Engineering Practice and the Potential Role of Category Theory in Systems Engineering

Eswaran Subrahmanian (CMU/NIST)
Yves Keraron (Isadeus, France)

December 12, 2024
Topos Institute

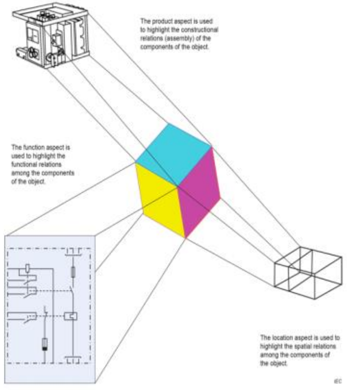
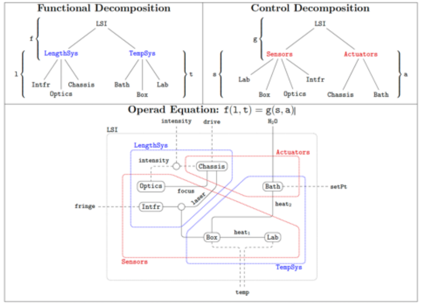
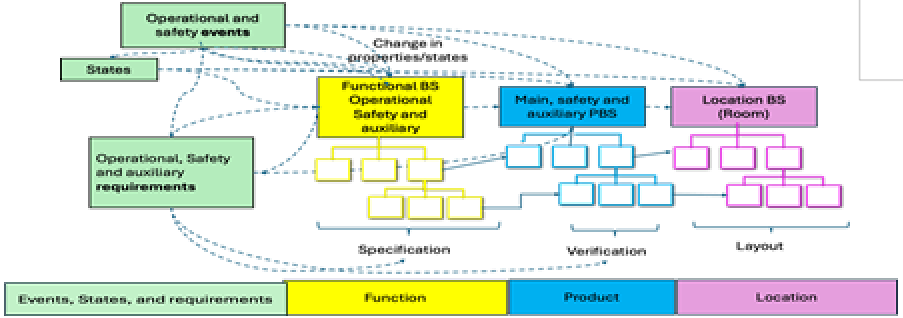


Figure 3 – Aspects of an object





Talk outline

1. Introduction
2. Systems engineering features and trends
3. Needs and obstacles to overcome
4. Role of Abstract Mathematics to Overcome Some Identified Obstacles
5. CT structures For SE
6. Conclusion

1 Engineering Practice

Systems engineering is a robust methodology to design complex systems. ISO 15288 describes the processes of this methodology. Books of knowledge have been edited to apply this methodology by describing in detail these processes in a paper-based engineering world.

According to INCOSE ***Model Based Systems Engineering*** ***is* the *formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases.***

To achieve this trend towards MBSE:

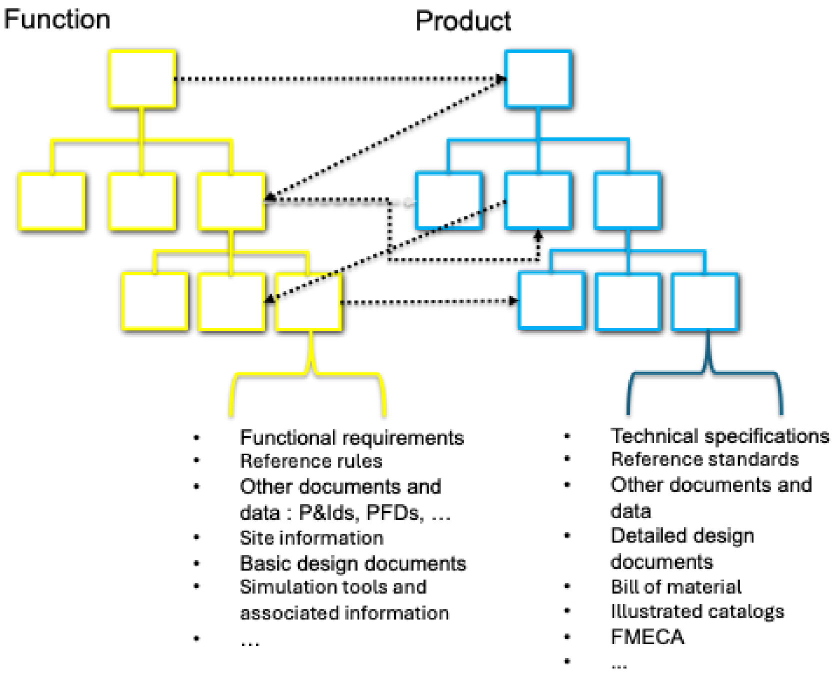
- It is first necessary to **clarify the basic concepts of systems engineering**,
- **Analyze industrial standards to support this approach** and draw **lessons from advanced industrial practice**s,
- **Focus on the structure and the relational feature of engineering,**
- **Identify the appropriate constructs to organize and control the system engineering information system** to efficiently **support the needs of the stakeholders and enable user interactions with the information syste**m, and during the operation phase with the real system itself to realize a digital twin.

Several authors have already identified the category theory as a foundation for Model Based Systems Engineering.

This presentation aims to develop some aspects of systems engineering and to make a first exploration where advanced mathematics may specifically **support Model Based Systems Engineering and data driven feedback** from later life cycle phases by addressing identified issues of the current industrial state of the art.

1 Introduction - Engineering Practice

Engineering practice in Systems engineering and information structures. The characterization of **zigzaging** between functional BS and Product BS has been theorized by Suh (Axiomatic design theory).



2 Current Engineering Practices and Trends Towards Model-based SE

Addressing the complexity of systems engineering.

- Standards
  - NASA
  - Power systems/ Oil and Natural gas) - ISO 81346, ISO 14224, …
- Use of PLM systems for requirements and design documents
- Digital mock-ups for components and subsystem modeling for design and analysis for verification

What is missing?

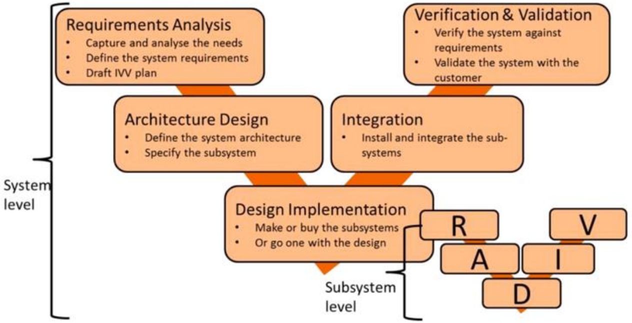
- Systems integration of these **fragmented components**
  - **Requirements - Documents**
  - **Function Structure modeling - Hierarchical Modeling and representation**
  - **Product modeling _ mapping for Function structure to Product structure**
  - **Location modeling - Mapping products to specific location**
  - **Interactive design process support**
  - **Feedback from later phases of the systems life-cycle**

2 Systems Engineering (SE) Features and Trends

- Model - based system and process descriptions
- Standards for specific systems engineering models (Process industries, for example)
- Use of Product Life cycle management systems - mainly documents
- 3-D geometry of physical parts and systems
- Verification of systems against requirements.

All of this is in service of the V model of SE

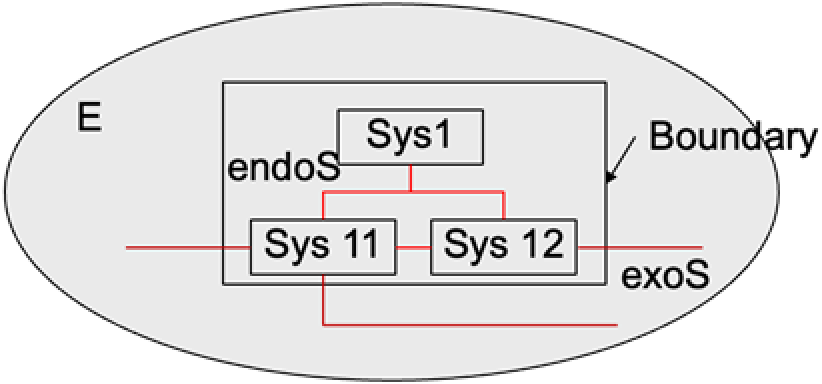
Recursive steps at each level of the system



√

2.1 System: A Definition

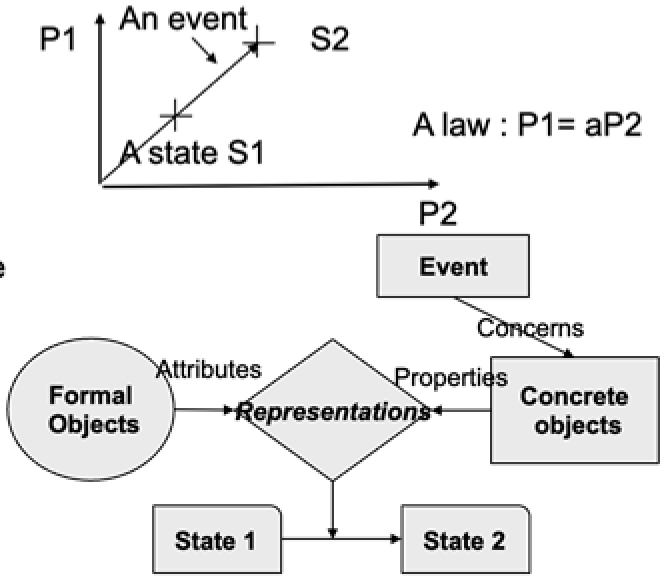
- What is a **system**?
  - ✓ Triple (C, E, S) : Cardinality, Environment and Structure.
  - ✓ A system is made up of interacting parts that interact with their environment.
- Properties of a system : :
  - ✓ Properties of Si different from properties of Sij due to relations between Sij
    - Notion of emergence
    - Concrete systems are diverse with specific contexts and laws
- There are reciprocal impacts of the environment on the system and of the system on its environment
- Engineers focus on some internal and external relations/properties of interest involving some internal and external objects
- Common framework and language for mastering the complexity of diversity



1. Bunge, M., Treatise on basic philosophy, Vol. 3, Ontology 1, The furniture of the world, D. Reidel Publishing Company Dordrecht-Holland/Boston-U.S.A, 1977
2. Bunge, M., Treatise on basic philosophy, Vol.4, Ontology 2, A world of systems, D. Reidel Publishing Company Dordrecht-Holland/Boston-U.S.A, 1979
3. Micouin, P., Models based systems engineering- fundamentals and methods, Hoboken, New Jersey: Wiley-ISTE, 2014

2.2 Some fundamental entities of a Systems Engineering ontology

- The **system** as defined in the previous slide
- Some other essential **properties** defined in a mathematical theory in science e.g. the theory of gravitation of Newton, the theory of heat of Carnot
- **Essential properties** are linked by a **law** e.g. the Newtonian mechanics linking force, mass and acceleration or Carnot's theory giving the maximum efficiency ratio of a power system linking this ratio property of a power system, with the temperature of the hot source, property of the fluid cooling a burner and the temperature of the cold source, property of the fluid before heating.
- A concrete object has physical properties
- A formal object has properties expressed in a mathematical way
- A **state** is a position in a space of properties of interest e.g. Gibbs Diagram
- An **event** is a change of state, of position in a space of properties
- A **requirement** is a constraint on a property in some realizable conditions, for a state of an object



3 Needs and Obstacles

Engineering is progressive; **the importance of architecture in SE**; **interface management; trade-offs; decision traceability**

Towards an **increased consistency of an evolving and more detailed system** until the system is in operation

Maintain the consistency of the information system during the later phases of the system life-cycle

Needs in engineering, procurement, construction, and test phases. Needs in downstream phases; examples of needs in design and in operation.

