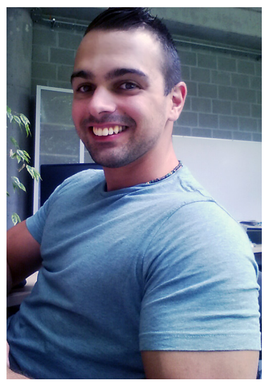
A Multi-Paradigm Modeling Foundation for Collaborative Multi-view Model/System Development

István Dávid
Modelling, Simulation and Design Lab
University of Antwerp - Flanders' Make, Belgium
istvan.david@uantwerpen.be | istvandavid.com



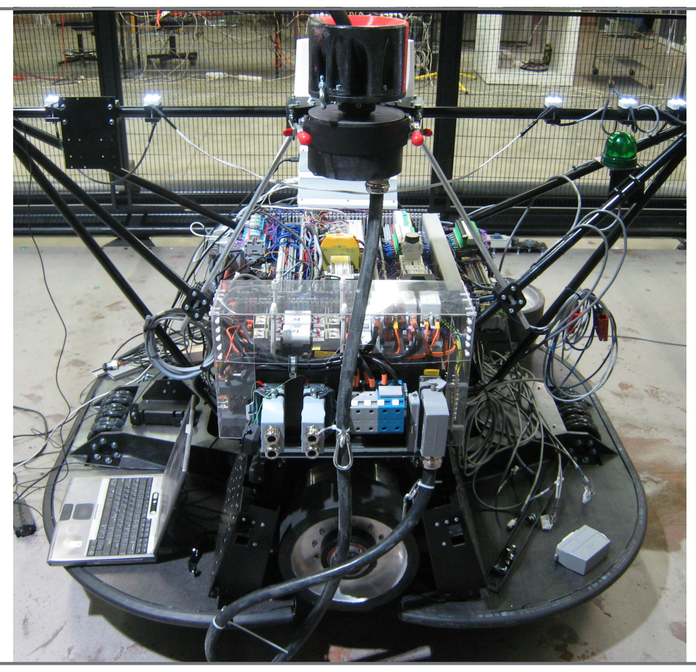
Summary

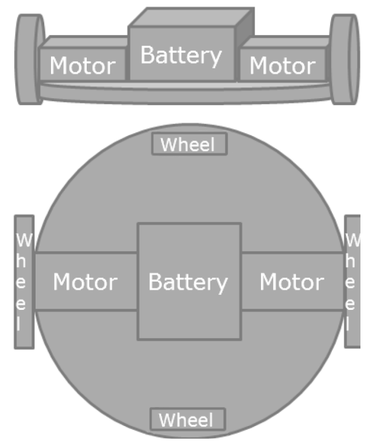
The complexity of current engineered systems has increased drastically over the last decades. To tackle to this complexity, these systems are typically developed in a collaboration involving stakeholders from different domains. Such collaborative endeavors are severely hindered by inconsistencies that arise due to semantic overlap between different models.

To cope with this problem, we propose an **inconsistency management** approach for better understanding how inconsistencies arise, evolve and how they should be managed, including **inconsistency tolerance**. The core of our approach is a rich **process modeling** formalism that allows modeling multiple aspects of the development workflow. The efficient and inconsistency-free cross-domain collaboration is achieved by **optimizing the original process** for various optimality criteria, such as consistency and development costs.

We support our approach with an open-source **prototype tool** for modeling engineering processes, specifying inconsistency patterns and their respective management alternatives; and for enacting the optimized process for orchestration and tool interoperability.

*An **inconsistency** is present if two or more statements are made that are not jointly satisfiable [such as a] failure of an equivalence test, non-conformance to a standard or constraint and the violation of physical or mathematical principles.* (Herzig)



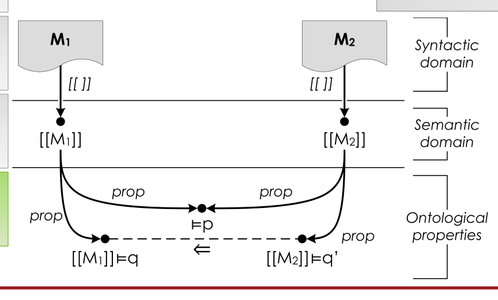


Automated guided vehicle (AGV), a pertinent example of truly heterogeneous, complex systems, result of an interplay between mechanical-, electrical-, control- and software engineering.

