Designing Systems of Systems With Category Theory

Towards Ontology-Informed Mathematical Foundations of Systems Architecture

Monte Mahlum

Tapestry RDI

February 17, 2026

Section 1

Applied Category Theory, Briefly

Applied Category Theory, Briefly

The What

Definition

A *category* $\mathbf{C}$ consists of

- a set of *objects* $\mathrm{Ob}(\mathbf{C})$,
- for each pair $C, D \in \mathrm{Ob}(\mathbf{C})$, a set of *morphisms* $\hom_{\mathbf{C}}(C, D)$, and
- a (unital, associative) composition rule

$$\begin{tikzcd}
X \arrow[r, "g"] \arrow[dr, "f \circ g"'] & Y \arrow[d, "f"] \\
 & Z
\end{tikzcd}$$

Applied Category Theory, Briefly

The What

Definition

A *category* $\mathbf{C}$ consists of

- a set of *objects* $\mathrm{Ob}(\mathbf{C})$,
- for each pair $C, D \in \mathrm{Ob}(\mathbf{C})$, a set of *morphisms* $\hom_{\mathbf{C}}(C, D)$, and
- a (unital, associative) composition rule

$$\begin{tikzcd} X \arrow[r, "g"] \arrow[dr, "f \circ g"'] & Y \arrow[d, "f"] \\ & Z \end{tikzcd}$$

Examples

Set, **Vect**, **Grph**, a single graph **G**, **Poset**, a single poset **P**, and importantly $\mathbf{Sh}(X)$,...

Applied Category Theory, Briefly

A Sliver of Why

Vision

Category Theory powering compositional reasoning across engineering, computation, and the sciences.

Motivating Observation

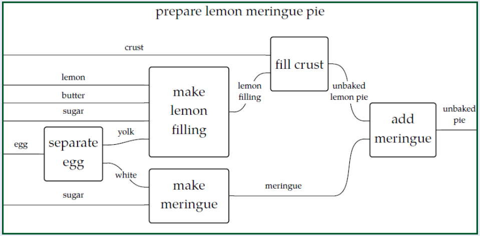
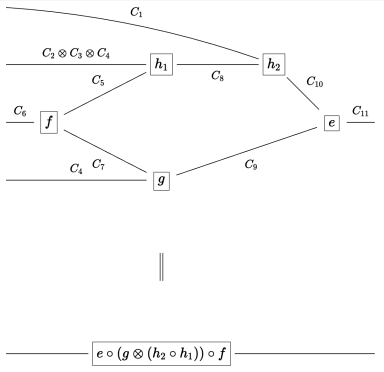


Figure: Above diagram courtesty of Brendan Fong and David I. Spivak. (2019) “Seven Sketches in Compositionality.” In: An Invitation to Applied Category Theory. Cambridge University Press.



Applied Category Theory, Briefly

A Sliver of Why

Vision

Category Theory powering compositional reasoning across engineering, computation, and the sciences.

Personal Outlook

Category Theory provides

1. an extensive library of data structures *and their transformations*,
2. theorems about their different representations, and
3. a framework for building more.

Section 2

Systems Architecture, Briefly

Systems Architecture, Briefly

The What

Definition

A *system architecture* is a conceptual model defining the structure, behavior, and views of a given system[1]. It is the language in which requirements and their solutions are declared and verified.

Hannu Jaakkola and Bernhard Thalheim. (2011) "Architecture-driven modelling methodologies." In: Proceedings of the 2011 conference on Information Modelling and Knowledge Bases XXII. Anneli Heimbürger et al. (eds). IOS Press. p. 98

.

Systems Architecture, Briefly

The What

Definition

A *system architecture* is a conceptual model defining the structure, behavior, and views of a given system[1]. It is the language in which requirements and their solutions are declared and verified.

Hannu Jaakkola and Bernhard Thalheim. (2011) "Architecture-driven modelling methodologies." In: Proceedings of the 2011 conference on Information Modelling and Knowledge Bases XXII. Anneli Heimbürger et al. (eds). IOS Press. p. 98

.