Main obstacles: volume of information (Tens of millions of data, every month update and mock-up exchanges between process and civil engineers; **different standards**; different tools; organization issues; **poor use of** business **expertise and knowledge** because of the **constraints of the digital tools**; **90 % of the time in rekeying information**!); less time for exchange between experts from different disciplines and **collective work**.

Lack of standards; lack of precision in the definitions jeopardizing the formalization or with a loose formalization based on poor definitions useless for practitioners; a **link between existing business standards and new technological standards**

3.1 Modeling

Modeling based on concepts, laws, and properties of interest.

Different computational codes, languages from different disciplines

How can formal and factual compliance with the requirements be checked?

How can we improve the models through the feedback of data in operation and maintenance with the digital twin technologies?

3.2 System Requirements

A requirement is an agreement between stakeholders translated into a formal language because it shall be verified according to an agreed method.

Define the requirement in terms of ***relations with other objects and processes***

Requirements include functional, structural (physical), safety, operations, and failure modes for system behavior as specifications and constraints.

Check consistency and completeness of requirements - Limit their number as necessary and sufficient as it is a cost driver

Verification of requirements by use of modeling (check formally and factually)

Goguen, J., 1999. Tossing Algebraic Flowers down the Great Divide. *People and ideas in theoretical computer science*, pp.93-129.)
Bhatt, D. et al. The CLEAR Way To Transparent Formal Methods Honeywell aerospace, July 2018

3.3 An example of a formal requirement

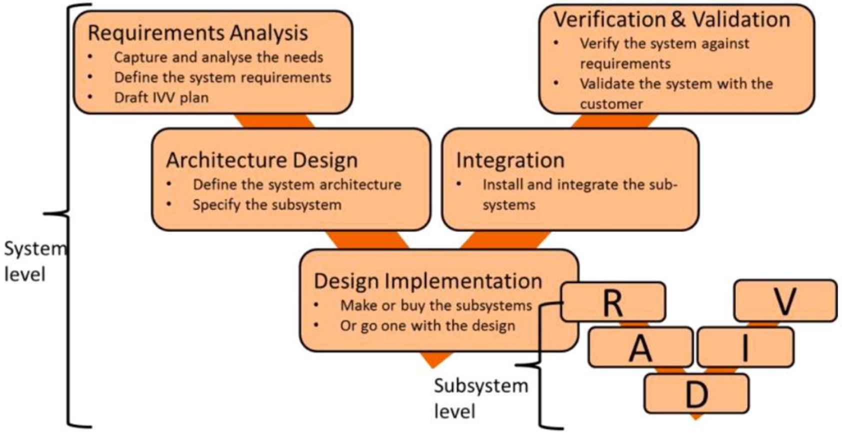
In normal operation condition, the flowrate of the pump shall be superior to 50m3/h

$$[\text{When } C] \longrightarrow val\,(O.P) \in D \subset Im\,(P)$$

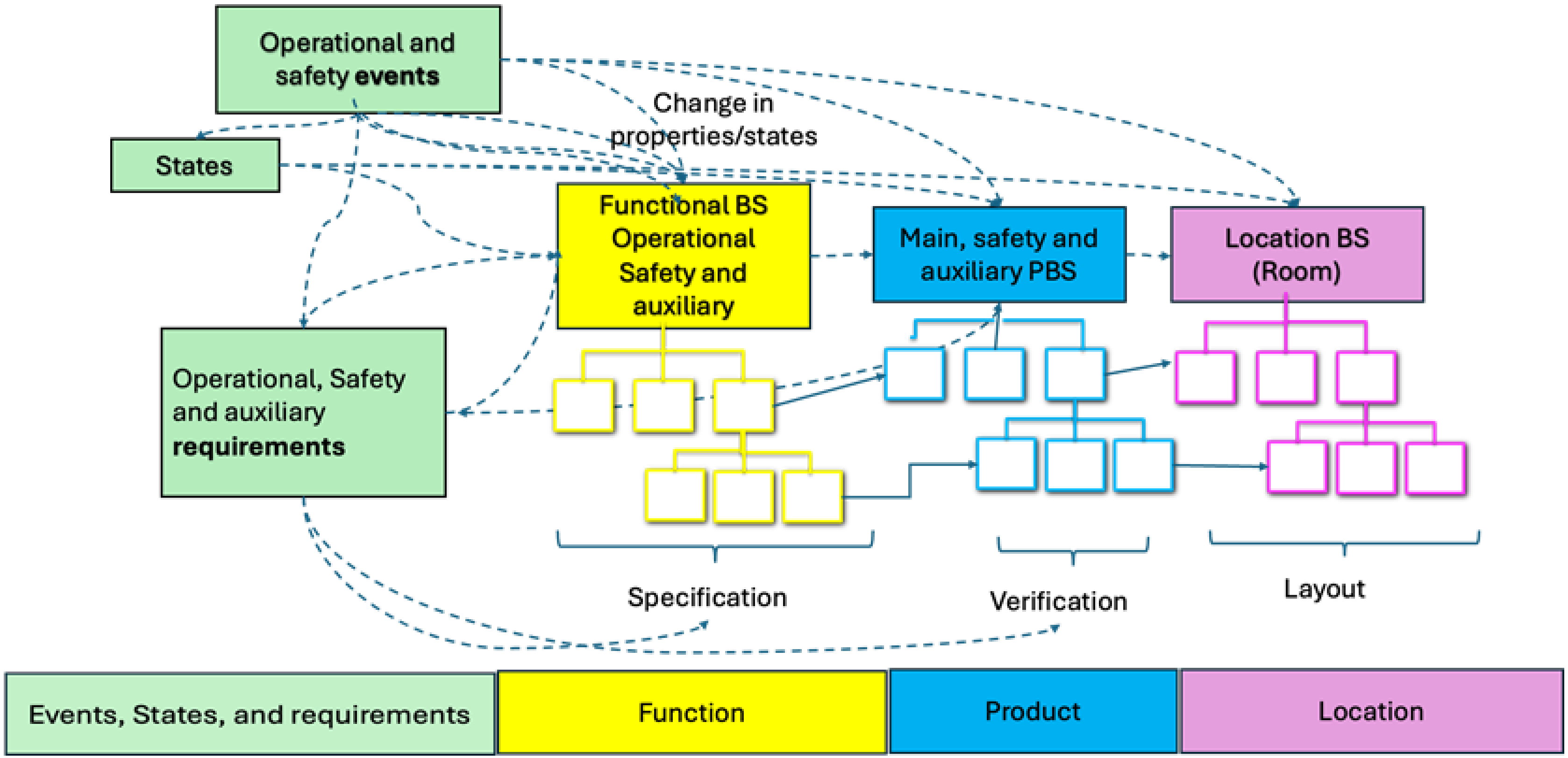
Micouin, P., Models based systems engineering- fundamentals and methods, Hoboken, New Jersey: Wiley-ISTE, 2014
Micouin, P., Toward a property-based requirements theory: System requirements structured as a semilattice. Systems Engineering 11(3):235 - 245, June 2008

3.4 Design Verification and Validation

- Use of simulation tools to verify and validate the design as early as possible in the life-cycle of the system
- Test operational scenarios to tune the requirements and verify the design
- Need to incorporate regulations and requirements on information related to equipment important for safety (Traceability and archiving)
- From regulation requirements to a compliant design.



