International Workshop on Reliability Engineering and Computational Intelligence, RECI 2020

New Challenges and Opportunities in Reliability Engineering of Complex Technical Systems

Prof. Antoine B. Rauzy

Department of Mechanical and Industrial Engineering
Norwegian University of Science and Technology
Trondheim, Norway

Chair Blériot-Fabre
CentraleSupélec / SAFRAN
Paris, France

One Observation, Two Questions

The observation:

> Software and control mechanisms become ubiquitous in nowadays technical systems.

The two questions:

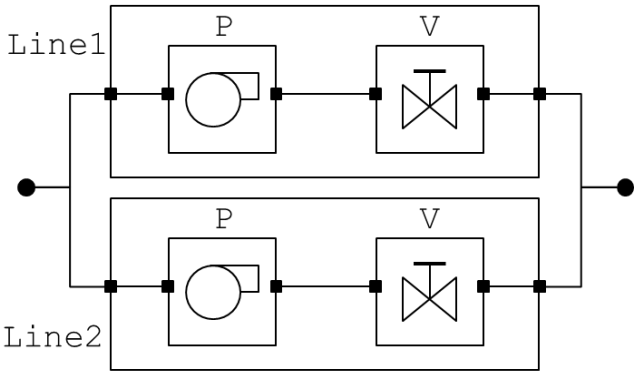
1. Are current modeling technologies for probabilistic risk/safety analysis, e.g. fault trees, still suitable to assess risks in new generations of systems?
2. Can we use the new capacities provided by information technologies to improve the probabilistic risk/safety analysis process?

Agenda

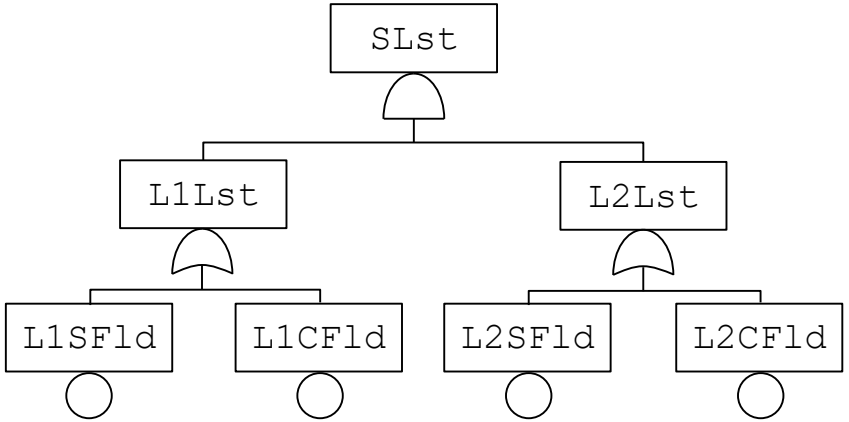
- (R)evolution in Reliability Engineering
- The S2ML+X Family of Languages
- The Dialectic of Expressive Power and Computational Complexity
- Model Synchronization
- Wrap-Up

Issues with Current Probabilistic Safety Analyses

System specifications



Reliability models



- Combinatorial models (fault trees, reliability block diagrams, event trees) lack of expressive power to represent faithfully reconfigurations, control mechanisms, time dependencies…;
- States/events models (Markov chains, stochastic Petri nets) lack of structure;
- All are very distant from system specifications, making model hard to author, to share with stakeholders and to maintain through the life-cycle of systems.

(R)evolution in Reliability Engineering

Today:

Mechanical systems

Recording of failures

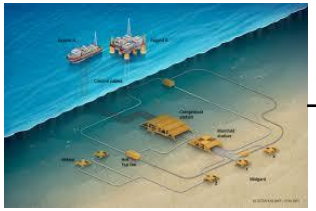
Local reliability databases

λ = 1.23e-6

Parametric distributions

Ad-hoc models, e.g. fault trees

Tomorrow:

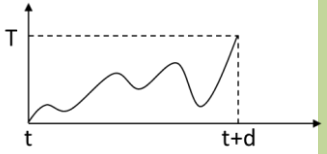


Health monitoring

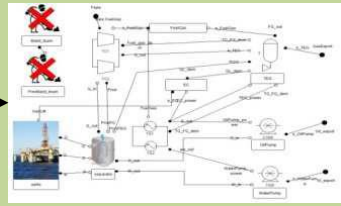
Sensors

Cyber-physical systems

Distributed health condition databases



Learned distributions



Behavioral models

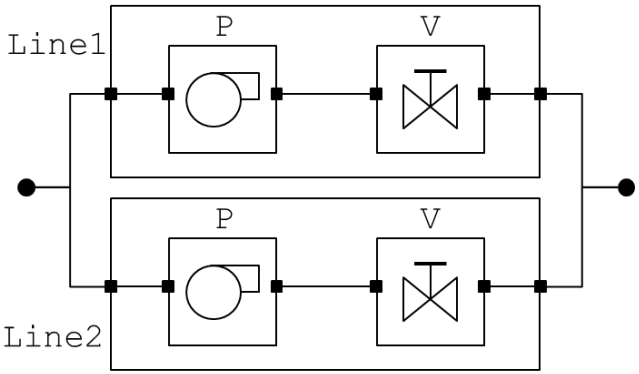
Agenda

- (R)evolution in Reliability Engineering
- The S2ML+X Family of Languages
- The Computational Complexity Challenge
- Model Synchronization
- Wrap-Up

Characteristics of Behavioral Models

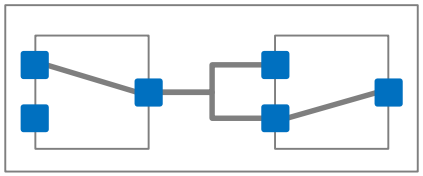
Behavior + Architecture = Model

- Any modeling language is the combination of a **mathematical framework** to describe the behavior and a **structuring paradigm** to organize the model.
- The choice of the **suitable mathematical framework** 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.



