Mathematical Foundations of Transformation of Knowledge into Models and between Models

Presentation to the Ontology PSIG
22nd of September 2022

Charles Dickerson — Loughborough University
Michael Wilkinson — BAE SYSTEMS
Siyuan Ji — University of York

Why This Presentation?

- The transformations in the paper mathsig/2022-09-01 mark the practical completion of the 2nd objective in the Math DSIG mission statement [https://www.omg.org/maths/].
- Future work includes advancing the formalisms in the mathsig paper to address emerging challenges in the integration of digital engineering and MBSE.
- Collaboration between the Math DSIG and the Ontology PSIG is considered a critical step for this mission.

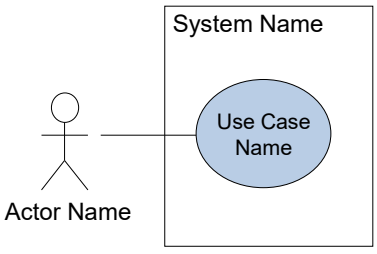
Agenda

- Mathematical Formalisms for Model Transformation
  - ***Matrix Representation of Graphical Models***
  - Transformation of Semantic Structures
- Review of Interpretation of Language into Graphical Models
- Interpretation and Transformation: Mathematical Metamodel

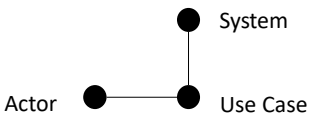
Use Case Diagram as a Graphical Structure

Graphical structure is similar to what one would find in any model developed using a standardised language such as UML/SysML:

UML Use Case Diagram



Mathematical Graph of Diagram



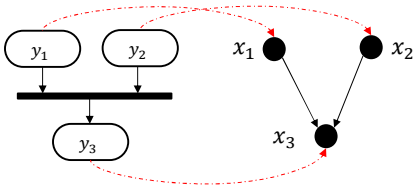
Matrix Representation of Graph

	Actor	Use Case	System
Actor		Association	
Use Case			Allocation
System			

ROSETTA Framework

The Relational Oriented Systems Engineering Technology Trade-off and Analysis (ROSETTA) framework is a matrix representation of a relational orientation on systems.

The figure is a simple example of how it can be applied to the object-oriented graphical models of UML and SysML. The Design Objectives y_1, y_2, and y_3 could be three activities from an Activity Diagram in which the first two must be performed before the third one.



Target Model (Design Variables):

Source (Design Objectives):

M	y_1	y_2	y_3	N	x_1	x_2	x_3	Q
			✓	y_1	✓			
			✓	y_2		✓		
				y_3			✓	

The matrix (Q) is the functional allocation $y_1 \rightarrow x_1$, $y_2 \rightarrow x_2$, and $y_3 \rightarrow x_3$

$(y_i, y_j) \in \boldsymbol{M}$ with $(y_i, x_k), (y_j, x_l) \in \mathbf{Q}$ implies $(x_k, x_l) \in \boldsymbol{N}$

Terminology for Architecture Definition*

- Ontology-driven Architecture
  - Interpretation of knowledge into models
  - Transformation of knowledge between models
- Semantic Structure
  - Loosely, a 'clean' structure with semantic types such as object-oriented diagrams that use a standardised language like UML/SysML
- Domain Structure
  - Semantic structure populated with domain knowledge
- Semantic Transformation
  - A mapping that preserves semantic and domain structures of models
- Interpretation
  - A type of mapping of domain knowledge into a semantic structure
  - A mathematical example is interpretation in first order model theory

*In the sense of ISO/IEC/IEEE 15288: 2015

Making Natural Language Precise
Semantic Structure: a representation of ontology?

- There are many definitions of the term *structure* e.g.,

Noun: Mutual relation of the constituent parts or elements of a whole as determining its peculiar nature or character; make, frame.

Adjective: Organized or arranged so as to produce a desired result. Also, loosely, formal, organized, not haphazard.