3.5 Overview: Requirements for Information System



3.6 Relationships: Functional, Product and Location Structure

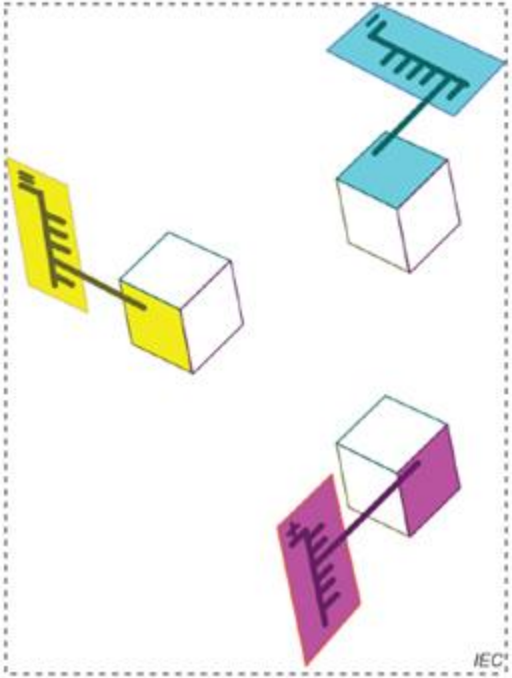


Figure C.1 – Three independently defined objects

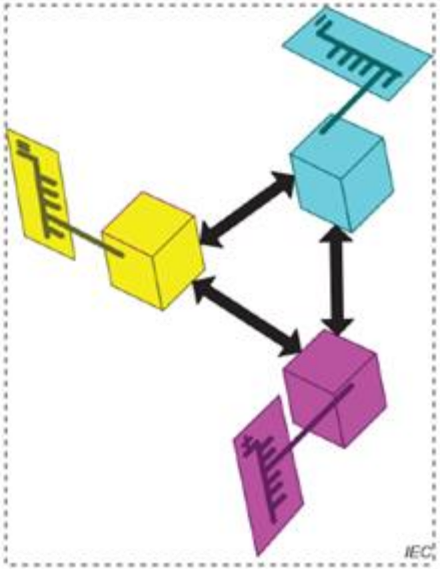


Figure C.2 – Three separate objects with mutual relations

Ref: ISO/IEC 81346-1

3.7 Integrated Functional, Product, and Location Breakdown Structures

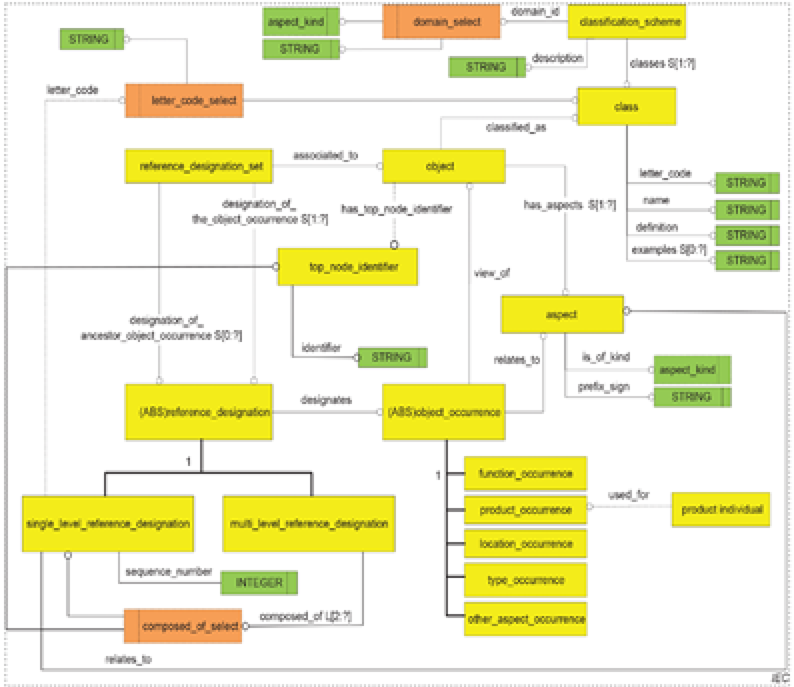


Figure A.1 – EXPRESS-G model of the reference designation system

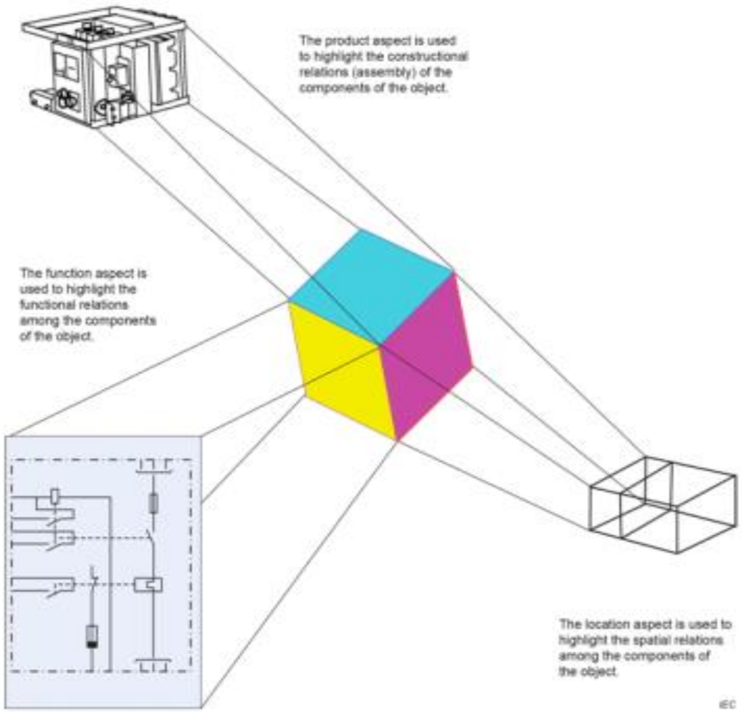


Figure 3 – Aspects of an object

Ref: ISO/IEC 81346-1

3.8 Function, Product and Location- Power Supply

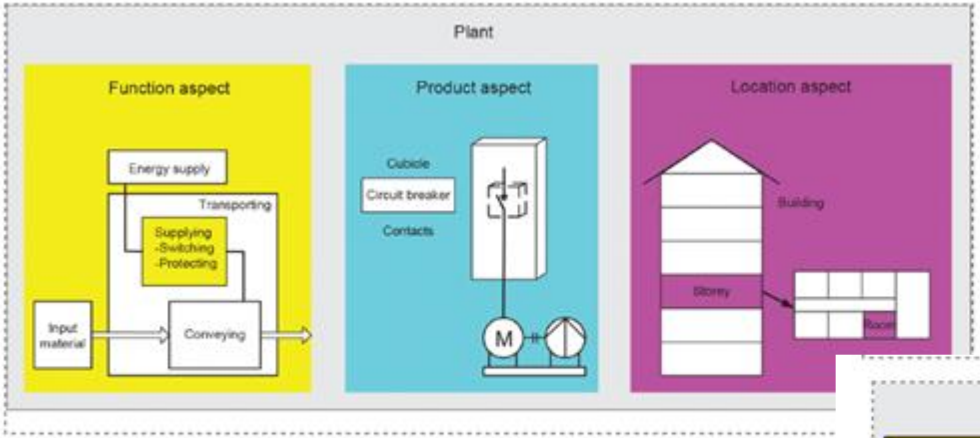


Figure C.4 – Overview of the process system

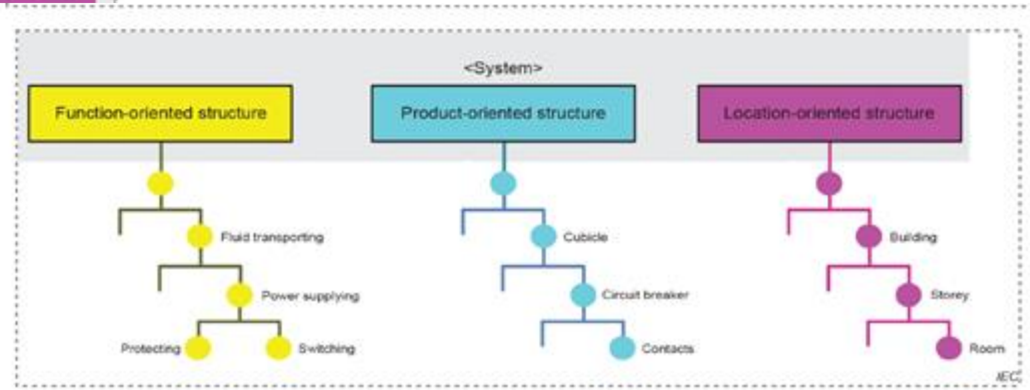


Figure C.5 – Tree-like structures of the system

Ref: ISO/IEC 81346-1

3.9 Reference Designations: Coding Scheme

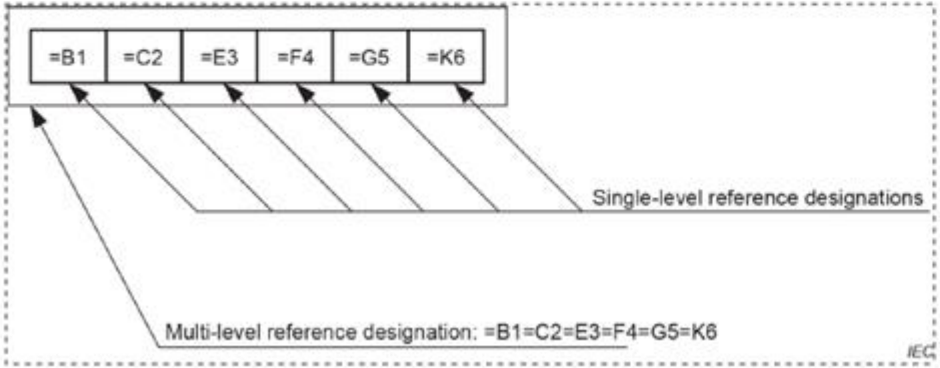


Table 3 – Examples of multi-level reference designations with multiple prefix signs

Multi-level reference designation of an object based on a function-oriented structure	Multi-level reference designation of an object based on a product-oriented structure	Multi-level reference designation of an object based on a location-oriented structure
==C==B==W	--C1--B2--E3	++B1++2++G2++M1++P2

3.10 Structures with Sub-Object Definition

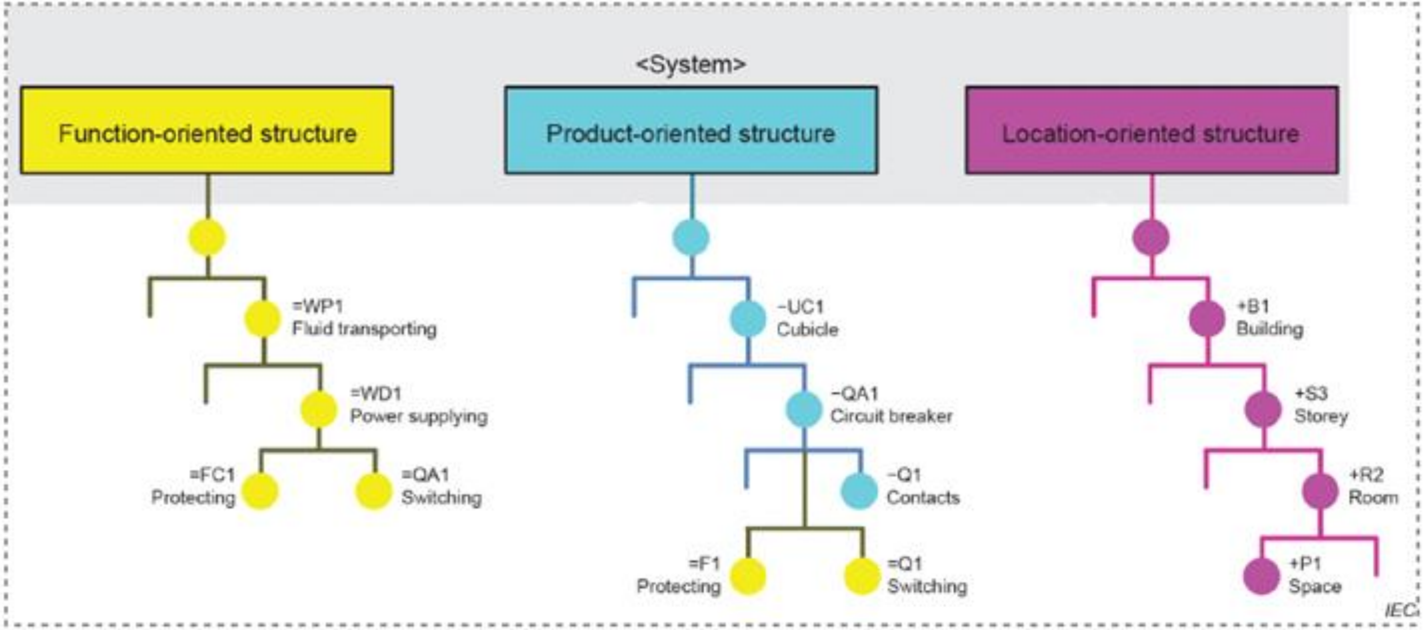


Figure C.7 – Structures with designated sub-objects

Component system coding are hierarchically classified according to ISO/IEC81346-1