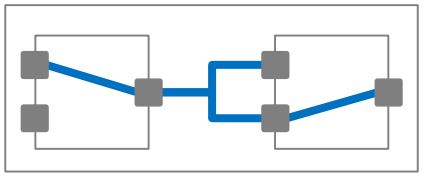
S2ML: Meta-Model of Behavioral Models

Port



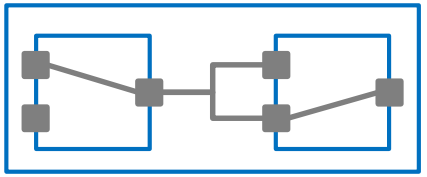
Variable, event…

Connection



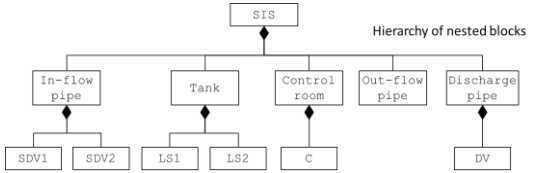
Equation, transition…

Container



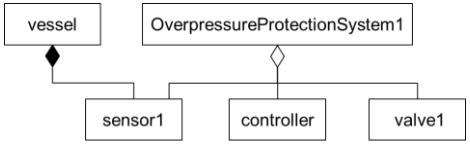
Model, component…

Composition



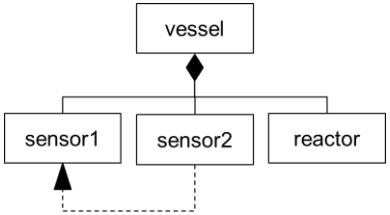
Is-part-of

Aggregation

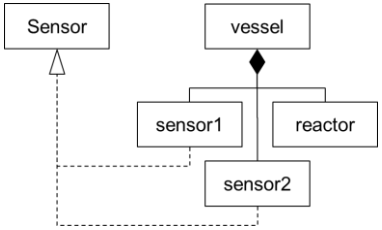


Uses

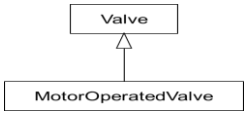
Prototype/Cloning



Class/Instantiation



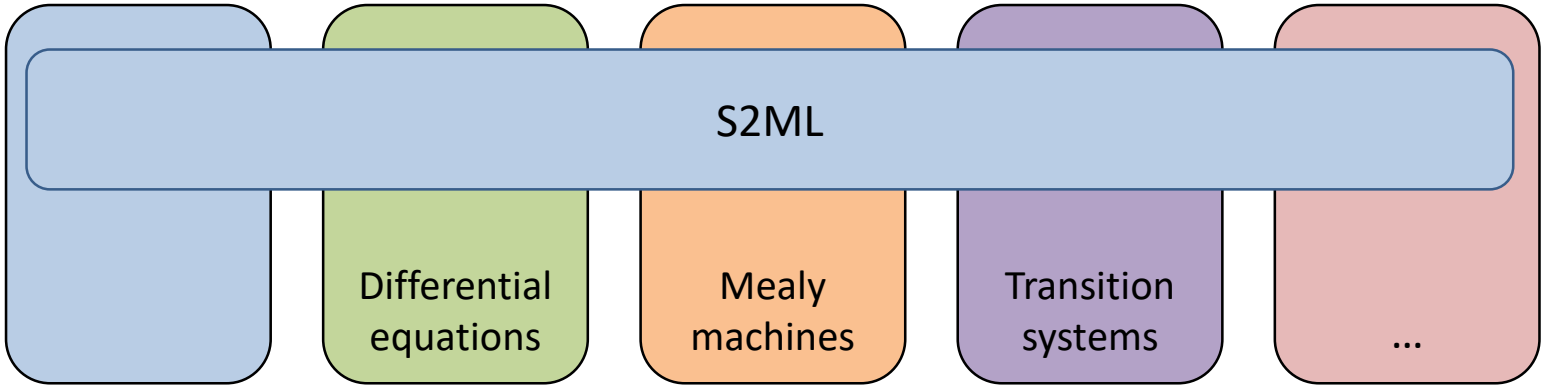
Inheritance



Is-a

The S2ML+X Promise

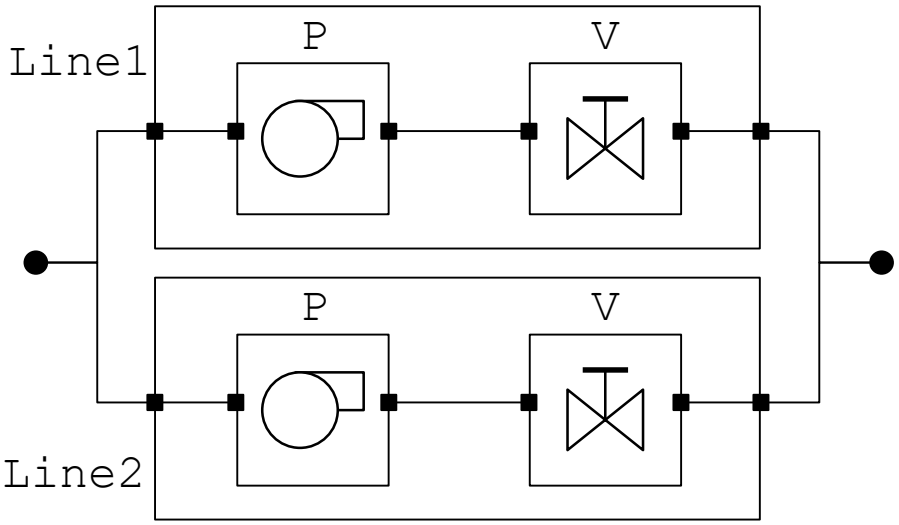
S2ML (System Structure Modeling Language): a coherent and versatile set of **structuring constructs** for any behavioral modeling language.



- The **structure of models** reflects the **structure of the system**, even though to a **limited extent.**
- **Structuring** helps to design, to debug, to share, to maintain and to align heterogeneous models.

S2ML + Stochastic Boolean Equations

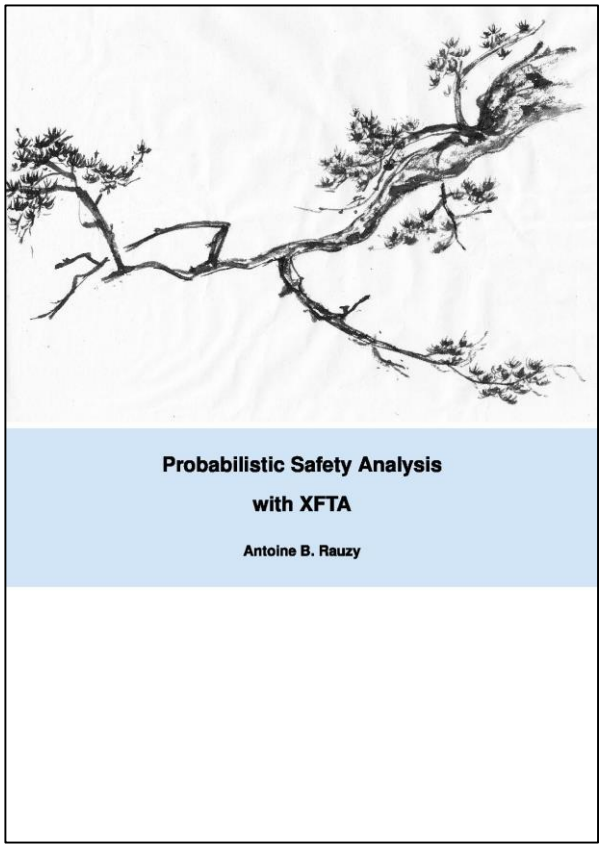