Example: National Airspace System (NAS)

- Capability: safe separation.
- Requirement: an aircraft must communicate with Air Traffic Control prior to landing.
- Future Revision: an aircraft must communicate with neighboring aircraft prior to landing.

Systems Architecture, Briefly

The What

Definition

A *system architecture* is a conceptual model defining the structure, behavior, and views of a given system[1]. It is the language in which requirements and their solutions are declared and verified.

Hannu Jaakkola and Bernhard Thalheim. (2011) "Architecture-driven modelling methodologies." In: Proceedings of the 2011 conference on Information Modelling and Knowledge Bases XXII. Anneli Heimbürger et al. (eds). IOS Press. p. 98
.

Example: National Airspace System (NAS)

- Capability: safe separation.
- Requirement: an aircraft must communicate with Air Traffic Control prior to landing.
- Future Requirement: an aircraft must communicate with neighboring aircraft prior to landing.

$$\text{requirements} \xrightarrow{\text{enable}} \text{capabilities}$$

Systems Architecture, Briefly

Why & How

The Central Problem

Critical systems across aviation, energy, and technology are undergoing dramatic transformation while amassing architectural (or “operational” or “system”) entropy

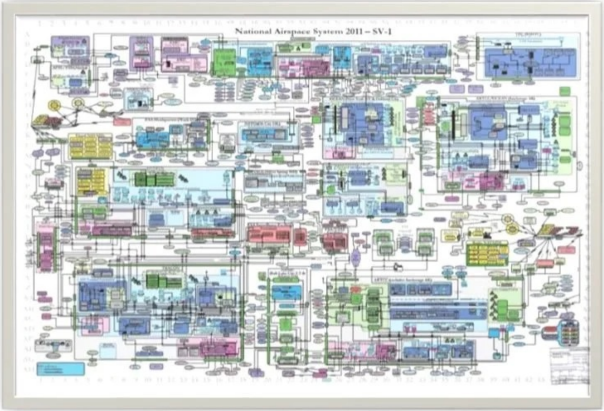


Figure: A simplified schematic of the NAS architecture; source unknown.

Systems Architecture, Briefly

Why & How

The Central Problem

Critical systems across aviation, energy, and technology are undergoing dramatic transformation while amassing architectural (or "operational" or "system") entropy.

The Solution: Interpretable Systems Architecture (ISA)

- Systems Ontology to integrate concept and legacy architectures together with test and operational data.
- Ontology-informed, category-theoretic structures and their transformations for novel architectural reasoning and as scaffolding for analysis & ML.

Systems Architecture, Briefly

Why & How

The Central Problem

Critical systems across aviation, energy, and technology are undergoing dramatic transformation while amassing architectural (or “operational” or “system”) entropy.

The Solution: Interpretable Systems Architecture (ISA)

- Systems Ontology integrates concept and legacy architectures with test and operational data
- Ontology-informed, category-theoretic structures and their transformations for novel architectural reasoning and as scaffolding for analysis & ML.

Vision (Tapestry)

An efficient and adaptive civilian society in command of its systems through their architectures.

Mission (Tapestry)

Enable rapid evolution of complex systems in a safe, interpretable, and cost-effective manner.

Section 3

Ontological Foundations

Ontological Foundations

The Basic Formal Ontology (BFO)