- Formal definitions in and derived from [1, 2]

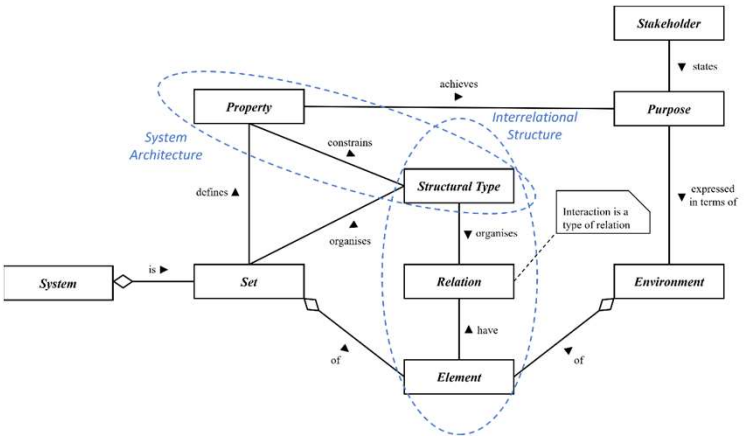
Structure is junction and separation of the objects of a collection defined by a property of the collection or its objects.

Semantic structure is a structure whose objects are semantic types.

Definition of Other Key Terms

- *A Semantic Class Structure* is a semantic structure whose types have been instantiated by (mathematical) classes.
- *Architecture* is structural type in conjunction with consistent properties that can be implemented in a class of structure of that type [1, 2].
- In first order model theory, a *Model* is a relational structure for which the interpretation of a sentence in the Predicate Calculus becomes valid (true).
- A *System* is a set of interrelated elements that comprise a whole, together with an environment [1, 2].

A Logical Model
Is this a representation of ontology?



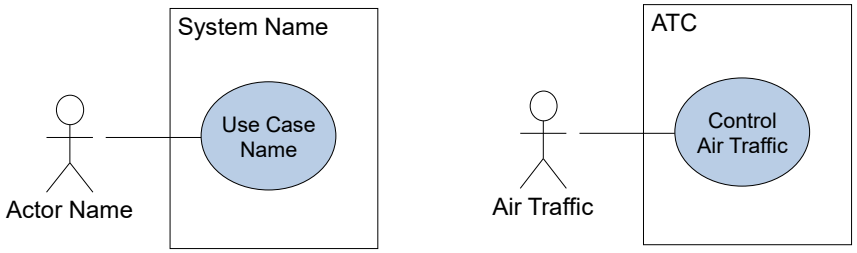
©Figure 4.11 Integrated model of architecture and system [2]

Agenda

- Mathematical Formalisms for Model Transformation
  - Matrix Representation of Graphical Models
  - ***Transformation of Semantic Structures***
- Review of Interpretation of Language into Graphical Models
- Interpretation and Transformation: Mathematical Metamodel

Semantic Structure Interpreted into Domain Structure*

Semantic Structure: loosely, a 'clean' structure with semantic types such as object-oriented diagrams that use a standardised language like UML/SysML



Semantic Structure

Domain Structure

*refer to previous PSIG presentation [3]: language → structure

ATC: Air Traffic Control

Semantic Transformation [3]