Enhancing classical **reliability models** (fault trees, reliability block diagrams) with the **expressive power of object-orientation** at **no algorithmic cost**




```
class Pump
    extends RepairableUnit
    …
end

block System
    block Line1
        Pump P;
        …
    end
    clones Line1 as Line2;
    …
end
```

```
Line1.in := in;
Line1.P.in := Line1.in;
Line1.P.out := Line1.P.in and not Line1.P.failed;
…
```

Release of the β-version of XFTA 2 + XFTA Book

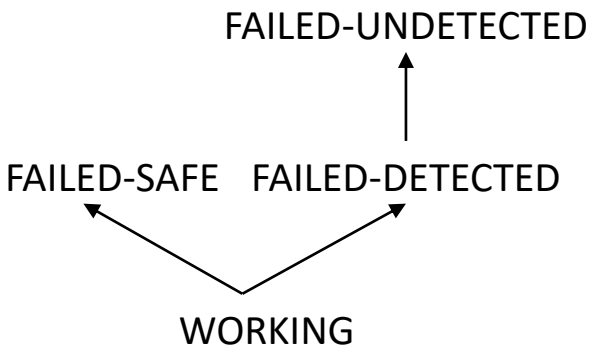


XFTA 2:

- Calculation engine for fault trees and related models.
- Input language: S2ML+SBE
- State of the art assessment algorithms: as of today most efficient calculation engine
- Calculation of all usual risk indicators:
  - Top event probability
  - Importance factors
  - Sensitivity analyses
  - Approximation of system reliability
  - Safety integrity levels
- Free of use, including for commercial purposes.

S2ML + Finite Degradation Structures

Lifting-up all classical concepts of reliability engineering to **multi-valued logics** and giving these logics the **expressive power of object-orientation**.