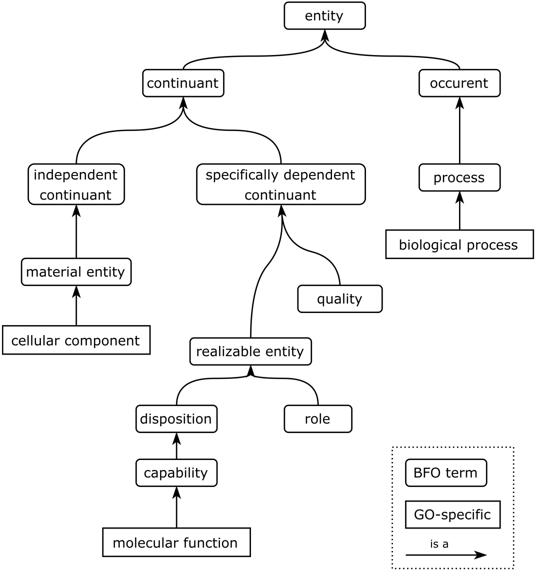


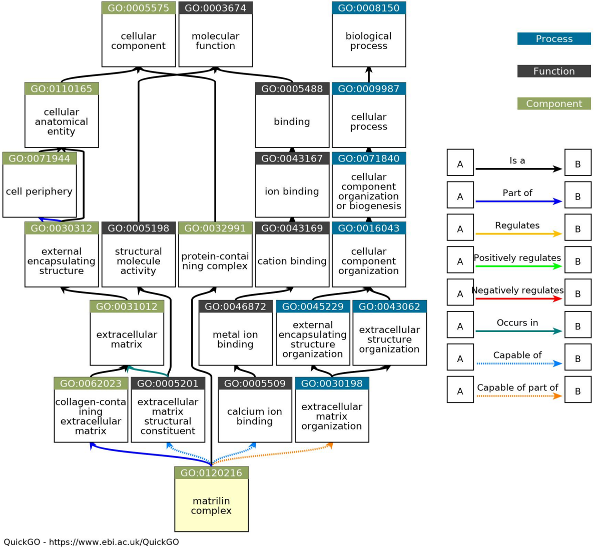
Figure: A sample of terms from BFO. The vocabulary is further connected by relations not shown here, e.g., every process has participating independent continuants.

Ontological Foundations

The Gene Ontology (GO) as Inspiration

Figure: A sample of terms from the Gene Ontology and their relation to *matrilin complex* (bottom), a type of cellular component.

Sourced from EMBL's European Bioinformatics Institute, CC0, via Wikimedia Commons



Ontological Foundations

The Systems Ontology (SO)

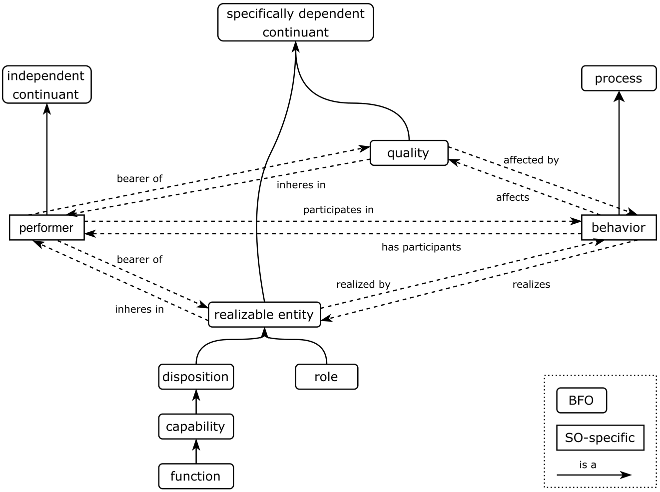


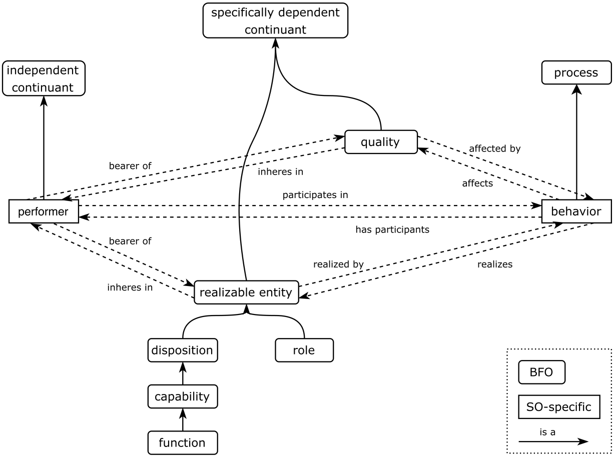
Figure: The core terms and relations of SO, including the central realization pattern (lower dashed triangle).