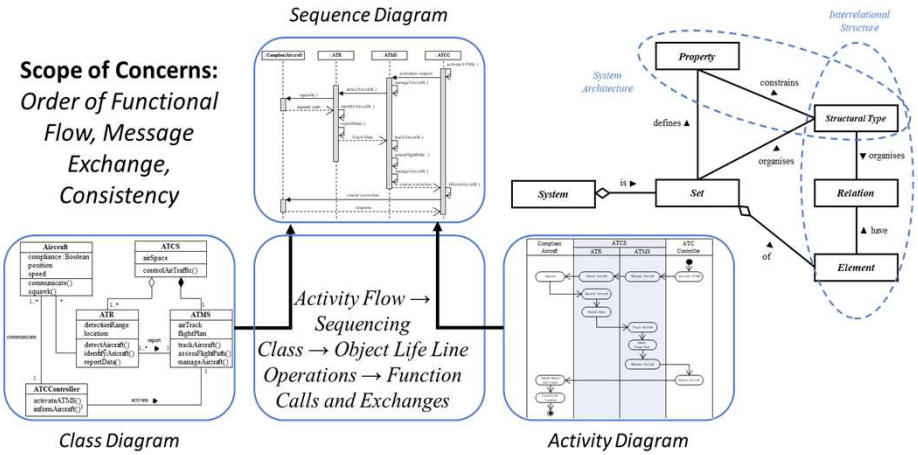
- Structure developed in different views (diagrams) of the system model are interconnected based on semantics of the underlying modelling languages
  - e.g., A set of **Actions** *structured* in a specific way to achieve the functionality as modelled in a **Use Case**
- Transforming from a diagram (e.g. Use Case Diagram) to a semantically richer diagram (e.g. Activity Diagram) requires engineering knowledge
- *Semantic transformation* provides a way to preserve semantic structures from one view to another
  - Ensuring consistency
  - Guiding engineering effort

In what follows, refer to Mathsig/2022-09-01 [4]

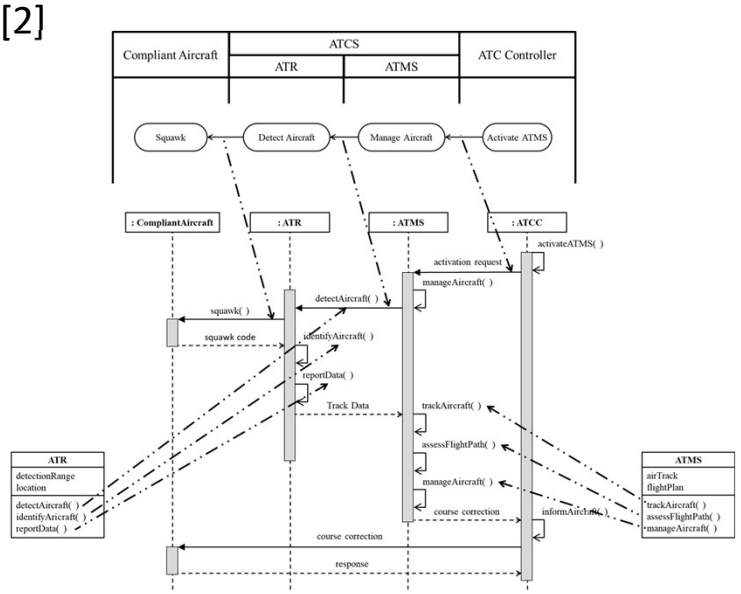
Class + Activity Diagram → Sequence Diagram

Logical models are needed! [2]

Scope of Concerns: *Order of Functional Flow, Message Exchange, Consistency*



Transformation Details [2]



Agenda

- Mathematical Formalisms for Model Transformation
  - Matrix Representation of Graphical Models
  - Transformation of Semantic Structures
- ***Review of Interpretation of Language into Graphical Models***
- Interpretation and Transformation: Mathematical Metamodel

Review of Observations [3]

- Semantic Transformation and Interpretation are NOT done in series: a maths basis and joint cognitive tools are needed
- Not all information can be interpreted into just one semantic structure
  - Domain knowledge interpreted in to a set of semantic structure, representing different views → full system model
  - Information interpreted into different semantic structures, with guided rules (MBSE methodologies!)
- One semantic structure can contain more information than what is provided
  - Tacit knowledge and innovation
  - Increasing design commitment

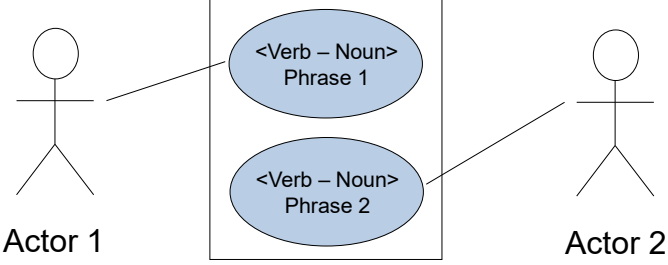
What is a Predicate? [1, 2]

Generic example of use cases for a system as a semantic structure

Note that the System and <verb - noun> phrase always form a sentence.