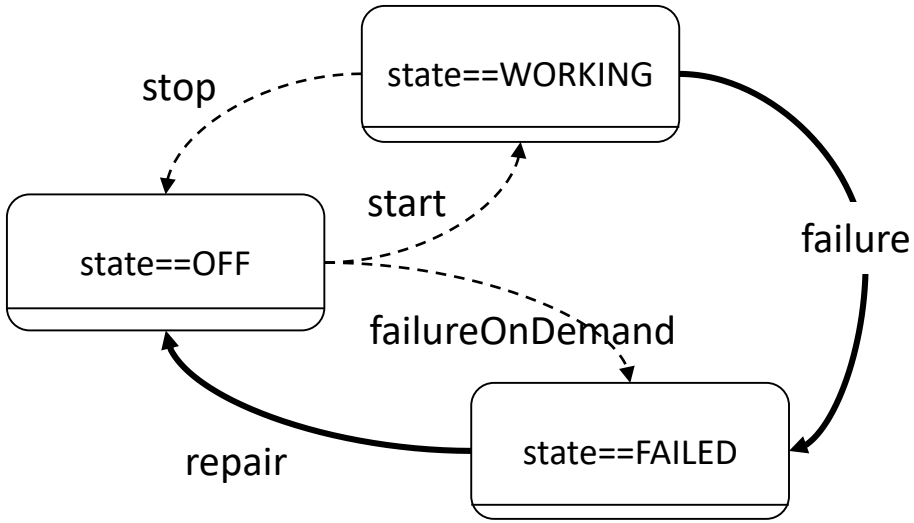
||	W	Fs	Fd	Fu
W	W	W	W	W
Fs	W	Fs	Fs	Fs
Fd	W	Fs	Fd	Fd
Fu	W	Fs	Fd	Fu

```
domain IEC61508
    {WORKING, FAILED_SAFE,
     FAILED_DETECTED,
     FAILED_UNDETECTED}
    WORKING<FAILED_SAFE,
    WORKING<FAILED_DETECTED,
    …

operator Parallel
    …
end
```

AltaRica 3.0 (S2ML + Guarded Transitions Systems)

Guarded Transitions Systems:

- Are a probabilistic Discrete Events System formalism.
- Are a compositional formalism.
- Generalize existing mathematical framework.
- Take the best advantage of existing assessment algorithms.



OpenAltaRica
SYSTEM SAFETY ANALYSIS TECHNOLOGY
System X
INSTITUT DE RECHERCHE TECHNOLOGIQUE

SAFRAN

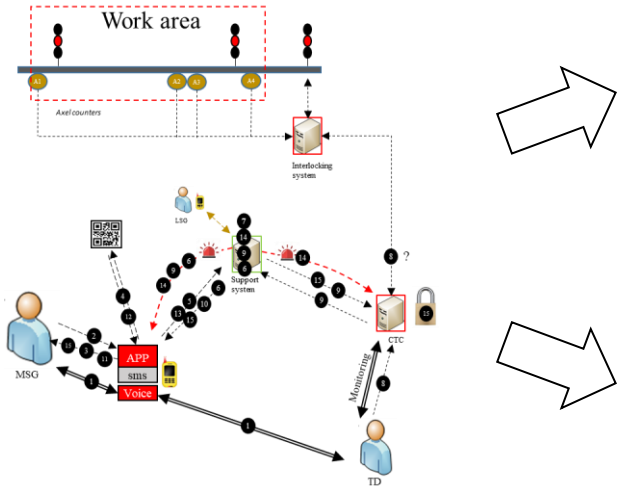
THALES

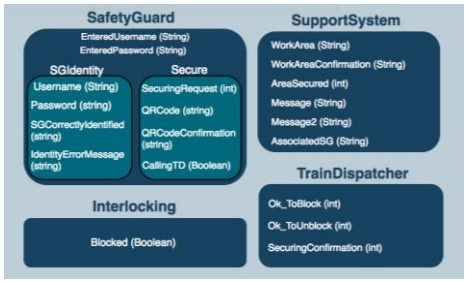
AIRBUS

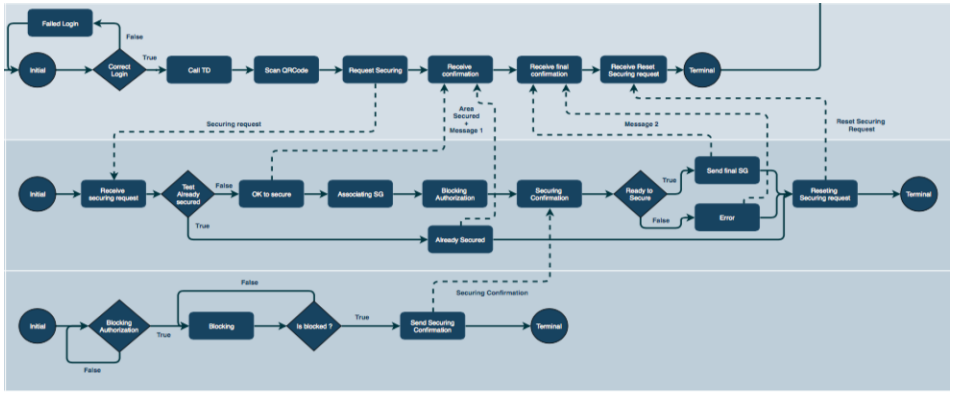
Scola (S2ML + Process Algebra)