3.11 Evolution of Function and Product Component - Motor

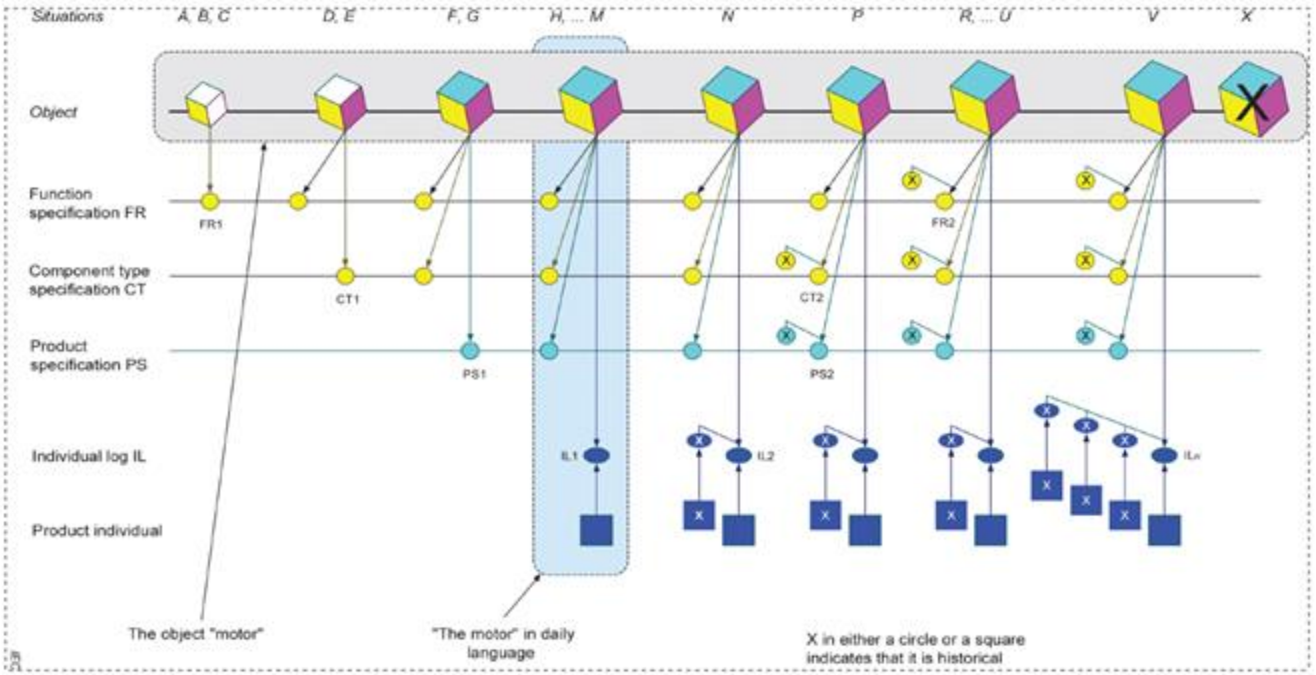


Figure B.2 – The object's life cycle

Ref: ISO/IEC 81346-1

3.12 Relationships: FBS, PBS, and LBS-Power Supply

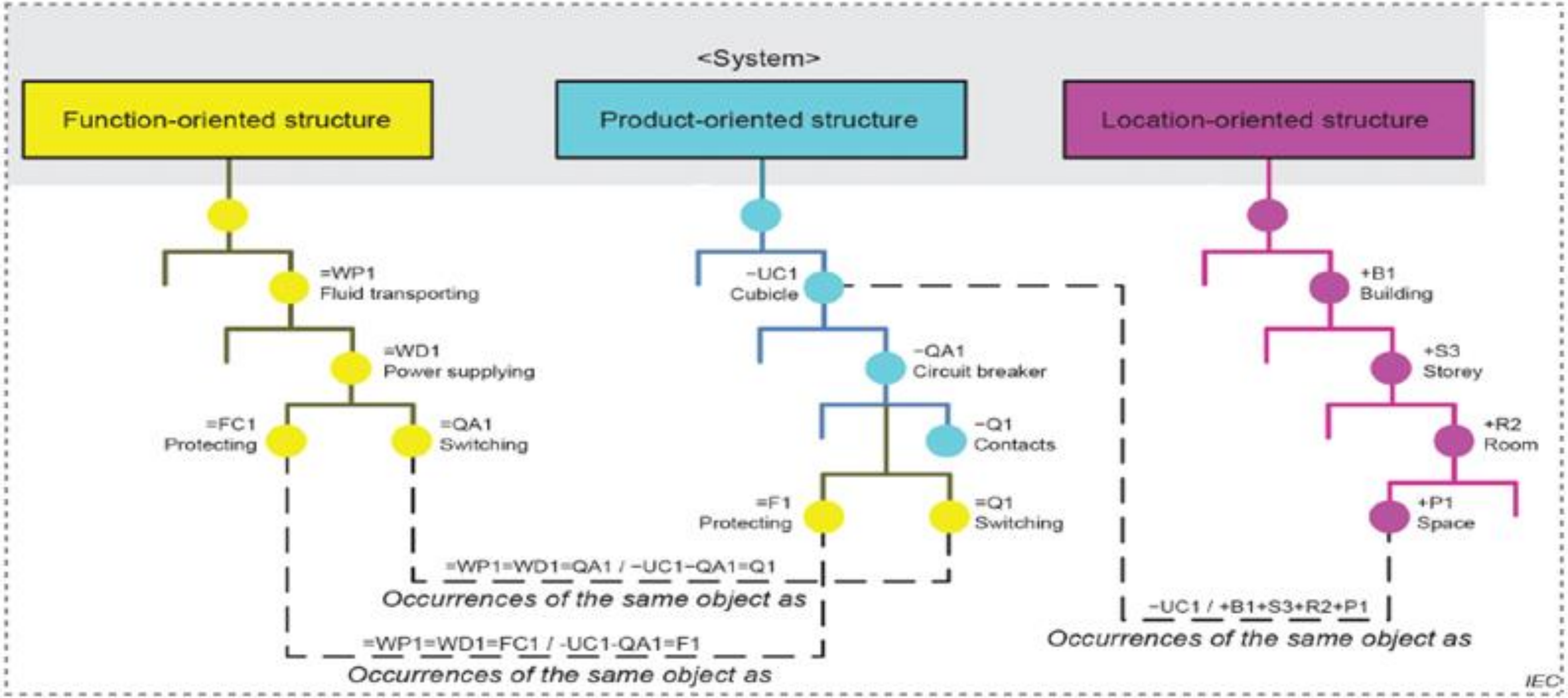


Figure C.9 – Relations expressed by reference designation sets in which both designations are unambiguous

Ref: ISO/IEC 81346-1

3.12 Relationships: FBS, PBS, and LBS-Power Supply

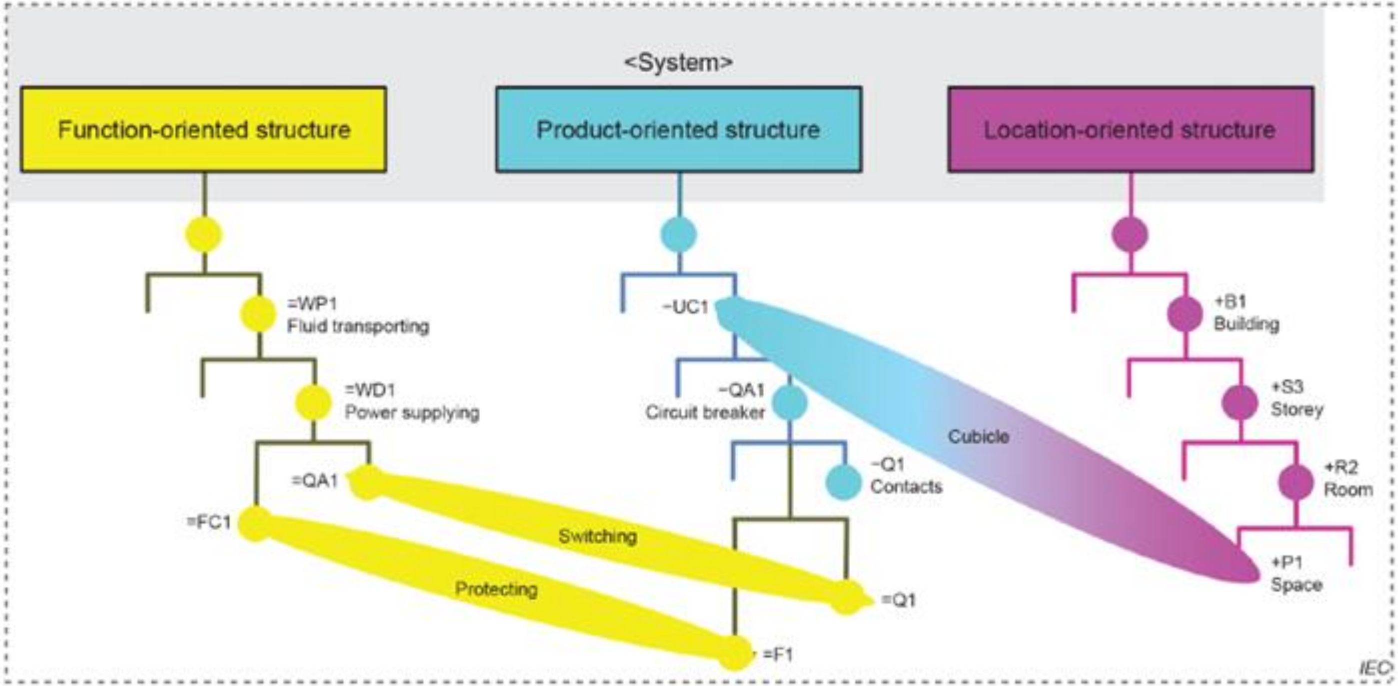
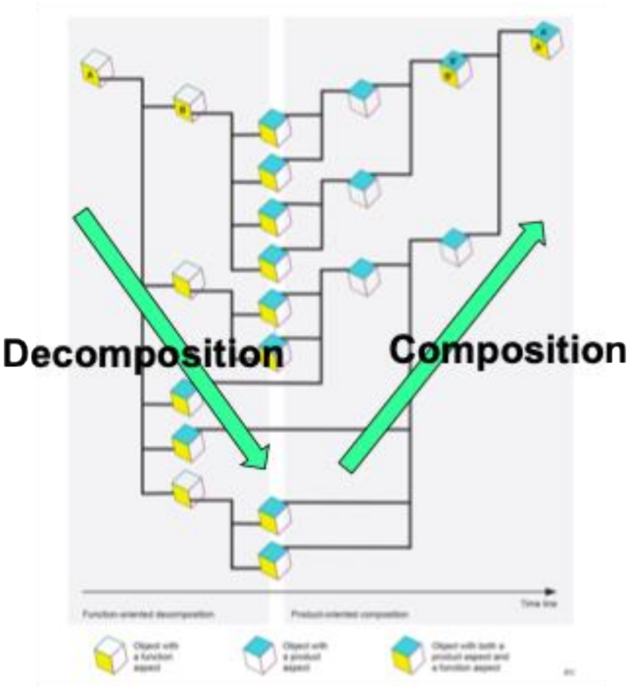