Actor <interacts with> system.

System <interacts with> actor.



Actor 1

(e.g., subject of verb)

Actor 2

(e.g., object of verb)

A predicate is that part of a sentence containing a verb and stating something about the subject.

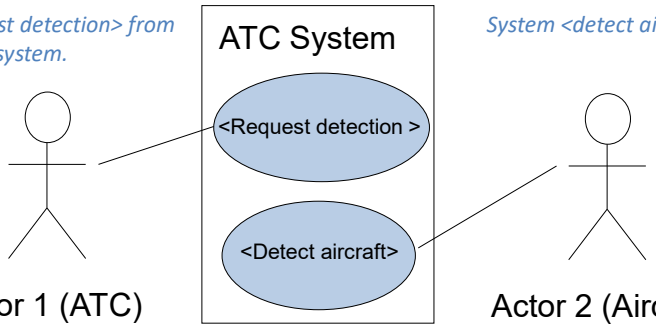
What is a Predicate? [1, 2]

Specific example of use cases for a radar system

Note that the System and <verb - noun> phrase always form a sentence.

ATC <request detection> from system.

System <detect aircraft>.



Actor 1 (ATC)

(e.g., subject of verb)

Actor 2 (Aircraft)

(e.g., object of verb)

A predicate is that part of a sentence containing a verb and stating something about the subject.

ATC: Air Traffic Control

Agenda

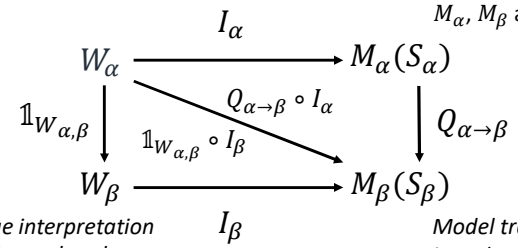
- Mathematical Formalisms for Model Transformation
  - Matrix Representation of Graphical Models
  - Transformation of Semantic Structures
- Review of Interpretation of Language into Graphical Models
- ***Interpretation and Transformation: Mathematical Metamodel***

A Supporting Mathematical Formalism [5]

Synchronisation through structure preserving transformations:

- Semantic Transformation, $Q_{\alpha\to\beta}$ that preserves one structure into another, e.g. Use Case to Activity Diagram, without knowledge content populated
- Interpretations, I_α and I_β, that map domain knowledge into the structures to make them domain models

W_α, W_β are knowledge based on ontology



Knowledge interpretation needs to be explored

Model transformation is understood

What Next?

References

[1] Dickerson, C.E. et al, 2021. "Architecture definition in complex system design using model theory." *IEEE Systems Journal* 15, no. 2: 1847 – 1860.

[2] Dickerson, C.E. and Siyuan Ji, *Essential Architecture and Principles of Systems Engineering*. Boca Raton Florida: CRC Press Auerbach Publication, September 2021.

[3] Ji S., C.E. Dickerson & M.K. Wilkinson, "UPR: Architecture Specification with Structures and Transformations", presentation to Ontology PSIG, ontology/22-03-02

[4] Dickerson, C.E., S. Ji & M. K. Wilkinson, "SysML Model Transformations Using Relational Orientation", Mathematical Formalism DSIG Paper, mathsig/2022-09-01

[5] Ji, S., M.K. Wilkinson & C.E. Dickerson, "Structure Preserving Transformations for Practical Model-based Systems Engineering", arXiv:2209.07935, to appear in the 8th IEEE International Symposium on Systems Engineering (ISSE 2022)