Scenario-oriented modeling methodology

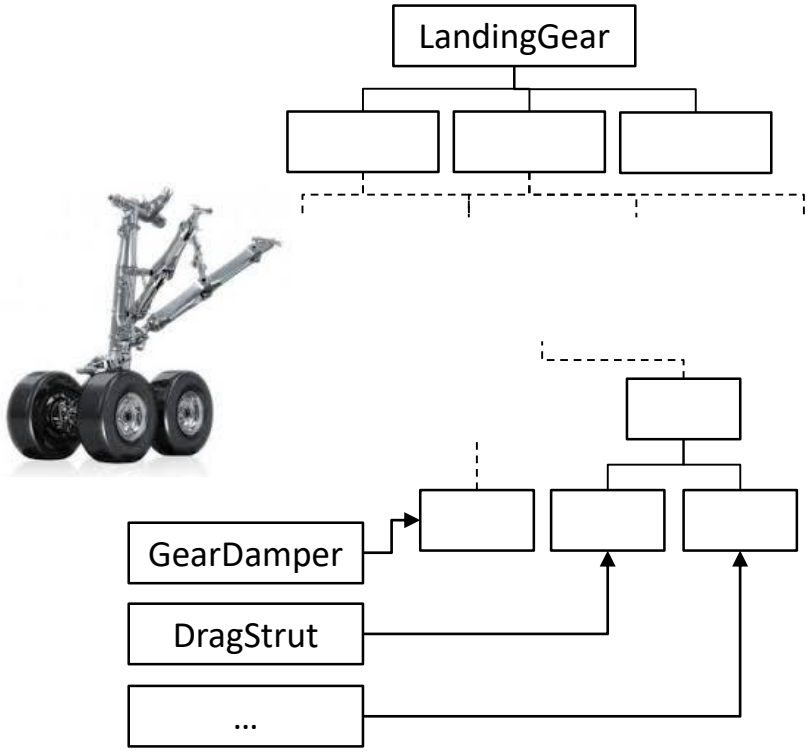
- Architecture description
- Dynamic modification of components
- Moving components
- Dynamic creation/deletion of components







Modeling Approaches



- 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

Models as Scripts

The **model "as designed"** is a script to build the **model "as assessed"**.

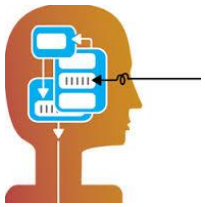
```
domain WF {WORKING, FAILED} WORKING<FAILED;

operator Series arg1 arg2 =
   (if (and (eq state1 WORKING) (eq state2 WORKING))
      WORKING
      FAILED);

class Component
   WF state(init = WORKING);
   WF in, out(reset = WORKING)
   probability state FAILED = (exponentialDistribution lambda (missionTime));
   parameter Real lambda = 1.0e-3;
   assertion
      out := (Series in state);
end
```

Complex models can be built using **libraries** of **reusable modeling components** and **modeling patterns**.

Agenda

- (R)evolution in Reliability Engineering
- The S2ML+X Family of Languages
- The Dialectic of Expressive Power and Computational Complexity
- Model Synchronization
- Wrap-Up

Virtual Experiments in Reliability Engineering



System (existing or under design)

Mathematical framework

analyze

draw conclusions

Models *in abstracto*

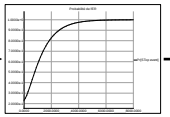
analyst

Models & experiments *in silico*

design model

model

perform experiments



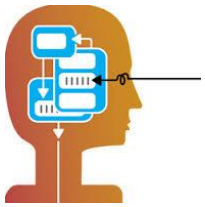
results

Modeling language

Assessment algorithm

Computational cost

Virtual experiments in Reliability Engineering



System (existing or under design)

$$\frac{dS}{dt} = \alpha_1 B^{\phi_{11}} \left(1 - \frac{S}{K_S}\right)_+^{\phi_{12}} \quad (1)$$

$$\frac{dP}{dt} = \alpha_2 S^{\phi_{21}} \left(1 - \frac{S}{K_S}\right)_+^{\phi_{22}} + \alpha_3 P^{\phi_{31}} \left(1 - \frac{S}{K_S}\right)_+^{\phi_{32}} - \beta_1 P^{\phi_{33}} C^{\phi_{34}} - \delta P, \quad (2)$$

$$\frac{dB}{dt} = \beta_1 P^{\phi_{41}} C^{\phi_{42}} - \beta_2 B^{\phi_{43}} - \alpha_4 B^{\phi_{44}} \left(1 - \frac{S}{K_S}\right)_+^{\phi_{45}}, \quad (3)$$

$$\frac{dC}{dt} = \alpha_5 S^{\phi_{51}} P^{\phi_{52}} (r + B)^{\phi_{53}} \left(1 - \frac{S}{K_S}\right)_+^{\phi_{54}} - \beta_2 C^{\phi_{55}}, \quad (4)$$

$$\frac{dz}{dt} = -k_1 C + k_2 B. \quad (5)$$

analyze

draw conclusions

Models *in abstracto*

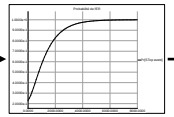
analyst

Models & experiments *in silico*

design model

model

perform experiments



results

A model results always of a **tradeoff** between the **accuracy of the description** and the **computational cost** of virtual experiments.