Ontological Foundations

The Systems Ontology (SO)

Interpretation

Systems Ontology is a lightweight *framework* more than a complete ontology in itself.



Ontological Foundations

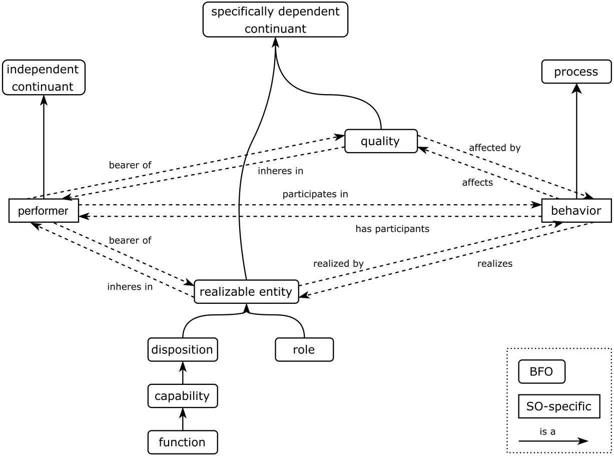
The Systems Ontology (SO)

Interpretation

Systems Ontology is a lightweight *framework* more than a complete ontology in itself.

Intended Utility

A double-categorical schema to build domain-specific system ontologies.

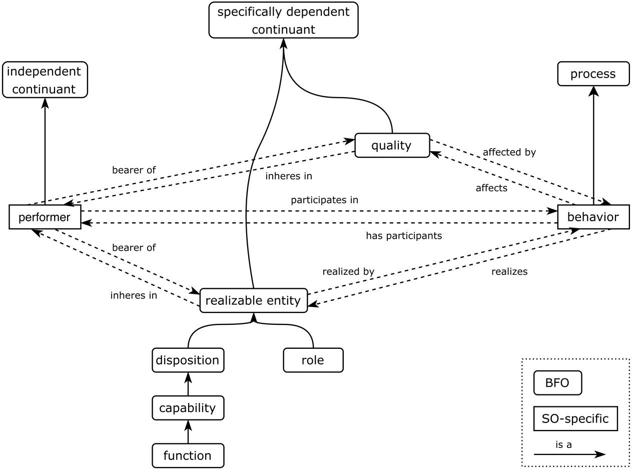


Ontological Foundations

The Systems Ontology (SO)

Examples of Possible Performer Types

- pilot or plane,
- computer,
- measurement tool,
- cellular component (GO),
- any other class of physical components deemed relevant to the system,

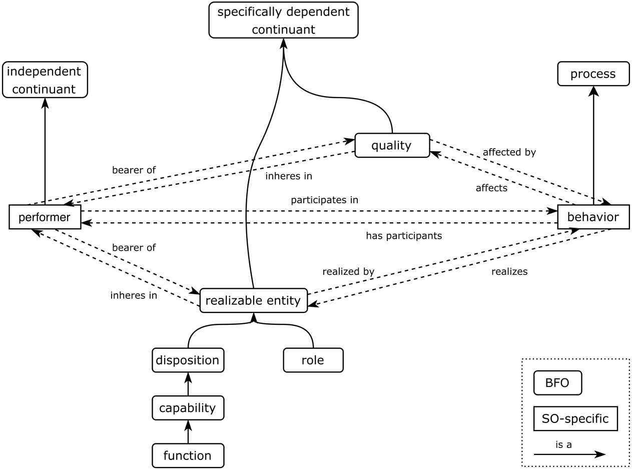


Ontological Foundations

The Systems Ontology (SO)

Examples of Possible Behavior Types

- take off,
- high-altitude maneuver,
- photovoltaic conversion,
- signal transduction (GO),
- DNA replication (GO)



