Towards Simulated System Architectures with Σ

Prof. Antoine B. Rauzy

Norwegian University of Science and Technology
Trondheim, Norway

&

Systemic Intelligence
Paris, France

Presentation

What is Σ?

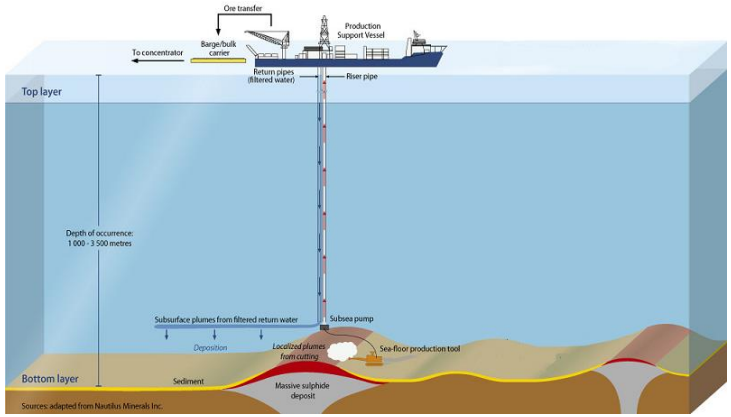
Σ (pronounce Sigma) is a **modeling language** dedicated to Model-Based Systems Engineering (MBSE)

Why Σ?

Σ aims at taking full advantage of **computerized simulations** to support **decision making**.

Typical (and First) Application: Subsea Mining

Collecting polymetallic nodes on the seabed of the Norwegian continual shelf.



A holistic problem:

- Technical choices
- Field data
- Economical data
- Regulations

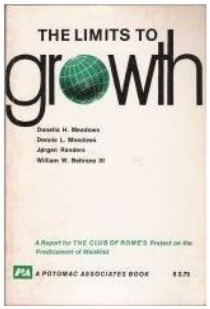
Key performance indicators to be assessed in the different scenarios:

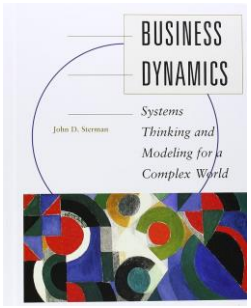
- Annual production of ore
- Energy consumption / economical viability
- Waste disposal
- Safety

Modeling Frameworks Landscape

MBSE graphical notations

System Dynamics



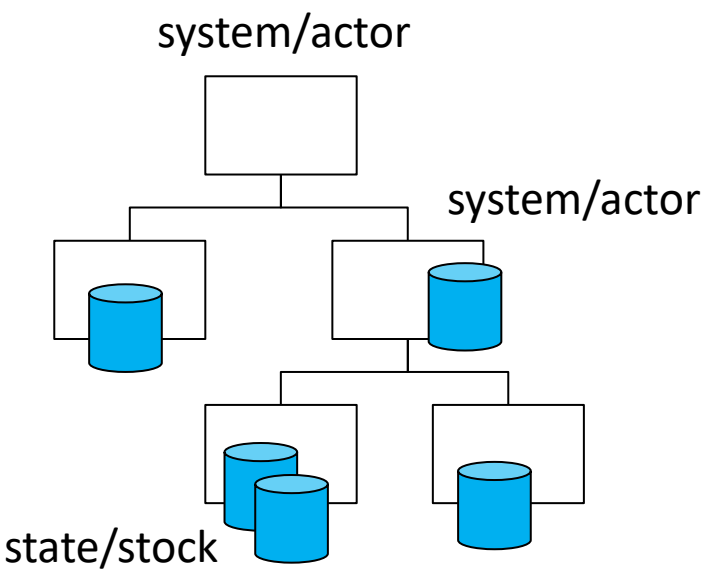


Model-based reliability analyses

Multiphysics simulation

The Σ Modeling Framework

Σ ontology: **Systems + Stocks + Activities**



Activities

- trigger
- action at start
- action at completion
- duration

Formal Language

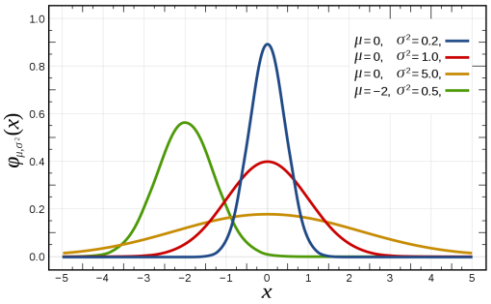
```
system SeaMiningWorld.SeaMiningSystem.MiningSupportVessel
      // Location
      parameter float distanceToHarbor(unit = "NM") = 400;
      // Cargo
      parameter float storageCapacity(unit = "t") = 39000;
      float totalStock(init = 0, unit = "t");
      float rawOreStock(init = 0, unit = "t");
      float dewateredOreStock(init = 0, unit = "t");
      float waterStock(init = 0, unit = "t");
      observer Cargo(unit = "t") = totalStock;
      indicator CargoIndicator(mean = true, standardDeviation = true) = mean(Cargo);
      // Bulk vessels
      parameter float bulkVesselDemandThreshold(unit = "t") = 1000;
      int bulkVesselTurn(init = 1);
end

activity SeaMiningWorld.SeaMiningSystem.MiningSupportVessel.DemandBulkVessel1
      trigger:
            return dewateredOreStock>=bulkVesselDemandThreshold and bulkVesselTurn==1
                  and not main.SeaMiningSystem.BulkVessel1.demandedOnSite;
      start:
            bulkVesselTurn = 0;
      completion:
            main.SeaMiningSystem.BulkVessel1.demandedOnSite = true;
      duration:
            return 0;
end
```