Classes of Modeling Languages

Combinatorial Formalisms
- Fault Trees
- Event Trees
- Reliability Block Diagrams
- Finite Degradation Structures

States Automata
- Markov chains
- Dynamic Fault Trees
- Stochastic Petri Nets
- …

Process Algebras
- Agent-based models
- Process algebras
- Python/Java/C++
- …

Expressive power

States → States + transitions → Deformable systems

Complexity of assessments

#P-hard but reasonable polynomial approximation → PSPACE-hard → Undecidable

Difficulty to design, to validate and to maintain models

Best in Class Modeling Languages

Combinatorial Formalisms

Boolean models:

- Stochastic Boolean Equations
- S2ML+SBE
- XFTA

Multistate systems:

- Finite degradation structures
- S2ML+FDS
- Emmy (proof of concept)

States Automata

- Guarded Transition Systems
- S2ML+GTS = AltaRica 3.0
- AltaRica Wizard

Process Algebras

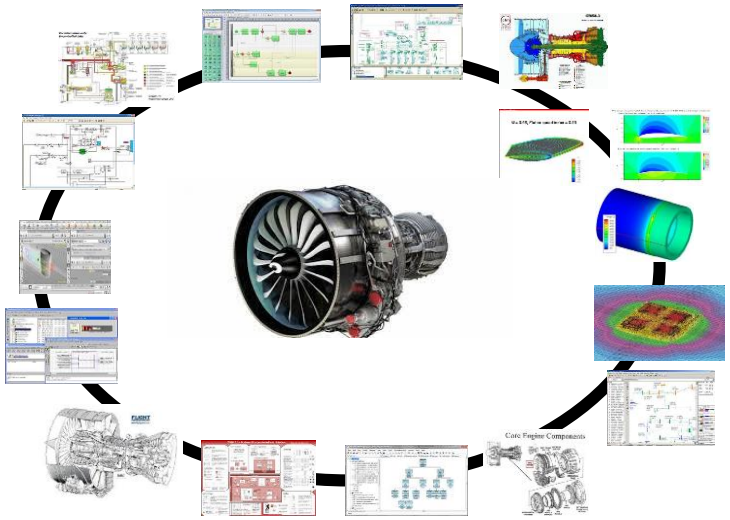
- Stochastic Process Algebras
- S2ML+SPA = Scola
- Scola Simulator (proof of concept)

Agenda

- (R)evolution in Reliability Engineering
- The S2ML+X Family of Languages
- The Dialectic of Expressive Power and Computational Complexity
- Model Synchronization
- Wrap-Up

Model Diversity

Models are designed by **different teams** in **different languages** at **different levels of abstraction**, for **different purposes**, making **different approximations**. They have also **different maturities**.



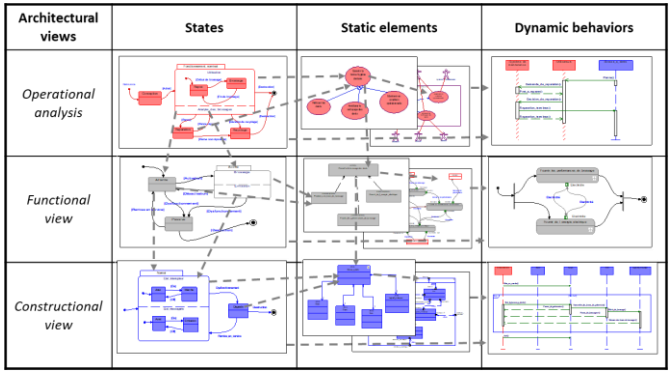
complexity → simplexity

The **diversity** of models is **irreducible.**

Pragmatic versus Formal Models

System Architecture

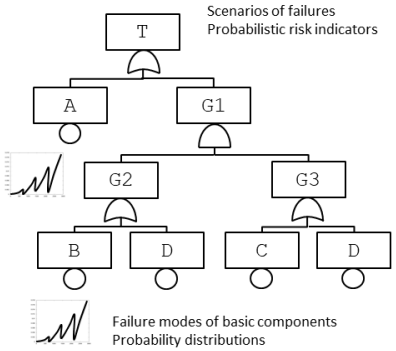
Models to communicate
amongst stakeholders



Pragmatic proof that there exists a system that meets the given specification.

Reliability Engineering

Models to calculate
performance indicators



Formal proof that the specified system is reliable enough to be operated.