Figure C.8 – Structures with some merged-and-shared objects

Ref: ISO/IEC 81346-1

3.13 Decomposition and Composition: FBS and PBS



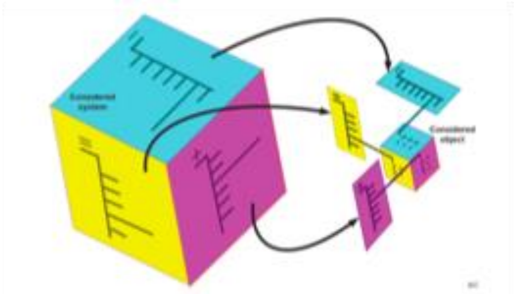


Figure 20 – Illustration of an object accessible from three aspects, and where these aspects are used also for internal structuring

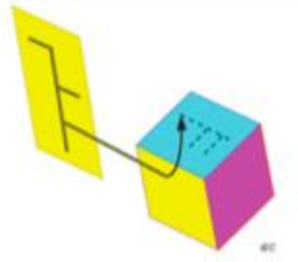
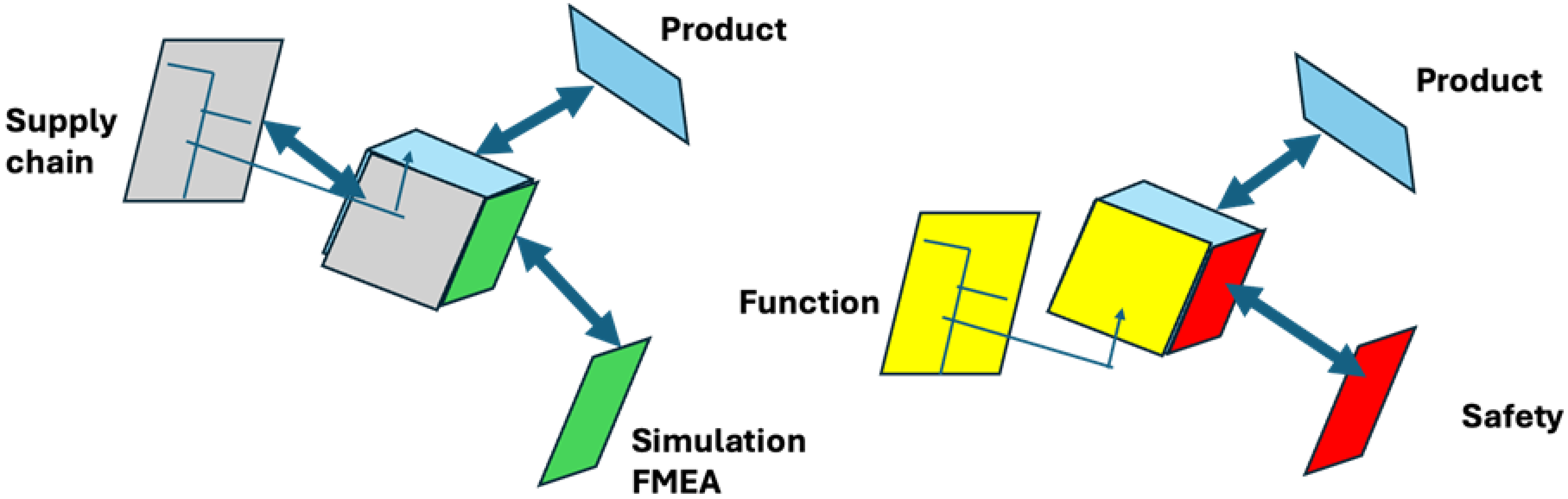
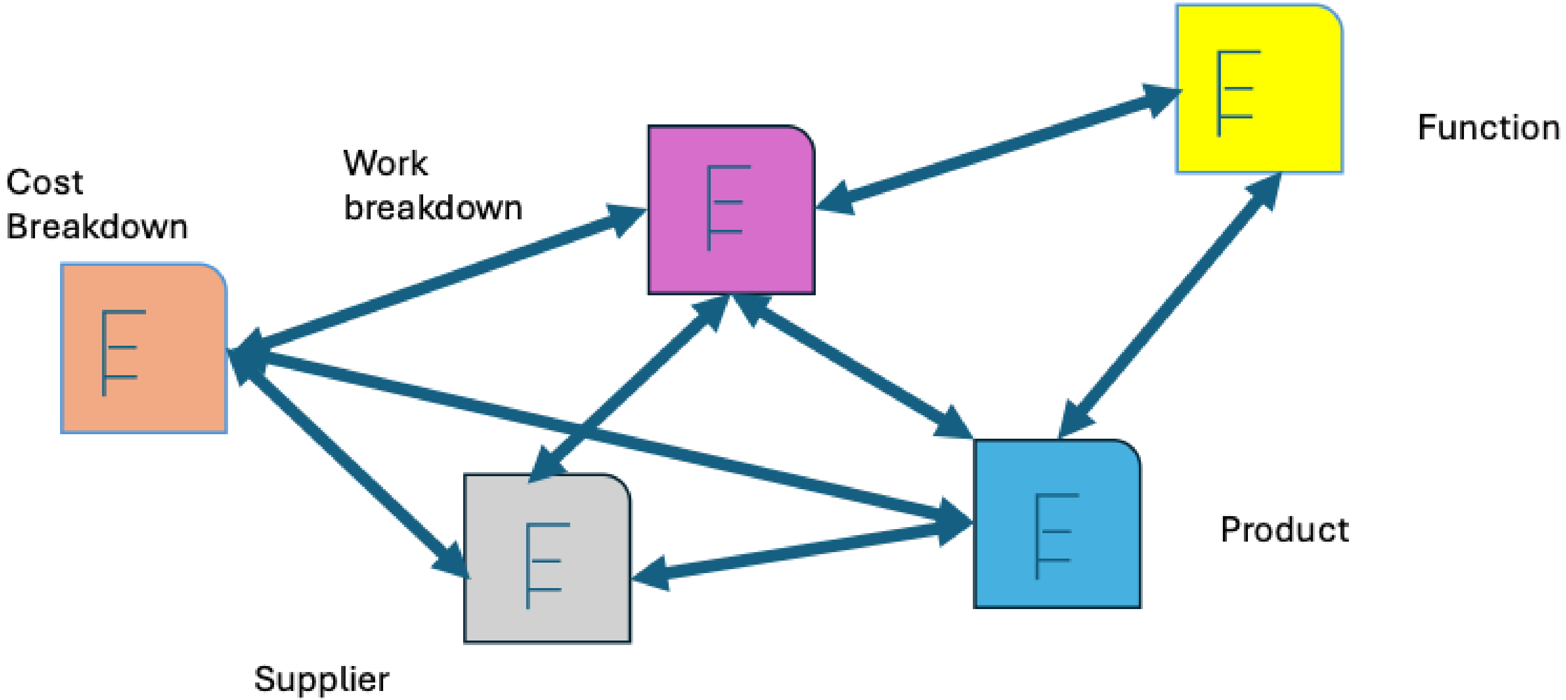


Figure 21 – Illustration of an object identified by means of one aspect and with sub-objects identified by means of another aspect

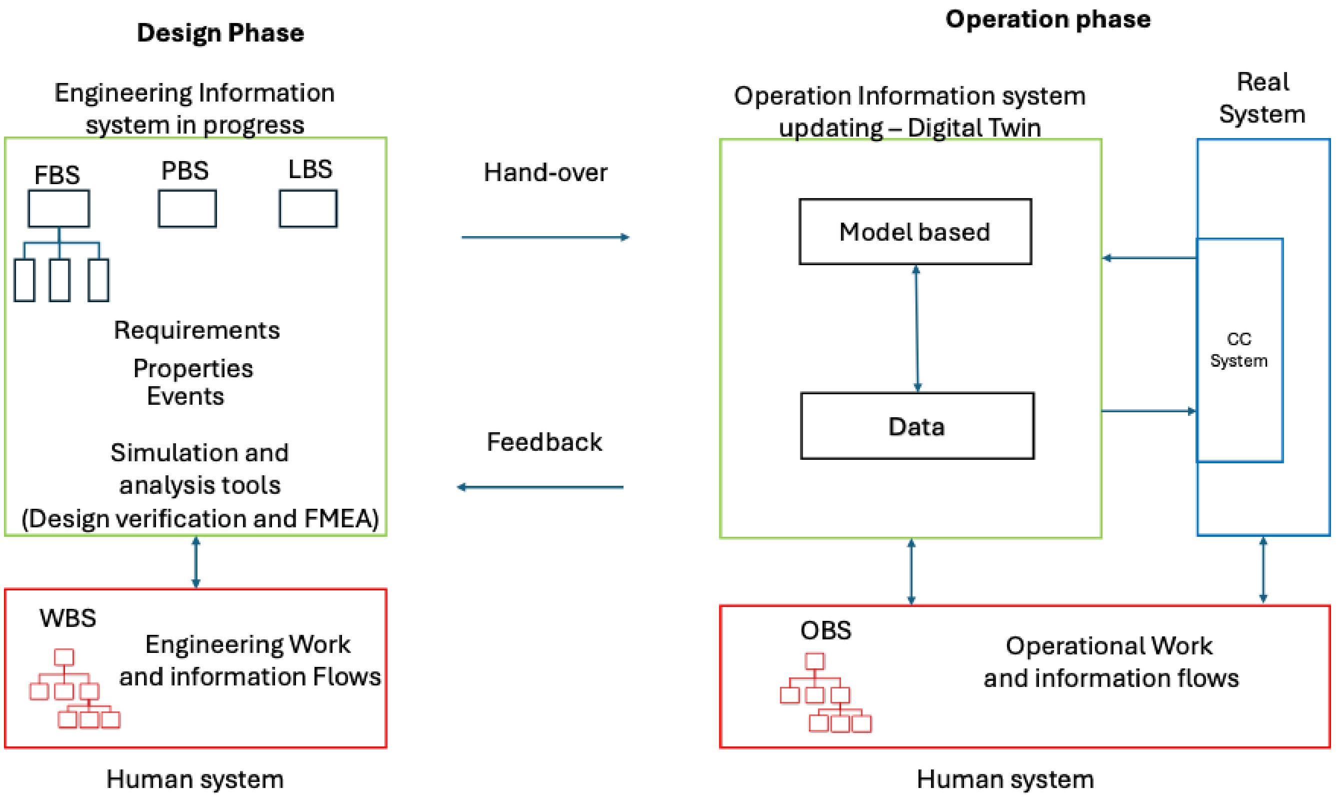
3.14 Relationship to other Breakdown Structures



