Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Radical Transitions in Generic Design: Frame Semantic Hypergraphs and Topological Diagrams in C/K Theory

Ira Monarch

iramonarch@gmail.com

Research Article

Keywords: Generic design, C/K theory, Topos theory, Frame semantics, Hypergraph/Topological visualization, Radical transitions

Posted Date: November 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11403138/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Radical Transitions in Generic Design:

Frame Semantic Hypergraphs and Topological Diagrams in C/K Theory

Ira Monarch*

* Independent Researcher, Pittsburgh, PA, USA

iramonarch@gmail.com ORCID 0000-0002-2587-173X

Acknowledgements. The author thanks the members of the BGI Nexus working group, and in particular the team developing a process and platform for human–AI synergy, for discussions that shaped the perspective of this paper.

Abstract. Generative AI appears to be forcing a radical transition whose outcome cannot yet be known, and past transitions in the technics of language suggest that utopian and dystopian outcomes tend to arrive together. Drawing on the linguistic turns that followed those transitions, we approach design as the composition of conceptual materials grounded in language use, guided, in Sellars's reading of Plato, by the soul as craftsman. We supplement C/K (Concept-Knowledge) design theory with frame semantic hypergraphs built in TypeDB (FSN-TDB), in which C becomes a schema of framed possibilities and K their instantiation by textual evidence, and we extend this to graph-directed design execution. We then argue that C/K Topos theory, built around creative preservation, cannot address radical transitions in which concepts and even data change, and show how thought experiments and impossible figures analyzed through sheaf cohomology can extend it. Frame semantic hypergraphs and topological diagrams prove complementary: the first show discrete structure, the second continuous coverage, strain and gaps. We conclude that design can right an imbalance between the correctness of the picture and the correctness of the performance. The paper was itself synergistically designed with an AI participant.

Keywords: *Generic design; C/K theory; Topos theory; Frame semantics; Hypergraph/Topological visualization; Radical transitions.*

1. Introduction and Background

Our aim is to extend design theory to come to terms with what appears to be an imminent radical technological transition being forced by Large Language Models (LLMs). As with all interesting transitions, its participants do not know how it will be seen in retrospect, what sustained it, or what it will transition to. It crosses the boundaries of the arts, philosophy, the sciences, engineering, the political, and what might be called the ordinary. We[1] propose that it crosses these boundaries on the basis of a linguistic passport, one stemming from three linguistic turns.

1.1 The Language Turn that Made Way for the Anthropocene

The oldest linguistic turn separated modern humans from other proto-humans and catalyzed the Anthropocene. Language functions as a revolutionary form of "*mnemotechnics*" (techniques for creating memory external to the mind) (Stiegler 1998), and for Stiegler the question of technics as "tertiary retention" opens "a new perspective on political economy and its critique" (Stiegler 2010). Other animals adapt to their environments through genetic change over millions of years; modern humans used language to make the environment adapt to them. Pagel describes how language overcame the limits of "visual theft," in which a skill spreads only by watching and copying: language packages complex, abstract instructions for transmission, so that knowledge accumulates across generations instead of dying with the individual (Pagel 2011). Dunbar argues that language let humans replace physical grooming with gossip and storytelling, enlarging the groups they could sustain (Dunbar 1996), and shared myths, laws and abstractions such as money and borders aligned thousands of unrelated individuals toward common goals (Harari 2015). Language also turned humans from niche constructors, like beavers and their dams, into "niche modifiers" who coordinate labor, transmit engineering knowledge and model the deep future, changing ecosystems irrespective of natural adaptive pressures (Low et al. 2019). Pinker (2010) adds that language is a "non-rival good": it can be duplicated and traded at no loss to its original owner. Stiegler would disagree. For him, language as mnemotechnics opens the pharmacological question, in which *hypomnesis*[2] is a *pharmakon,* at once poison and remedy, and mnemotechnics can lead to inequitable outcomes. Language is not the be-all and end-all of human existence: for Heidegger, care is being's inchoate guide to language; for Laruelle, the real is its guide in the last instance. While these are important considerations, we do not pursue them here. Later transitions in the technics of language show the same pharmacological polarity. On the one hand, the advent of writing preserved knowledge across generations, freeing the mind from the burden of memorization (Ong 2012). Derrida (1997) adds that writing, or arche-writing, must be distinguished from the alphabetic writing privileged by a Western bias he calls phonocentrism: the flawed assumption that writing is valuable only when it faithfully transcribes speech, and so offers no real alternative to it. On the other hand, writing forced a visual, specialized and individualistic way of viewing reality, fracturing communal bonds and creating new social hierarchies as a tool of political control (McLuhan 1962; Lévi-Strauss 1974). Similarly, Eisenstein (1979) saw the printing press as a stabilizing agent of progress that improved the fidelity of scientific data and historical preservation, while McLuhan (1962) saw it as a disruptive technology that fractured the balance of the senses, leading to psychological isolation and nationalist warfare. McLuhan went beyond speech, writing and print to include electronic media as "wrapping around" and integrating previous technics of language (McLuhan 1964)**.** LLMs and GenAI portend an even greater polarity, uncertainty and precarity.