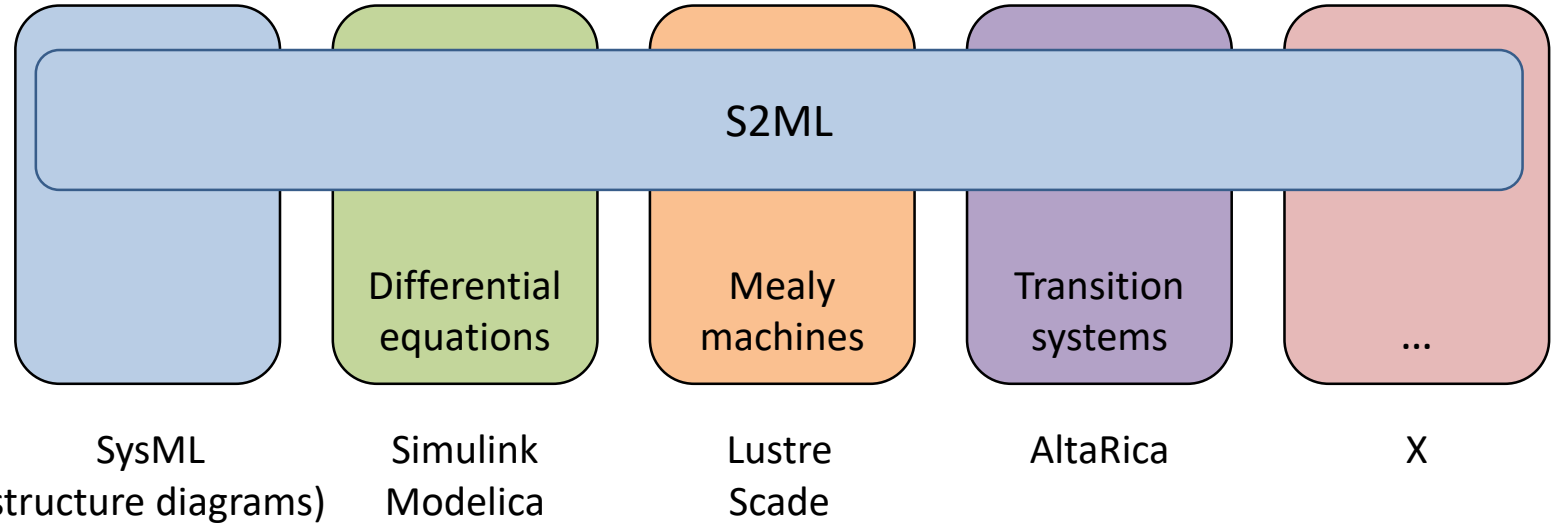
Epistemic gap

Alignment of Heterogeneous Models

Models are designed by **different teams** in **different languages** at **different levels of abstraction**, for **different purposes.** They have also **different maturities**.

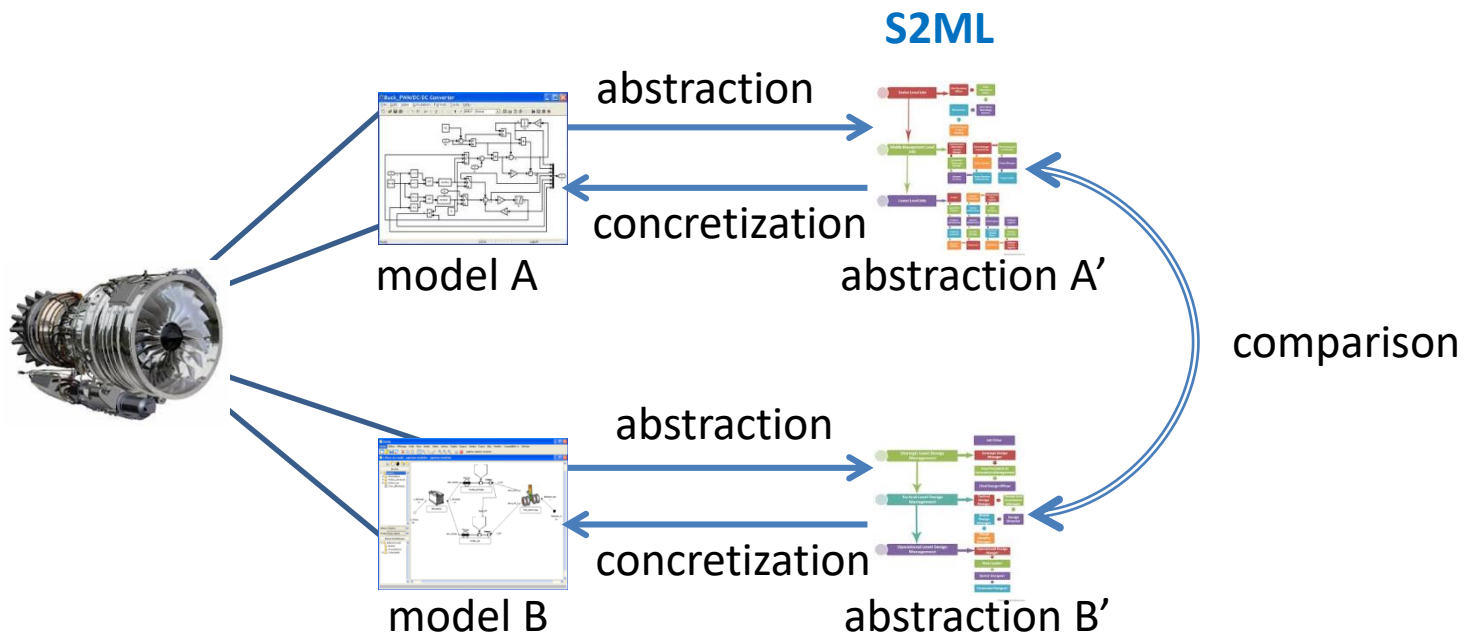
The question is how to ensure that they are "speaking" about the **same system**, i.e. to **align** them.