3.15 Relationships: FBS, PBS, WBS, CBS & Supplier Breakdown Structures



3.16 Relationship: Design/EPC → Operational Phase with the Digital Twin



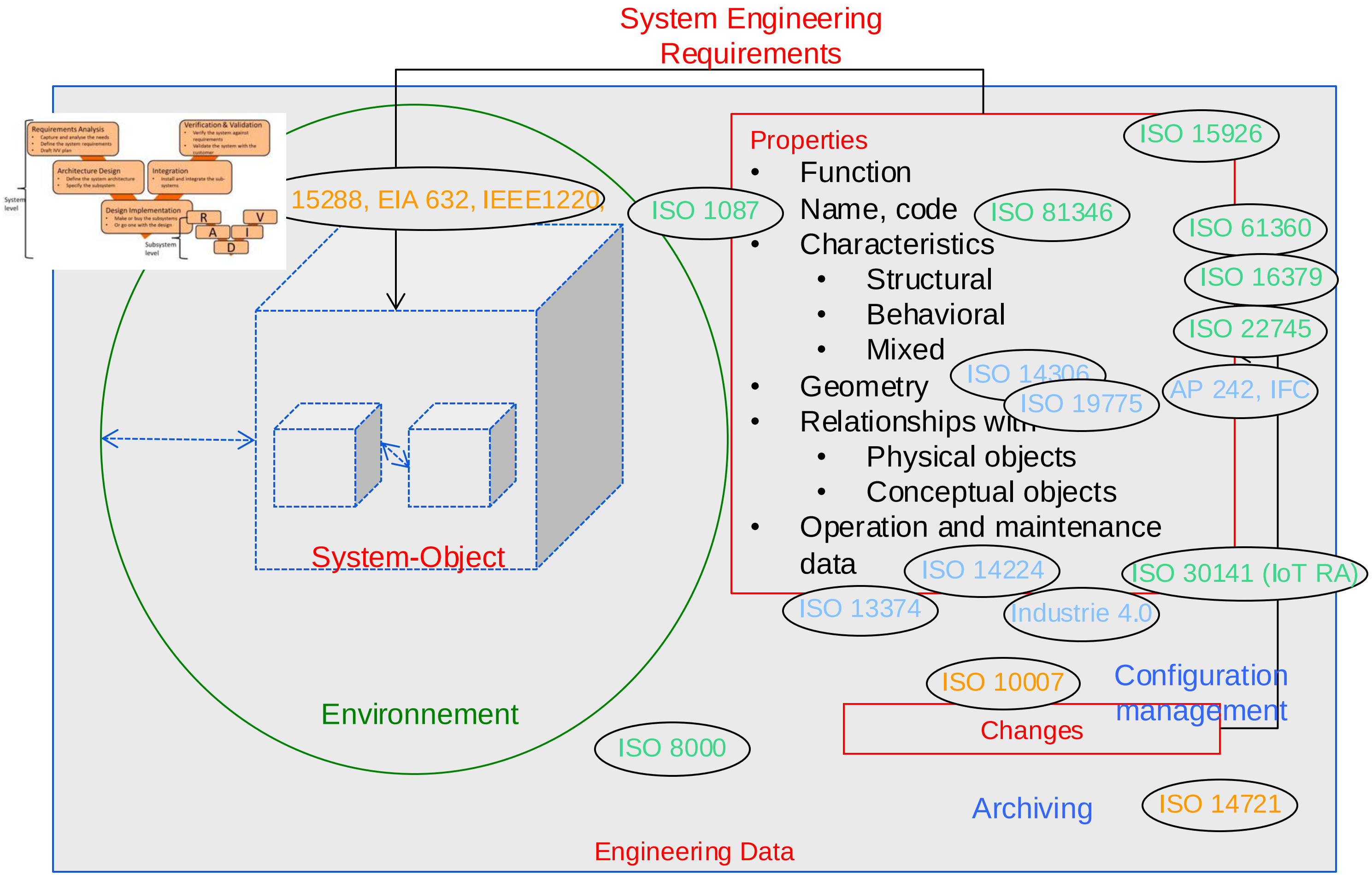
3.15 Sample of Needs - Real-Life Experience on an NPP

- Analysis of the consistency between the classification of electrical consumers and their switchboards, (FBS-PBS)
- Loss of electrical sources: For the loss of an electrical supply panel, provide the list of consumers lost and safety functions affected (PBS-SFBS)
- If a room is lost, a list of lost components and impacted safety functions is produced (LBS-PBS-SFBS)
- If a DC cabinet is lost, what components are lost and what safety functions are affected? (LBS-PBS-SFBS)

These tasks are supported through access to information by browsing the information organization and control structures

An additional need is to enable the interaction of the users with the information system using the interaction structures for updating the information system

3.16 Engineering Data of an Asset and some Standards of Interest



4 Role of Abstract Mathematics to Overcome a few Obstacles

Current approaches: SysML, OPM, AADL , Mathworks languages, Modelica, VHDL-AMS, BPML, Petri-nets, and other analysis models (differential equations, Finite machines, Bayesian nets for diagnosis).

All of these modeling approaches are **fragmented**.

Need for a relational approach

The goal is **to explore the application of category theory** to address the **composition of a plurality of disciplines, modeling languages and models.**

4.1 State of MBSE (Excerpts)

Recent studies have shown that engineers continue to spend inordinate amounts of time searching for information and assembling reports. This trend has only grown with increasing scale and complexity of systems, resulting in dramatic increase in system requirements. Thus, managing requirements using simplistic methods as such as *checklists or ad hoc methods such as disconnected databases no longer suffices.*

The value of MBSE stems from the fact that all system-related information is stored and configuration managed in a central repository[1]. This characteristic enables the interconnection of model elements, effective information retrieval, and reasoning about the system. This interconnectivity also enables automatic propagation of design changes, consistency checking, and error identification. Collectively, these characteristics are the key discriminators of MBSE.

1 Comment: It should be a distributed repository

Madni, A.M. and Sievers, M., 2018. Model-based systems engineering: Motivation, current status, and research opportunities. *Systems Engineering*, *21*(3), pp.172-190.

4.2 Hierarchical Decomposition of a System-Operads

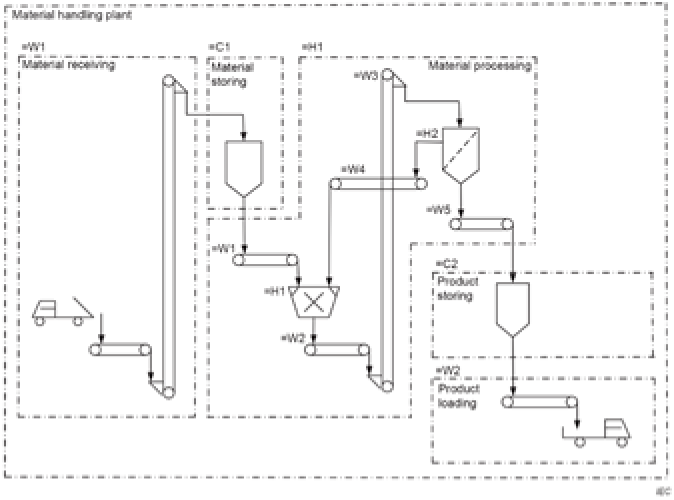
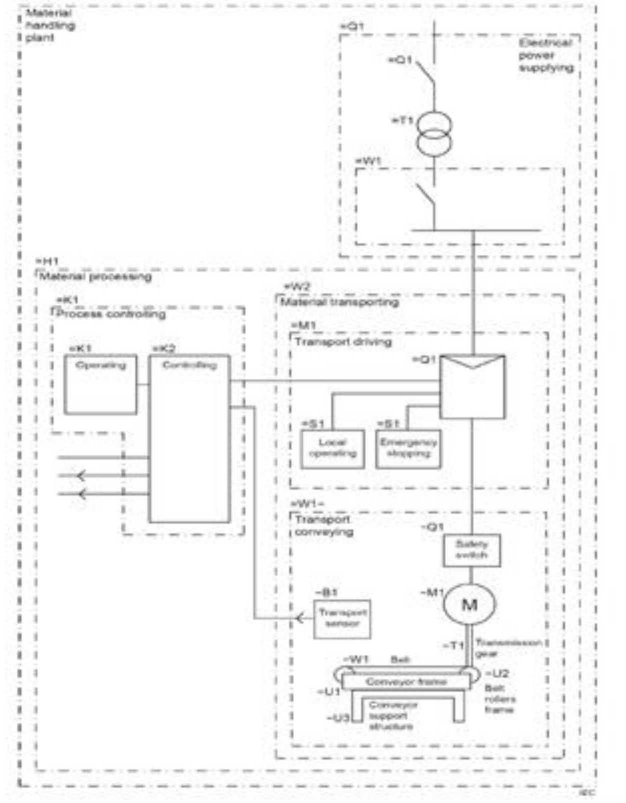
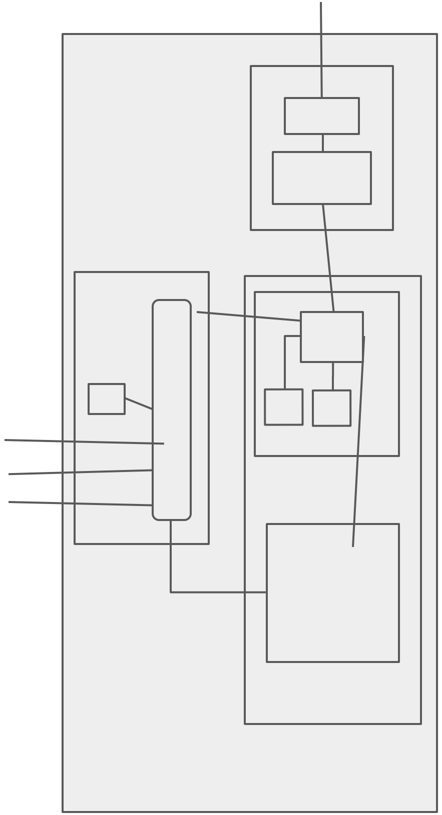


Figure H.1 – Process flow diagram for a material handling plant





4.3 **Adjunctions across Systems Structures**

We can think of it forward mappings (right) and inverse (left) mappings as adjunction between any two information structures determined recursively.

Recursion on Function structure to product structure to behavior as designed.

Inverse mapping from behavior to structure from observations (verification).