---

[1] Here the use of 'We' harkens to the use of the royal we in academic literature, but it will be used in this essay in another way. We are not trading individual subjectivity for collective authority, that is, as an Imaginary defense mechanism to make the author appear unified, objective, and beyond reproach. We are using it in the sense of Lacan's Big Other, which is intended to fundamentally dismantle Kant's *"I think"* (the transcendental unity of apperception) by stripping the subject of its foundational autonomy, self-mastery, and internal coherence. For Lacan, the Big Other does not exist as a complete, harmonious entity; it is barred, lacking, and riddled with inconsistencies. The dismantling of the Kantian *"I think"* is therefore double: the subject is shattered by an absolute Symbolic Order, only to discover that this order itself is an incomplete, fragile fiction—perhaps in the guise of GenAI.

[2] In the dialog, *Phaedrus*, Plato famously contrasted anamnesis (living, internal recollection arising directly within the soul) with *hypomnesis* (passive reminding through external signs or inscriptions). In the myth of Theuth, the Egyptian king Thamus critiques the invention of writing, arguing it provides an elixir not of true memory (*mneme*), but merely of reminding (hypomnesis), which will cause people to become forgetful and rely on "dead" external marks rather than active, internal understanding.

1.2 Two Linguistic Turns

Two much more recent linguistic turns followed. One has its roots in twentieth-century analytic pragmatism, itself stemming from the work of C. S. Peirce. The other emerged across linguistics, anthropology, political philosophy and psychoanalysis in structuralism, and continued in post-structuralism, associated with Continental philosophy. Despite their differences, both focus on linguistic functions and structures, meaningful textual units standing for concepts, prototypes, frames, performatives (including interpellations), and paradigms. Read together, stereoscopically, like a counterpoint in music, the two turns cover at least two strains of philosophy, linguistics, anthropology, sociology, political economy, the history of science, art and engineering, and semiotics, extending to biosemiotics and collective intelligence. "Linguistic" here also includes non-disciplinary reflection on ordinary language use, as in Wittgenstein and Austin. Both turns develop the idea of concept or language engineering: in the Anglo-American tradition from Carnap's formal language ideal through Sellars, Goodman and Floridi, and in the Continental tradition in Deleuze and Guattari's *What is Philosophy?* in Derrida and Stiegler, and in Laruelle's non-philosophy, whose materialist conceptual engineering unfolds into generic design.

Our aim is to view design theory as the composition of conceptual materials grounded in language use that turn into design implementations. Seen this way, the linguistic turns are colliding with the uncertain impacts of the new LLM technologies, which may bring radical transitions. Design theory can be instrumental in designing these technologies, and they in turn can be instrumental in extending design theory. We pursue this mutual interaction by exploring how C/K (Concept-Knowledge) design theory[3] can be supplemented with neuro-symbolic techniques that combine the capabilities of neural GenAI with *Frame Semantic Networks* (FSNs). Such networks are the basis for composing conceptual materials grounded in language use, and they may also help in understanding what patterns an LLM is using and in communicating with it. We currently use Claude Opus 5.5 as the LLM that generates Frame Semantic Networks (FSNs; see (FrameNet 2026; Fillmore 1982)) from textual data. These are realized as schema definitions and data insertions in the TypeDB hypergraph database (TypeDB 2026), whose Studio interface (TypeDB Studio 2026) supports searching, visualizing and manipulating them. We call this the Frame Semantic Network TypeDB (FSN-TDB) approach, and we argue that it can supplement C/K theory in exploring the full range of generativity in design, whether *creative preservation* or *creative innovation*.[4]

We use this discussion to think through the ambiguous promise of GenAI, doom or utopian abundance, given the precarity of the times and its emotional and ethical ramifications. A linguistic perspective of this kind must include human emotion and human flourishing in the design process it supports, and it requires forging common ground between humans, and between humans and GenAIs; from the human side, the interaction can feel like conversing with something not yet categorized. To this end, we identify some of the more important contexts and create constructs that grasp them in ways conducive to generating new concepts and new ways of composing them in practice. Articulating where we stand in the GenAI transition depends on a kind of generativity that measures up to the conceptual changes it requires. The frame-semantic, performative, networked constructs we draw on are compatible with the formalism of C/K theory but allow a more comprehensive conception of context and practical performance. We propose a design science that does not dismiss utopian thinking and that takes steps to avoid what appears to many as an inevitable dystopian transition.