(In)consistency management: correctness vs efficiency

Multi-paradigm modeling: *Model everything at the most appropriate level(s) of abstraction using the most appropriate formalism(s) explicitly modeling processes.*

Explicit modeling of the system
Explicit evaluation
Consistency management
Correct product

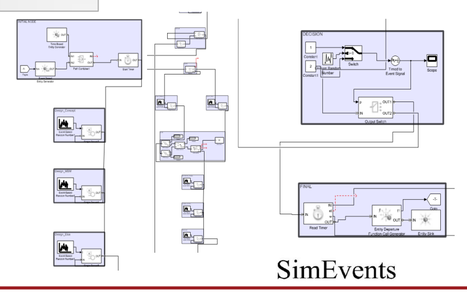


Correctness

- ▶Ensure that the system satisfies the required properties
- ▶Relationships in the semantic domains

Efficiency

- ▶Maximize the appropriate performance measures
- ▶Typically multi-objective problems

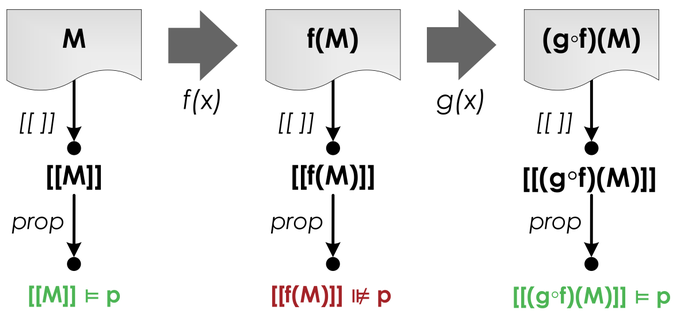


Explicit modeling of the process(es)
Explicit evaluation
Process optimization
Improved performance

Inconsistency tolerance

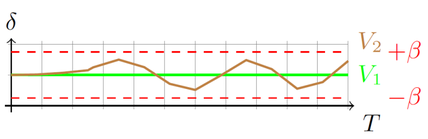
Temporal tolerance

Inconsistencies are stateful entities that might occur, evolve and later potentially disappear as the natural consequence of the design workflow.



- ▶Postponing resolution to a more appropriate phase
- ▶No resolution is required

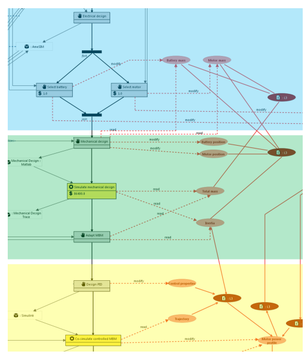
Parameter tolerance



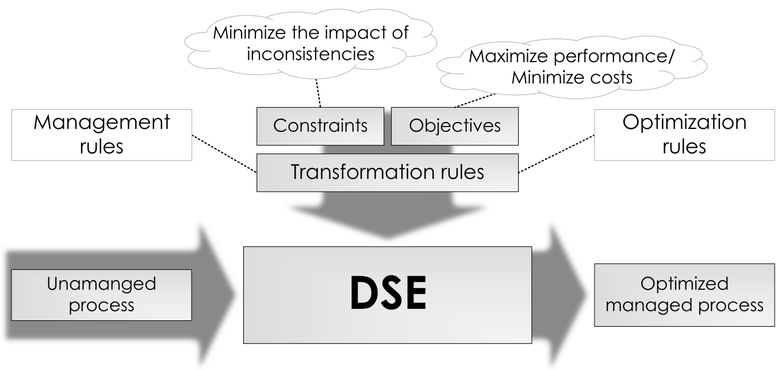
- ▶Relaxed parameter constraints
- ▶Composition of tolerance rules

Spatial tolerance

- ▶Smart scoping
- ▶Relaxed impact propagation

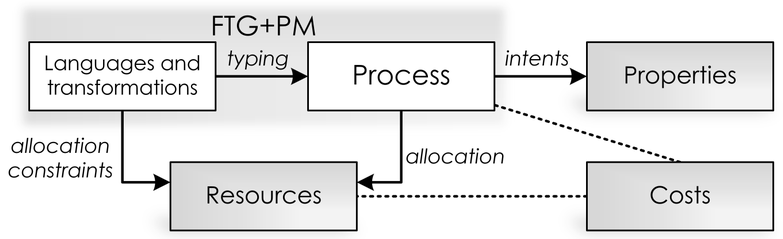


Process optimization
by Multi-objective Design Space Exploration



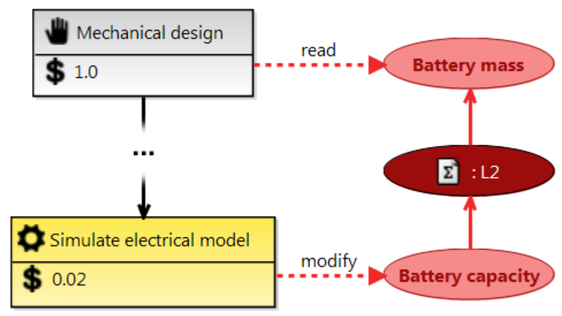
Process modeling

Formalism for modeling processes



- ▶Processes
- ▶Languages
- ▶Ontological properties
- ▶Resources
- ▶Costs (Performance metrics)