Ontological Foundations

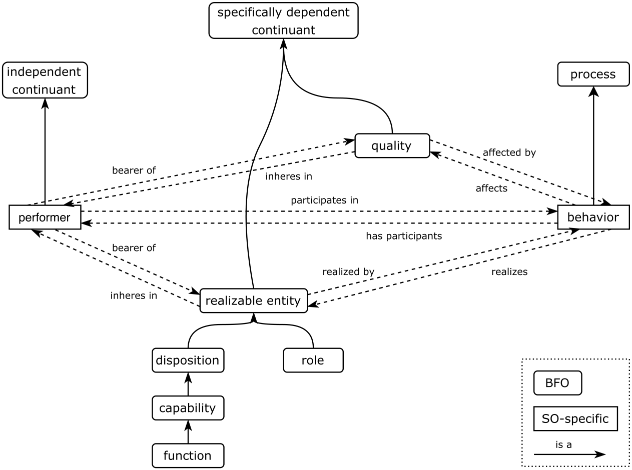
The Systems Ontology (SO)

Examples of Possible (Universal) Qualities

- temperature,
- voltage,
- protein configuration,
- location and velocity (although not standard BFO qualities).

Definition

A *quality space* S_Q for a quality Q is a set of possible values that a measurement of Q can take, e.g., if $Q = \mathrm{temperature}$, S_Q may be $\mathbb{R}_{\geq 0}$ interpreted in units of Kelvin.

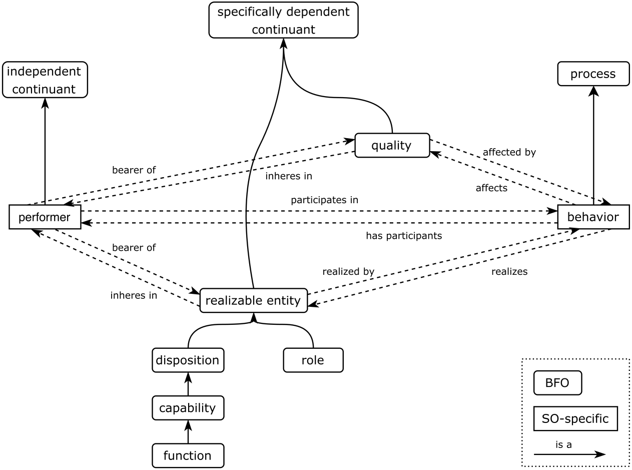


Ontological Foundations

The Systems Ontology (SO)

Examples of Possible Capability Types

- emergency water landing,
- *safe* emergency water landing,
- photovoltaic conversion,
- photovoltaic conversion with efficiency $\geq e$.
- signaling receptor binding (GO).



Section 4

Ontology-Informed Mathematical Structures

Ontology-Informed Mathematical Structures

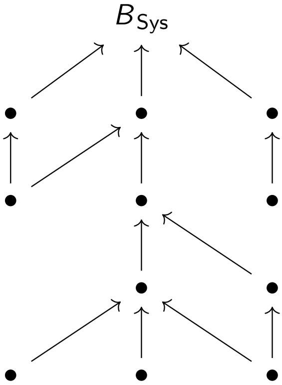
The Behavior Hierarchy

Definition

Let **B** denote the poset whose objects are system behavior types with a morphism $B \to B'$ if B' is a *subbehavior* of B'.

Remark

While **B** and $\downarrow\boxed{\text{behavior}}$ have the same objects, **B** typically has many more morphisms.



Ontology-Informed Mathematical Structures

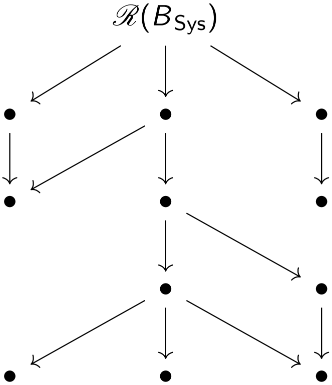
The Requirement Presheaf

Definition

Let $\mathbf{B}$ denote the poset whose objects are system behavior types with a morphism $B \to B'$ if B' is a *subbehavior* of B'.

Definition

Let $\mathscr{R} : \mathbf{B}^{\mathrm{op}} \to \mathbf{Set}$ be a presheaf declaring requirements on behaviors.