1.3 Plato's Forms as Linguistic Functions or Conceptual Materials: Plato, Sellars and Laruelle

To connect design theory and philosophy with the linguistic turn, we start with Sellars's approach to linguistic functions and patterns and his reading of Plato's Timaeus (Sellars 1967; Plato 2016), figured as conceptual engineering, conceptual design or generic design. We juxtapose for stereoscopic or counterpoint reading his book, *Science and Metaphysics: Variations on Kantian Themes* (Sellars 1968) with his essay, "The Soul as Craftsman: an Interpretation of Plato on the Good" (Sellars 1967). We argue that read through Sellars, the Timaeus offers a

[3] For an overall view of C/K theory see (Hatchuel and Weil 2009; Le Masson et al. 2018).
[4] Creative preservation is an approach to design and innovation that uses creativity not to replace or destroy old systems. Creative innovation is an approach to design and innovation what to preserve is called into question.

metaphysical fiction of craftsmanship[5] in which the ordering of the cosmos makes intelligible the organization of materials, activities, and capacities within a life. This account can be brought into relation with Laruelle's philo-fictions: the composition and production of concepts in *generic space* (non-disciplinary space free of philosophic or a-priori assumptions) through the cloning and transformation of conceptual materials drawn from philosophy, science, technology, politics, and ordinary life. The comparison concerns the operations through which inherited materials acquire new uses and relations, leaving the differences between the metaphysical commitments of the two projects aside.[6]

On this basis, we propose to develop a counterpoint between philosophy and design science as generic science—hence, generic design science. "Generic" designates a practice in which materials from heterogeneous domains become available for composition without any contributing discipline claiming final authority over their use. Sellars's account of craftsmanship in the Timaeus becomes material for this practice through its treatment of the soul's own capacities as available for shaping and its treatment of particular crafts as implicated in the organization of living. The allegory thus provides resources for investigating how designing transforms its agents, its conceptual means, and its purposes.

This use of "design fiction" follows from this account of philo-fiction and generic design. It retains a connection with contemporary design fiction through the productive use of fictional constructions to explore possibilities, while extending that inquiry to the conceptual practices through which possibilities, designers, and ends are articulated.

It is also worthwhile to note Sellars's use of diagrams in "The Soul as Craftsman" (six in a thirteen-page paper). They are indispensable for following the prose or propositional content of the essay, especially in checking how each proposition contributes to the philo-fiction or allegory as a whole. We intend to transfigure such diagrams into FSN-TDB diagrams. The latter diagrams will be relevant to the points made about design science reasoning and will be discussed extensively in Section 4.

In Sellars's *Science and Metaphysics*, abstract entities—which for Plato are not concrete, but nevertheless real, i.e., what he calls Forms—are replaced by linguistic roles, functions, or intensional meanings shared by equivalent fragments across different languages. With Sellars's functionalist and nominalist framework in mind, his essay "The Soul as Craftsman" can be read as showing what roles these abstract "functional roles" need to be playing as stand-ins for the way Plato expresses them as the soul remembering that it knew them before birth. A more likely story is that we come to learn how to use language meaningfully in a social context that existed before our birth, even if these uses are easier to learn because of something innate. This is not to say that it would not be useful to know what these innate conceptual capabilities are, if they exist, for carrying out design activities or to appreciate Plato's metaphysical allegory of the soul as craftsman.[7]

In Plato's Timaeus (Plato 2016), the Demiurge (the Craftsman) perceives the eternal Forms and uses them as a blueprint to shape the chaotic physical world. Sellars maps this tripartite dynamic directly onto his modern theory of language and mind. In "The Soul as Craftsman," from the point of view of Sellars's *Science and Metaphysics*, one can read the Platonic soul not as a ghost in a machine, but as standing for an organizing, rule-governed capability. The "Craftsman" represents the mind operating as a normative engine. It is the mechanism that binds itself to linguistic rules, ensuring that a German speaker saying "Rot" and an English speaker saying "Red" are

[5] Sellars is also alluding back to the Republic with respect to craftsmanship but widening its purview from political space to cosmological space. He does this through his allegory of the Demiurge, who is the model of the human soul as craft person.