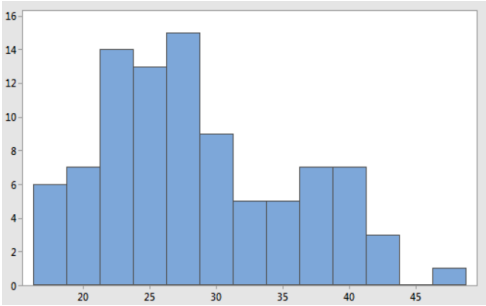
Embedding Data into Models

Results and durations of activities can be described by means of (probability) distributions

Parametric distributions

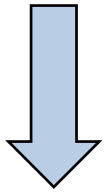


Empirical distributions



- On-line calibration of models
- Management of uncertainties
- …

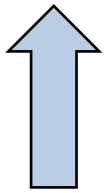
Full-Fledged Object-Oriented Language



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

system engineering

safety

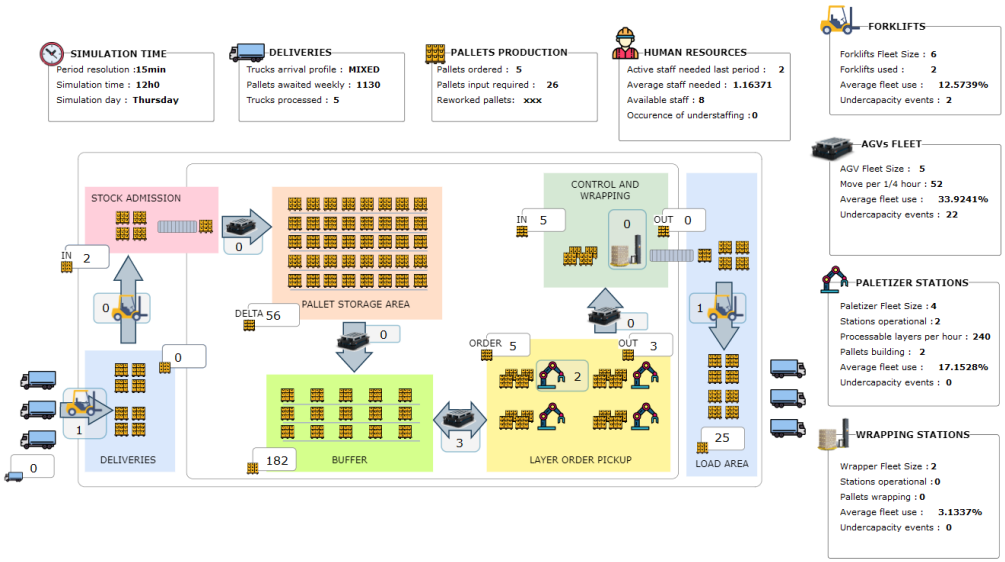


- Bottom-up model design
- Component level
- Reuse of modeling components
- Object-orientation

Multiphysics simulation

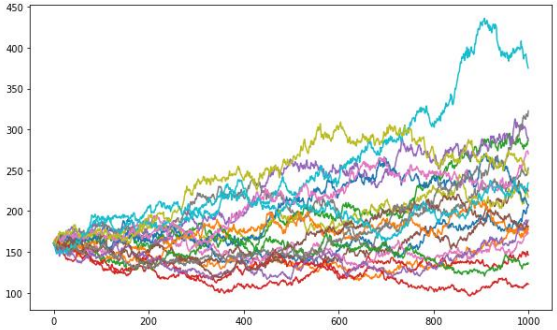
Interactive Simulation

- Σ models can be simulated graphically
- Video player metaphor
- Graphical animations described in WIDL



Automated Warehouse

Stochastic Simulation and Proof Scripts

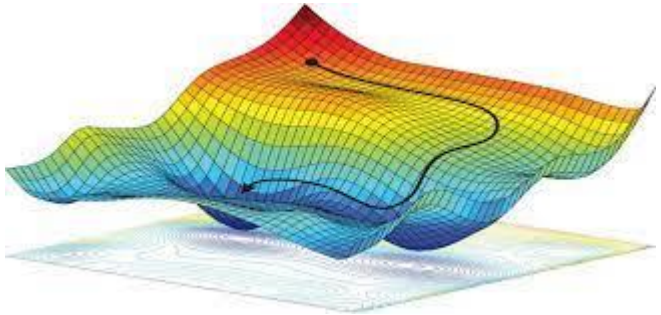


Monte-Carlo simulation

- Assessment of key performance indicators

Proof script

- Optimization
- What-if scenarios



Conclusion

What is Σ?

Σ is a **formal, object-oriented, modeling language** dedicated to MBSE.

Σ makes it possible to embed seamlessly **data** into models

Why Σ?

Σ aims at taking full advantage of **computerized simulations** to support **decision making**.

- Interactive simulations with dedicated graphical interfaces
- Stochastic simulations to assess key performance indicators
- Proof scripts for on-line calibration of models and optimization

Next generation of MBSE: Systemic Digital Twins