **configuration** and **change** management;
- One model, several assessment tools: **versatility** of assessments, **quality-assurance** of results;
- Fine grain analyses: to **avoid over-pessimism**.

Models





Systems Specifications



```
class HydraulicPump
  Boolean working (init = false);
  event failure (delay = exponential(lambda));
 transition
   failure: working -> working := false;
end
```

AltaRica 3.0

Thesis 4

Discrete Event Systems are the (only) suitable mathematical framework to describe behaviors at system level

Meaning and practical consequences:

- Safety models are **event-driven** and **probabilistic** in essence. Any safety model can be seen as a probabilistic discrete event systems (probabilistic state automaton).
- This applies to system architecture behavioral models as well (but without probabilistic aspect).
- Attempts to use other mathematical frameworks are doomed to failure.

Complexity of Calculations

Calculations of risk and safety related indicators are **extremely resource consuming**.

This is not a problem of technology. It has been **mathematically proven*** that they are **computationally intractable**.

Practical assessment tools perform **unwarranted approximations** that may impact strongly the significance of the result.

Safety models result always of a **tradeoff** between the accuracy of the description and the ability to perform calculations. Finding a suitable compromise for a given system is the expertise of the safety analyst.

(*) By L. Valiant in 1979. Valiant's work is one in a long series of impossibility results, starting from K. Goedel's incompleteness theorem and going through the whole computational complexity theory (including the seminal work of A. Turing).

Agenda

Model-Based Systems Engineering

Taxonomy of Models

Modeling Languages

Model-Based Safety Assessment

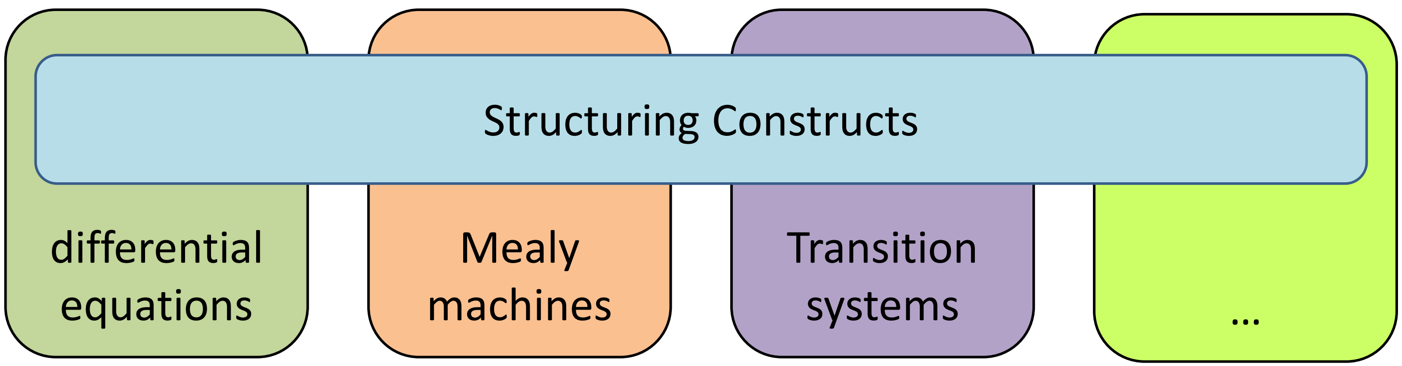
Model Synchronization

Model Comparison

The **design/production/operation/decommissioning** of a system involves the design of dozens if not hundred of models. These models are designed by **different teams** in **different languages** at **different levels of abstraction**, for **different purposes.** They have also **different maturities**.

The question is how to ensure that they are speaking about the **same system**, i.e. to **synchronize** them.

As the **behavioral part** of models is **purpose-dependent**, the main way to compare models is to compare their **structure**.

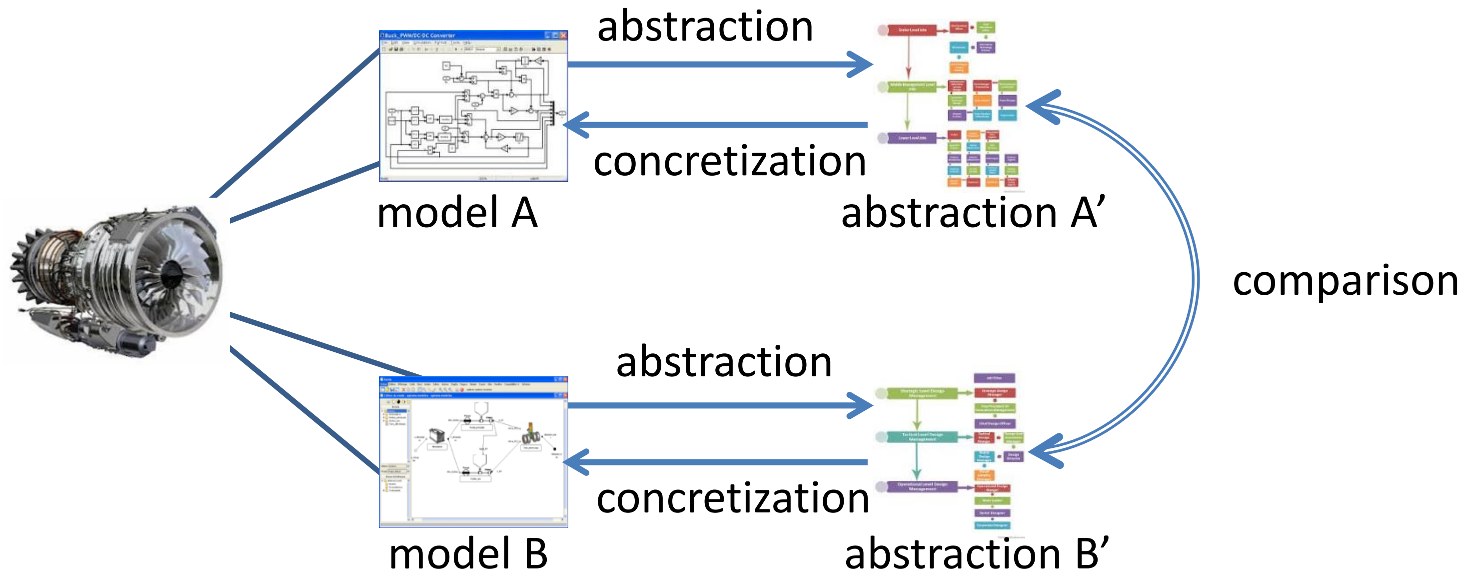


The **structure of models** reflects the **structure of the system**, even though to a **limited extent.**

Thesis 5

Abstraction + Comparison = Synchronization*

Meaning and practical consequences:



(*) Cousot’s abstract interpretation is thus the conceptual framework of model synchronization.