Data driven approach to determine inverse mappings - Left adjunctions

5. CT structures For SE

Hierarchical decomposition - Operads for design and diagnosis

Use of analysis tools: Operads for Design analysis and synthesis

Workflow breakdown and flow - String-diagrams

Delta lenses to incorporating change. (managing change and adjunctions)

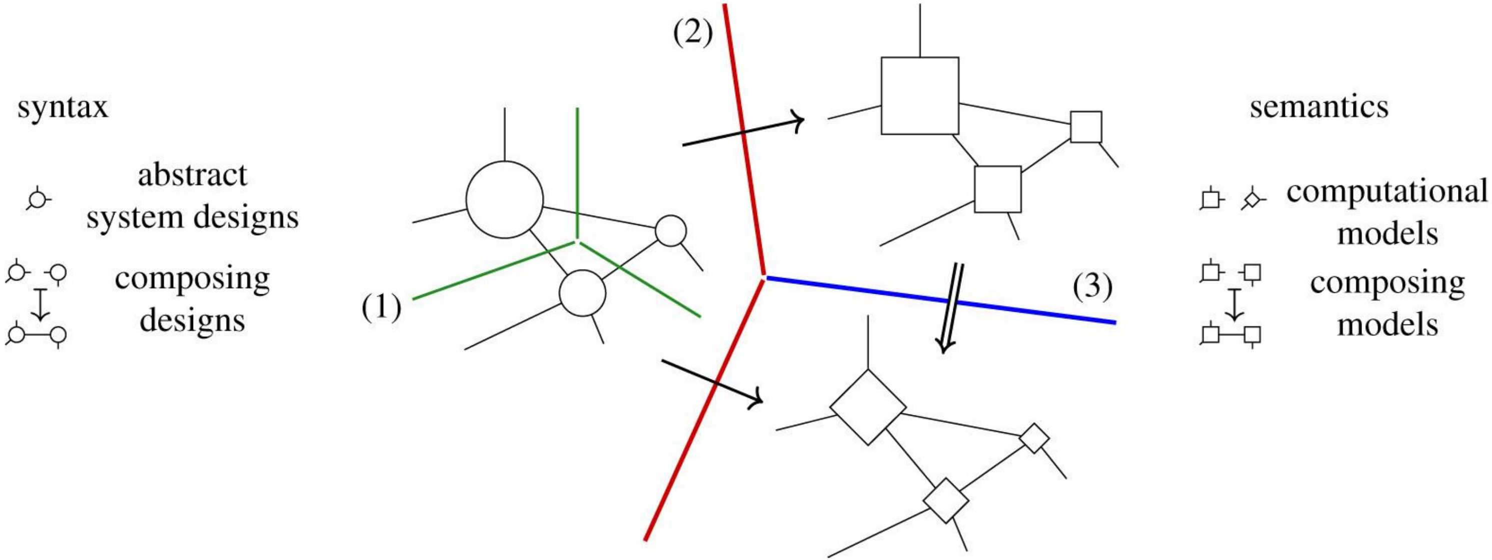
Modeling analysis tools (dynamic system models and others)

Model evolution

Maps between Functional, Product, Location and other hierarchies (Data Integration and Functorial Data Migration)

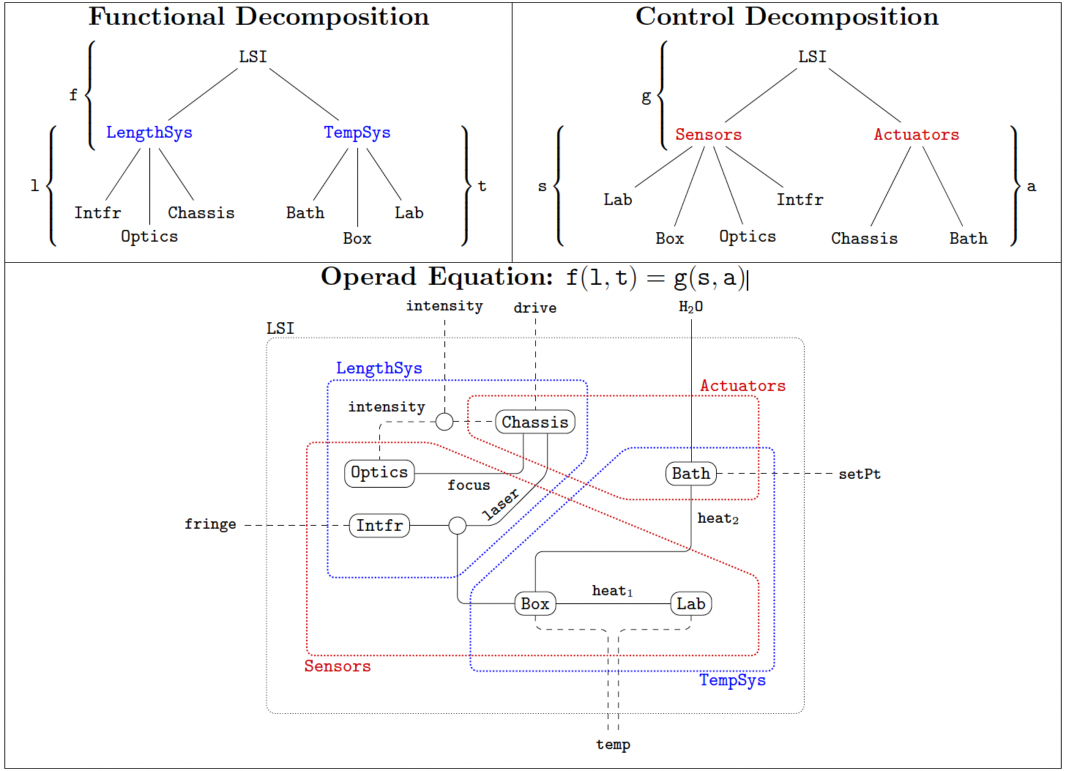
CT (Topos/Sheaf theory) may help manage variations from the common core of a product.

5.1 Operads for Specification, Analysis and Synthesis



Foley, J.D., Breiner, S., Subrahmanian, E. and Dusel, J.M., 2021. Operads for complex system design specification, analysis and synthesis. *Proceedings of the Royal Society A*, *477*(2250), p.20210099.

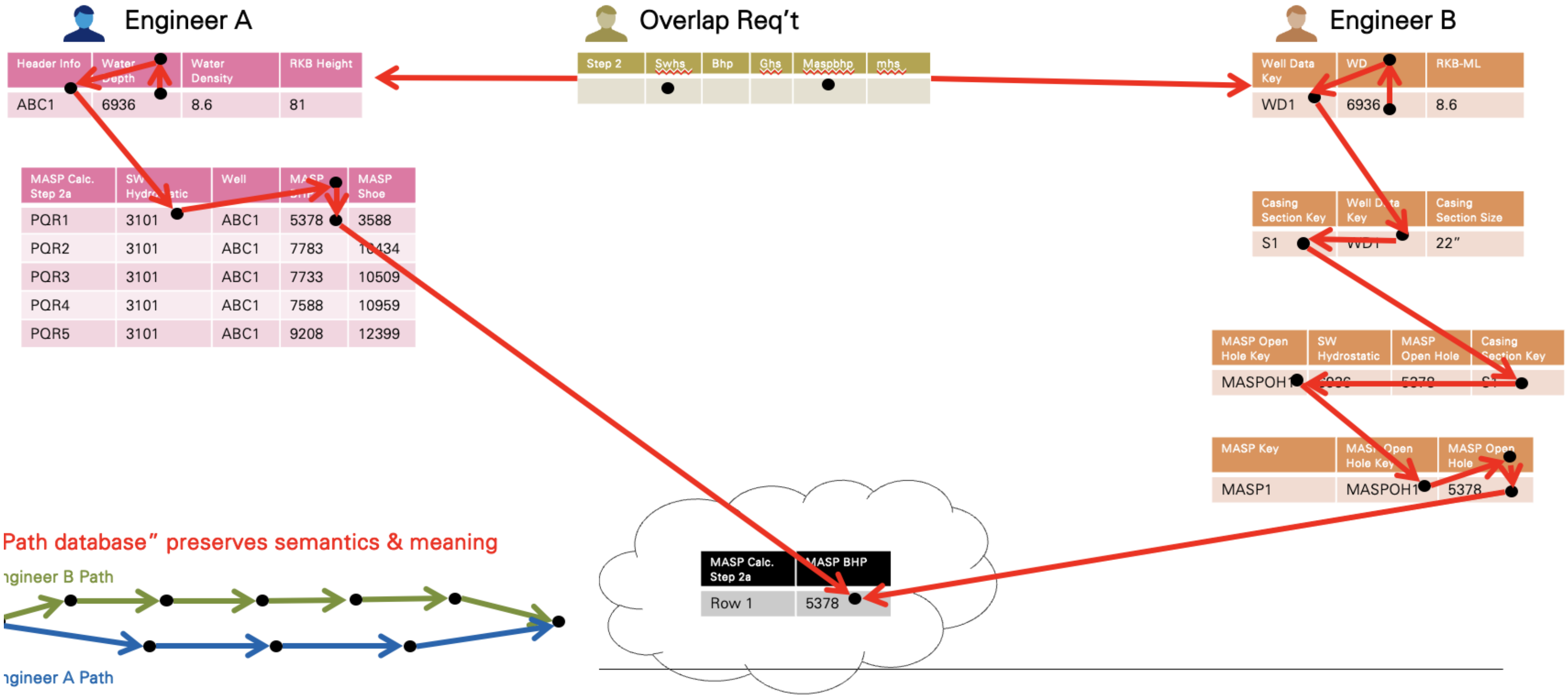
5.2 Operadic Decomposition : Multiple Views



Functional view and control view of the LSI system

Breiner, S., Pollard, B., Subrahmanian, E. and Marie-Rose, O., 2020. Modeling hierarchical system with operads. *arXiv preprint arXiv:2009.09848*.

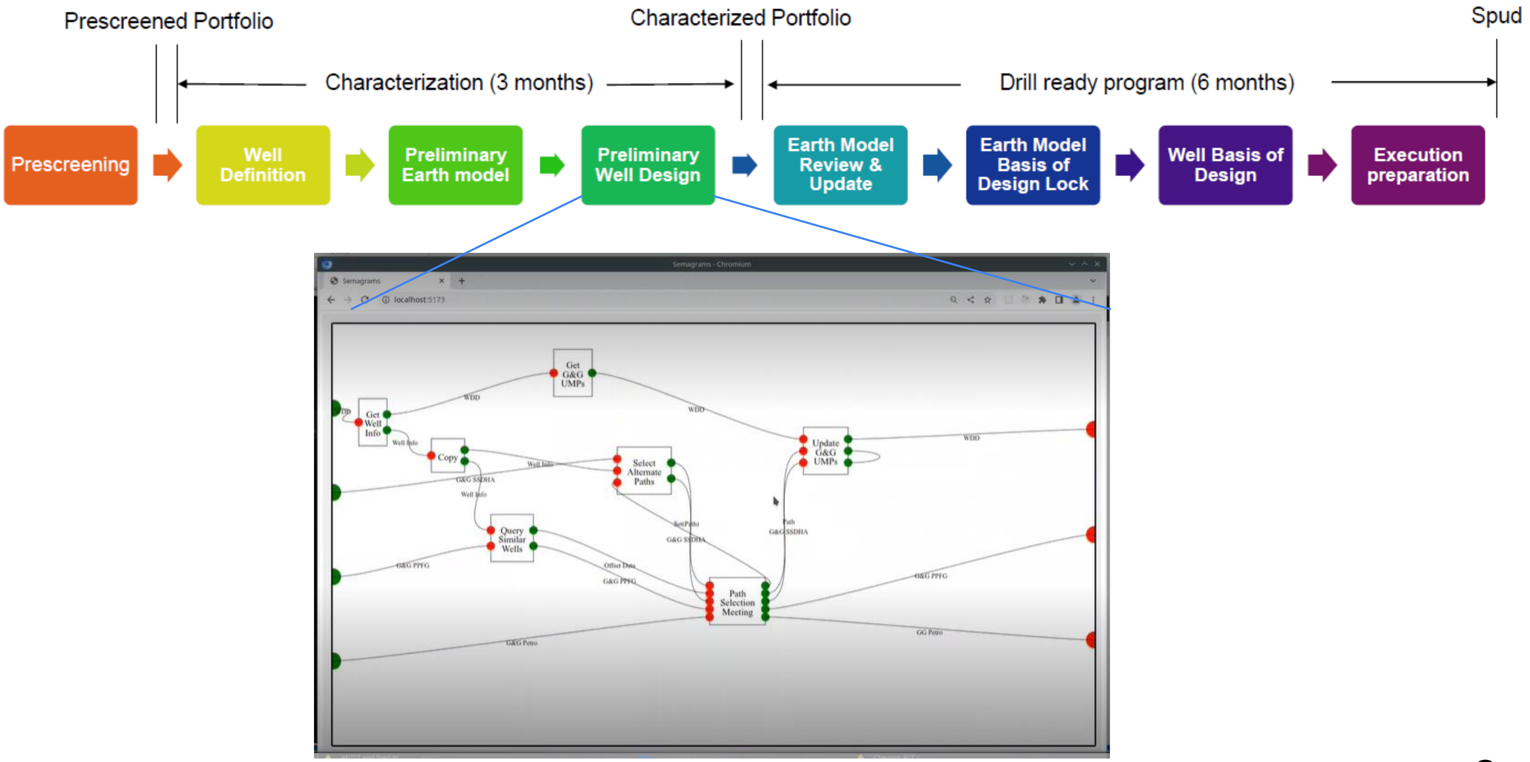
5.3 Categorical Data Integration - Chevron Example