Ontology-Informed Mathematical Structures

The Performer Hierarchy

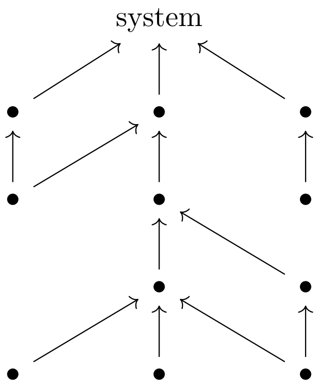
Definition

Let $\mathbf{P}$ denote the poset (or lattice, or frame), whose objects are those performers in the system with a morphism $p \to q$ if and only if p is *physically contained in* q.

All posets are treated as sites with the *minimal coverage* where $(p_i \to p)_i$ covers p if and only if for all minimal $q \to p$, $q \to p_i$ for some i. Here q is minimal if $q' \to q$ implies $q' = q$ or $\emptyset$.

Remark

While every $p \in \mathbf{P}$ will have a declared type $P \xrightarrow{\text{is a}} \boxed{\text{performer}}$, the structure $\mathbf{P}$ is distinct from $\downarrow\boxed{\text{performer}}$.



Ontology-Informed Mathematical Structures

Quality Sheaves and their Morphisms

Definition

Let $\mathscr{Q} : \mathbf{P}^{\mathrm{op}} \to \mathbf{Set}$ be a sheaf declaring quality spaces[1], and thus qualities, for each performer.

Remark

$\mathscr{Q}$ defines a state space $\mathscr{Q}(\mathrm{system})$ and records how these states descend to local regions of the system.

$$
\begin{tikzcd}
 & \prod_{i \in I_1} S_{Q_i} \arrow[dl] \arrow[d] \arrow[dr] & \\
\bullet \arrow[d] & \bullet \arrow[dl] \arrow[d] & \bullet \arrow[d] \\
\bullet & \prod_{i \in I_2 \cup I_3} S_{Q_i} \arrow[d] \arrow[dr] & \bullet \\
 & \prod_{i \in I_2} S_{Q_i} \arrow[dl] \arrow[d] \arrow[dr] & \prod_{i \in I_3} S_{Q_i} \arrow[d] \\
\bullet & \bullet & \bullet
\end{tikzcd}
$$

[1] For each (universal) quality $Q \xrightarrow{\text{is a}} \boxed{\text{quality}}$, there are many possible “quality spaces” S_Q which are sets of possible measurement values for Q each with respect to a different unit of measurement.

Ontology-Informed Mathematical Structures

Quality Sheaves and their Morphisms

Definition

Let $\mathscr{Q} : \mathbf{P}^{\mathrm{op}} \to \mathbf{Set}$ be a sheaf declaring quality spaces[1], and thus qualities, for each performer.

Remark

$\mathscr{Q}$ defines a state space $\mathscr{Q}(\mathrm{system})$ and records how these states descend to local regions of the system.

[1] For each (universal) quality $Q \xrightarrow{\text{is a}} \boxed{\text{quality}}$, there are many possible "quality spaces" S_Q which are sets of possible measurement values for Q each with respect to a different unit of measurement.

Covariant Grothendieck Construction

$(\mathbf{P}, \mathscr{Q}) \in \int^{\mathrm{cov}}_{\mathbf{FinPos}^{\mathrm{op}}} \mathrm{Sh}(-)$ where

$$\begin{aligned} \mathrm{Sh}(-) : \mathbf{FinPos}^{\mathrm{op}} &\to \mathbf{Cat} \\ \mathbf{P} &\mapsto \mathrm{Sh}(\mathbf{P}) \\ (\mathbf{P} \xrightarrow{f} \mathbf{P}') &\mapsto f_* \end{aligned}$$

A *quality sheaf morphism*

$$(\rho, \alpha) : (\mathbf{P}, \mathscr{Q}) \to (\mathbf{P}', \mathscr{Q}')$$

is then a pair

$$\begin{aligned} \rho : \mathbf{P}' \to \mathbf{P} &\quad \text{in } \mathbf{FinPos} \\ \alpha : \rho_* \mathscr{Q} \to \mathscr{Q}' &\quad \text{in } \mathrm{Sh}(\mathbf{P}') \end{aligned}$$