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



Model Synchronization

Abstraction + Comparison = Synchronization



How to agree on disagreements?

Agenda

- Introduction
- The S2ML+X Family of Languages
- The Dialectic of Expressive Power and Computational Complexity
- Model Synchronization
- Wrap-Up

Wrap-Up & Conclusion

- "Traditional" modeling approaches in reliability engineering are **no longer sufficient**:
  - Because the **systems** we are dealing with are **more complex**.
  - Because **new information technologies** open **new opportunities**.
  - Because **reliability models** should be **integrated** with models from other engineering disciplines.

- **Huge benefits** can be expected from a full-scale deployment of model-based systems engineering. However, this requires:
  - To set up solid **scientific foundations** for **models engineering**.
  - To **bring to maturity** some **key technologies**.

- The **biggest challenge** is to **train new generation of engineers**:
  - With skills and competences in **discrete mathematics** and **computer science**, and
  - With skills and competences in **system thinking**, and
  - With skills and competences in **specific application domains**.

Selected Publications (1)

General purpose articles:

- Rauzy A. (2018) Notes on Computational Uncertainties in Probabilistic Risk/Safety Assessment. Entropy. MDPI. 20:3. doi:10.3390/e20030162.

S2ML:

- Rauzy A and Haskins C. (2018) Foundations for Model-Based Systems Engineering and Model-Based Safety Assessment. Journal of Systems Engineering. Wiley Online Library. doi:10.1002/sys.21469.
- Batteux M, Prosvirnova T and Rauzy A. (2018) From Models of Structures to Structures of Models. IEEE International Symposium on Systems Engineering (ISSE 2018). IEEE. Roma, Italy. October. doi:10.1109/SysEng.2018.8544424. Best paper award.

Boolean models:

- Rauzy A. (2001) Mathematical Foundation of Minimal Cutsets. IEEE Transactions on Reliability. IEEE Reliability Society. 50:4. pp. 389–396. December, doi:10.1109/24.983400.
- Rauzy A. (2008) BDD for Reliability Studies. Handbook of Performability Engineering. Krishna B. Misra Ed.. Elsevier. Amsterdam, the Netherlands. ISBN 978-1-84800-130-5. pp. 381–396. 2008.
- Rauzy A. (2020) Probabilistic Safety Analysis with XFTA. AltaRica Association. Les Essarts le Roi, France. ISBN 978-82-692273-0-7.

Finite degradation structures:

- Rauzy A. and Yang L. (2019) Finite Degradation Structures. Journal of Applied Logics - IfCoLog Journal of Logics and their Applications. College Publications. 6:7. pp. 1471–1495.
- Rauzy A. and Yang L. (2019) Decision Diagram Algorithms to Extract Minimal Cutsets of Finite Degradation Models. Information. MDPI. 10:368. pp. 1–28. doi:10.3390/info10120368.

Selected Publications (2)

AltaRica:

- Rauzy A. (2008) Guarded Transition Systems: a new States/Events Formalism for Reliability Studies. Journal of Risk and Reliability. Professional Engineering Publishing. 222:4. pp. 495–505. doi:10.1243/1748006XJRR177.
- Batteux M., Prosvirnova T. and Rauzy A. (2017) AltaRica 3.0 Assertions: the Why and the Wherefore. Journal of Risk and Reliability. Professional Engineering Publishing. September. doi:10.1177/1748006X17728209.
- Batteux M, Prosvirnova T and Rauzy A. (2019) AltaRica 3.0 in 10 Modeling Patterns. International Journal of Critical Computer-Based Systems. Inderscience Publishers. 9:1-2. pp. 133–165. doi:10.1504/IJCCBS.2019.098809.
- Prosvirnova T. and Rauzy A. (2015) Automated generation of Minimal Cutsets from AltaRica 3.0 models. International Journal of Critical Computer-Based Systems. Inderscience Publishers. 6:1. pp. 50–79. 2015 doi:10.1504/IJCCBS.2015.068852.
- Brameret P.-A., Rauzy A. and Roussel J.-M. (2015) Automated generation of partial Markov chain from high level descriptions. Reliability Engineering and System Safety. Elsevier. 139. pp. 179–187. doi:10.1016/j.ress.2015.02.009

Model synchronization:

- Legendre A., Lanusse A. and Rauzy A. (2016) Directions towards supporting synergies between design and probabilistic Safety assessment activities: illustration on a fire detection system embedded in a helicopter. Proceedings PSAM'13. IPSAM. Seoul, South-Korea.
- Batteux M., Prosvirnova T., Rauzy A. (2019) Model Synchronization: A Formal Framework for the Management of Heterogeneous Models. Model-Based Safety and Assessment. Yiannis Papadopoulos, Koorosh Aslansefat, Panagiotis Katsaros and Marco Bozzano Ed.. Springer. ISBN 978-3-030-32871-9. 11842. pp. 157–172. Thessaloniki, Greece.
- Batteux M., Choley J.-Y., Mhenni F., Prosvirnova T. and Rauzy A. (2019). Synchronization of system architecture and safety models: a proof of concept. Proceedings of the IEEE 2019 International Symposium on Systems Engineering (ISSE). IEEE. Edinburgh, Scotland.