Baylor, B., Daimler, E., Hansen, J., Montero, E. and Wisnesky, R., 2022. Consensus-Free Spreadsheet Integration. *arXiv preprint arXiv:2209.14457*.

Courtesy of Brandon Baylor

5.4 Process as String diagrams – Chevron example



Baylor, B, Filonik, D.. Carlson, K., Breiner S., e. Montero, Patterson, E., Subrahmanian, E. , Category theory for PLM, Unpublished Chevron Report, May 2024

Courtesy of Brandon Baylor

5.5 Delta Lenses for Model Updates From Data – Manufacturing Assembly Line

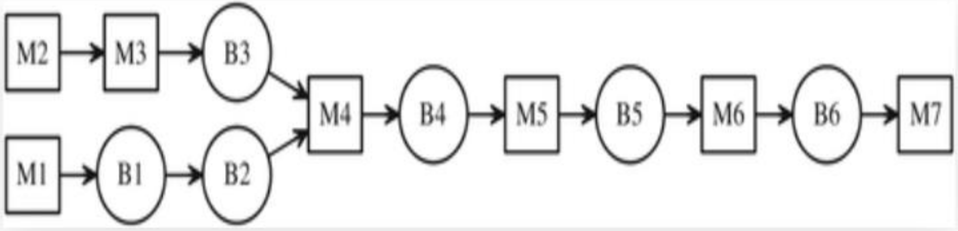
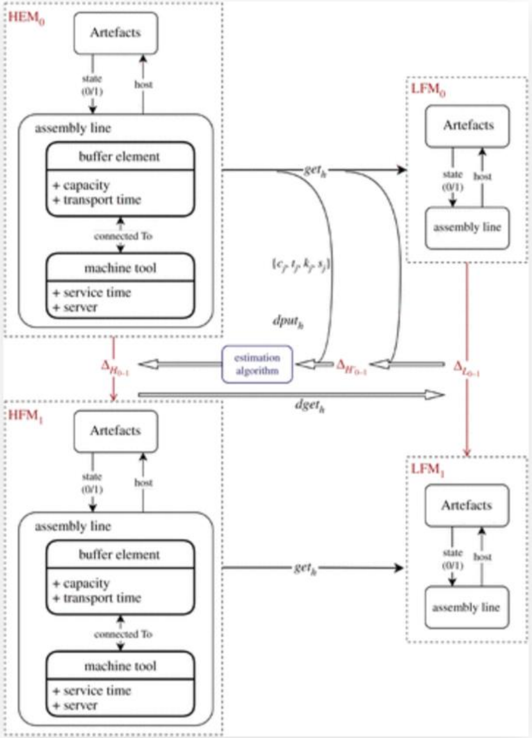
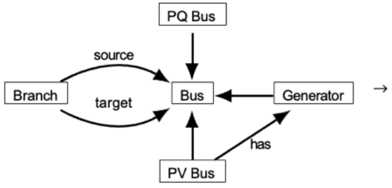


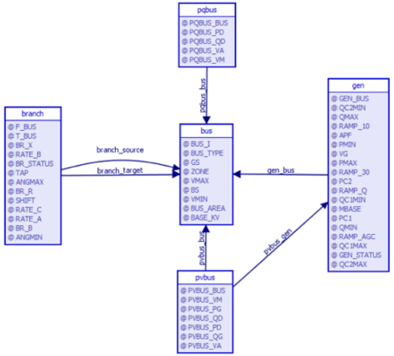
Figure 6. Representation of the assembly line. The squares denote machines and the circles as buffers.

Qi, Q., Terkaj, W., Urgo, M., Jiang, X. and Scott, P.J., 2022. A mathematical foundation to support bidirectional mappings between digital models: an application of multi-scale modelling in manufacturing. *Proceedings of the Royal Society A*, *478*(2264), p.20220156.



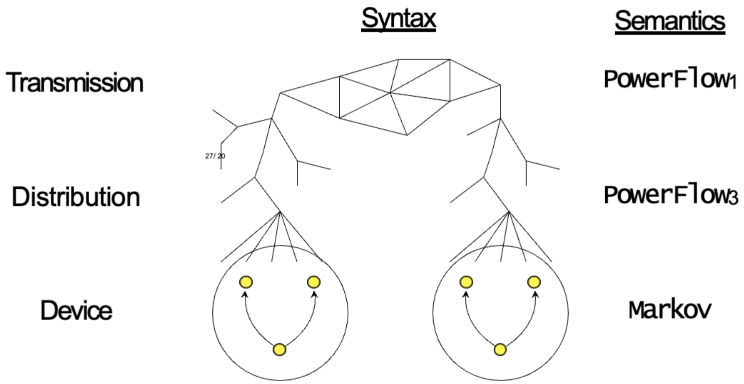
5.4 Modeling Power Flow - Analytical Tools





$$P_i = \sum_j V_i V_j \left(g_{ij}\cos(\theta_{ij}) + b_{ij}\sin(\theta_{ij})\right)$$

$$Q_i = \sum_j V_i V_j \left(g_{ij}\sin(\theta_{ij}) - b_{ij}\cos(\theta_{ij})\right)$$



Nolan, J.S., Pollard, B.S., Breiner, S., Anand, D. and Subrahmanian, E., 2020. Compositional models for power systems. *arXiv preprint arXiv:2009.06833*.

6 Conclusion: Challenges for CT

Systems engineering is a pluralistic set of information structures and of their relationships that need to be maintained and updated.

We need to

- **formalize the requirements** for Systems engineering.
- create **relational maps from hierarchies** of function to product to location and other decompositions.
- create **relational maps between hierarchies of classification of equipment** (catalogs, …)
- **identify specific CT formalisms** for the different modeling formalisms.
- **formalize change and consistency management**.
- **formalize work-flows and their connection the data space** for each breakdown structure.

Engineering Standards recommend methods for breakdown structures and classification structures. This should enable mapping(morphisms) between these structures built according to the standards

While **contextualizing these structures for specific contexts of the suppliers**, the suppiers should append using morphisms between these specific structures.

References

1. ISO/IEC/IEEE 15288:2023, Systems and software engineering - System life cycle processes
2. Suh N. P., Axiomatic Design: Advances and Applications, Oxford University Press, 2001
3.]Bunge, M., Treatise on basic philosophy, Vol. 3, Ontology 1, The furniture of the world, D. Reidel Publishing Company Dordrecht-Holland/Boston-U.S.A, 1977
4. Bunge, M., Treatise on basic philosophy, Vol.4, Ontology 2, A world of systems, D. Reidel Publishing Company Dordrecht-Holland/Boston-U.S.A, 1979
5. Micouin, P., Models based systems engineering- fundamentals and methods, Hoboken, New Jersey: Wiley-ISTE, 2014
6. Micouin, P., Toward a property-based requirements theory: System requirements structured as a semilattice. Systems Engineering 11(3):235 - 245, June 2008
7. SO/IEC 81346-1:2022-Industrial systems, installations and equipment and industrial products - Structuring principles and reference designations - Part 1: Basic rules
8. SO/IEC 81346-2:2019-Industrial systems, installations and equipment and industrial products - Structuring principles and reference designations - Part 2: Classification of objects and codes for classes
9. ISO/FDIS 81346-10:2022-Industrial systems, installations and equipment and industrial products - Structuring principles and reference designations - Part 10: Power systems
10. ISO/IEC 81346-12:2018-Industrial systems, installations and equipment and industrial products - Structuring principles and reference designations - Part 12: Construction works and building services
11. IEC 81355: 2024, Industrial systems, installations and equipment and industrial products - Classification and designation of information - Part 1: Basic rules and classification of information
12. SysML V2.0 OMG System Modeling Language, April 2024
13. Luzeaux Dominique. A formal foundation of systems engineering. In Complex Systems Design & Management, pages 133-148. CESAMES, Springer, 2014.
14. David Spivak. Category Theory for the Sciences. The MIT Press, 2014
15. Fong, B. and Spivak, D.I., 2018. Seven sketches in compositionality: An invitation to applied category theory. *arXiv preprint arXiv:1803.05316*.
16. Watson, M,, *Future of Managing Complexity*, NASA, May 2022
17. Bhatt, D. et al. The CLEAR Way To Transparent Formal Methods Honeywell aerospace, July 2018
18. Goguen, J., 1999. Tossing algebraic flowers down the great divide. *People and ideas in theoretical computer science*, pp.93-129.
19. Breiner, S., Sriram, R.D. and Subrahmanian, E., 2019. Compositional models for complex systems. In *Artificial Intelligence for the Internet of Everything* (pp. 241-270). Academic Press.

Thank you !

sub@cmu.edu

yves.keraron@isadeus.fr

We invite you to the journey for exploration in formalizing System Engineering practice to design an information support system

Acknowledgment- People the journey so far

NIST

Spencer Breiner
Blake Pollard (Ex-NIST)
Ram Sriram

Chevron

Esteban Montero
James Hansen
Brandon Baylor

Carnegie Mellon University

Daniel Filonik
Eswaran Subrahmanian

Topos Institute (Algebriac Julia)

Kevin Arlin
Evan Patterson
Owen Lynch
David Spivak

Conexus

Ryan Wisnesky
Eric Daimler