Ontology-Informed Mathematical Structures

Quality Sheaves and their Morphisms

$$
\begin{tikzcd}
& \text{system} & & & & & \text{system}' \arrow[lllll, "\rho"'] & \\
p_1 \vee p_2 \arrow[ur] & p_1 \vee p_3 \arrow[u] & p_2 \vee p_3 \arrow[ul] & & & p_1 \vee p_2 \arrow[ur] & p_1 \vee p_3 \arrow[u] & p_2 \vee p_3 \arrow[ul] \\
p_1 \arrow[u] \arrow[ur] & p_2 \arrow[ul] \arrow[ur] & p_3 \arrow[ul] \arrow[u] & & & & & \\
& {} \arrow[d, "\mathscr{Q}"'] & & & {} \arrow[dl, "\rho_*\mathscr{Q}"'] \arrow[r, Rightarrow, "\alpha"] & {} & {} \arrow[d, "\mathscr{Q}'"] & \\
& L \prod L \prod L \arrow[dl] \arrow[d] \arrow[dr] & & {} & & & S \prod S \prod S \arrow[dl] \arrow[d] \arrow[dr] & \\
L \prod L \arrow[d] \arrow[dr] & L \prod L \arrow[dl] \arrow[dr] & L \prod L \arrow[dl] \arrow[d] & & & S & S & S \\
L & L & L & & & & &
\end{tikzcd}
$$

Figure: Depiction of a quality sheaf morphism enabling abstraction. One interpretation of this morphism is as follows: the performers p_i denote airplanes in an airspace system, the quality space L is for the *aircraft location* quality, while $S = \{\text{unsafe}, \text{safe}\}$ is for the pairwise *seperation safety* quality.

Ontology-Informed Mathematical Structures

Towards a Unified Narrative

Guiding Principle

While behaviors, capabilities, and requirements are ontologically distinct, they should be represented in the same mathematical language using quality sheaves.

Behaviors & Capabilities

Towards a Unified Narrative

Guiding Principle

While behaviors, capabilities, and requirements are ontologically distinct, they should be represented in the same mathematical language using quality sheaves.

Difficulty

There are many disparate means of modeling processes:

- natural language specification (assumptions and guarantees),
- black box relation,
- wiring diagram,
- dynamical system, etc.

Ontology-Informed Mathematical Structures

Towards a Unified Narrative

Guiding Principle

While behaviors, capabilities, and requirements are ontologically distinct, they should be represented in the same mathematical language as a transformation of a quality sheaf.

Future Work

... Lots

Difficulty

There are many disparate means of modeling processes:

- natural language specification (assumptions and guarantees),
- black box relation,
- wiring diagram,
- dynamical system, etc.

Some Semi-Approachable References

Systems ACT

1. M.M., Samantha Jarvis, Nelson Niu, Angeline Aguinaldo, Amanda Hicks, and Ian Levitt. *Formal Structures in Systems Ontology towards Air Traffic Management Architectures*. NASA Technical Memorandum, 2025. TM-20250010771.
2. * Brendan Fong and David I. Spivak. *An Invitation to Applied Category Theory: Seven Sketches in Compositionality*. Cambridge University Press, 2019. Arxiv:1803.05316.

Ontology ACT

1. * David I. Spivak and Robert E. Kent. *Ologs: a categorical framework for knowledge representation*. arXiv preprint, 2011. Arxiv:1102.1889.
2. Evan Patterson. *Knowledge Representation in Bicategories of Relations*. arXiv preprint, 2017. arXiv:1706.00526. Arxiv:1706.00526.

CT Generally

1. Tai-Danae Bradley, Tyler Bryson, and John Terilla. *Topology: A Categorical Approach*. The MIT Press, 2020. Prepublication version.
2. Emily Riehl. *Category Theory in Context*. Dover Publications, 2016. Prepublication version.