Example



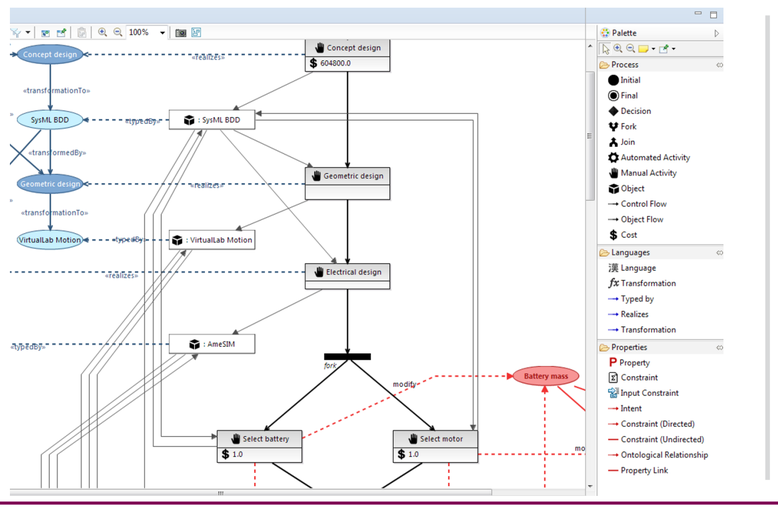
1. During the *Mechanical design*, the *Battery mass* property is used
2. The *Simulation of the electrical model* modifies the *Battery capacity*
3. The *Battery capacity* influences the *Battery mass*. (L2: sensitivity relationship)
4. The information used during the *Mechanical design* may be incorrect

Prototype tool

Specification and analysis

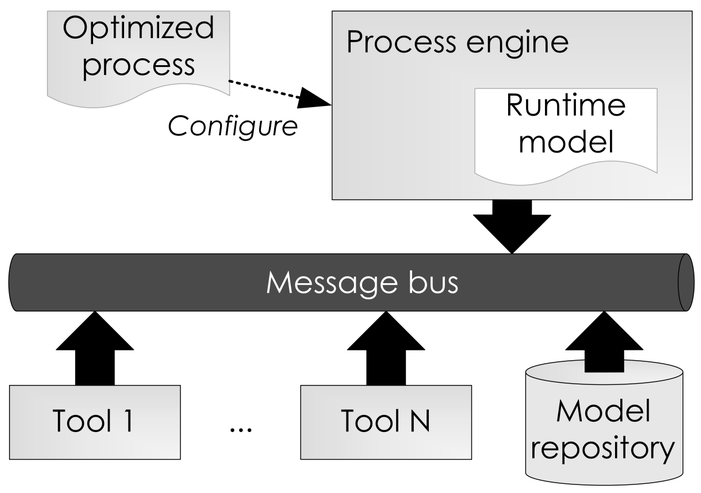
- ▶Ontology-aided approach
- ▶Reuse of domain-knowledge
- ▶Open-source (EPL)

https://github.com/david-istvan/icm



Runtime architecture

- ▶Tool interoperability
- ▶Process orchestration
- ▶Explicitely modeled execution engine
- ▶Models@run.time principles
- ▶Protocol automata



This work has been partially carried out within the framework of the Flanders Make project MBSE4Mechatronics (grant nr. 130013) of the agency for Innovation by Science and Technology in Flanders (IWT-Vlaanderen).
Acknowledgements. Hans Vangheluwe, Joachim Denil, Kristof Berx, Klaas Gadeyne, Simon Van Mierlo.

Dávid, Denil, Gadeyne, Vangheluwe: **Engineering Process Transformation to Manage (In)consistency**, 1st International Workshop on Collaborative Modelling in MDE, 2016
Dávid, Syriani, Verbrugge, Buchs, Blouin, Cicchetti, Vanherpen: **Towards Inconsistency Tolerance by Quantification of Semantic Inconsistencies**, 1st International Workshop on Collaborative Modelling in MDE, 2016
Dávid, Denil, Gadeyne, Vangheluwe: **Towards Inconsistency Management by Process-Oriented Dependency Modeling**, 9th International Workshop on Multi-Paradigm Modeling, 2015