[6] This comparison concerns the transformation and use of conceptual materials; it leaves intact the differences between Sellars's scientific realism and Laruelle's radical immanence, including their accounts of what ultimately determines thought.

[7] A functionalist theory of linguistic meaning does not have to mean that these abstract entities or meanings cannot be innate. There are currently viable research programs that propose empirically driven hypotheses articulating the specific contents of an innate semantics. Such scholars as Anna Wierzbicka, Ray Jackendoff, Jerrold Katz and Jerry Fodor and Igor Mel'čuk all have explicitly hypothesized the precise primitives, features, or structures that make up the innate conceptual bedrock of language. While this may constrain how meaningful language functions, it does not mean that these functions cannot be determined empirically. Nor does it mean that these hypotheses underwrite Plato's theory of Forms as being necessary for design. It may mean that having concepts and the roles they play in domains of evolving knowledge is useful for keeping track of a design project.

both executing the same functional blueprint. Through the habits and wiring of the soul, abstract linguistic functions are able to "picture" or map the physical environment. The linguistic craftsman's labor results in an isomorphic relationship between our conceptual systems. By the same token, concepts are also bound to laws that underwrite material inferences. Law-like statements are essentially material rules of inference. To specify a material rule of inference involves objects, attributes, and relational structures that allow inferences about them. Socrates is a man and likes to debate, the Agora is a relational structure in which people gather and debate, and there is an Agora in Athens, where Socrates lived. So, if you are told that Socrates is in the Agora, you can infer (at least with high probability) he is in Athens and that he is probably debating. Or in a more law-like example: the attribute "soluble" has no meaning outside of this specific law-like inference. "If a solid object is placed in water, it dissolves," which allows the inferential move from "This cube is soluble." to infer: "If I drop this into that puddle, it will disappear."

2. From Conceptual Engineering to Generic and Generative Design

A growing literature treats an increasing number of philosophical issues as matters of conceptual or language engineering, with debates about how much of philosophy consists in engineering concepts for specific purposes. The idea of *generic design* can be seen against the background of the two theoretical linguistic turns, even though it is based on the work of François Laruelle (Laruelle forthcoming) and Anne-Françoise Schmid (Schmid and Hatchuel 2014), who would not place their work in this milieu. The term was introduced in "The Discrete Scaffold for Generic Design, an Interdisciplinary Craft Work for the Future" (Monarch et al. 2021), which paired Laruelle's non-standard philosophy with engineering principles to develop a formal "generic design method" that treats disciplines as raw, democratic material for building multi-scaled conceptual worlds. We first trace the connection between the analytic-pragmatist turn and conceptual engineering, because it makes visible a largely ignored connection with the structuralist and post-structuralist turn. Once that connection is visible, the step from concept engineering to concept design, and then to generic design, is a short one for both.

James I (2019 *The Technique of Thought* () shows how French thought after the linguistic turn went down this path. Drawing on Simondon's sense of technics as shaping human thought and experience, and on Leroi-Gourhan's proposal that technics and what makes us human are originarily and prosthetically coupled, James follows the development through Derrida's arche-writing and Stiegler's mnemotechnics (techniques for externalizing memory). He argues that Laruelle belongs to this milieu in transforming philosophy by cloning its conceptual materials. These materials are primarily fragments of language borrowed from different disciplines and milieux, philosophy included, and composed into philo-fictions in what Laruelle calls a generic space, which is not only non-disciplinary but suspends the fixations of ordinary language. Deleuze and Guattari's *What is Philosophy?* (1994) bears striking similarities to Laruelle's cloning of conceptual materials. It characterizes philosophy as the art of forming, inventing and fabricating concepts on a plane of immanence or consistency: a multi-dimensional diagram in which the virtual and the actual constantly influence each other, without any fixed separation between them.

In Anglo-American philosophy, the term "conceptual engineering" was introduced by Creath to characterize Carnap's technique of explication (Carnap and Quine 1990), and it has been discussed by Negarestani in his lecture "On Carnap's Aufbau." (Negarestani 2021) Carnap, according to Creath, took a methodological, formal and pragmatic stance toward explication, deliberately refining or replacing vague everyday concepts with precise, logically structured ones in the service of, but also limited to, logical reconstruction and the unification of science. Negarestani recasts conceptual engineering in a more expansive, constructive register, linking it to philosophical system-building in post-Hegelian and post-analytic approaches. Floridi's "conceptual design" (Floridi 2019) extends the coverage beyond science to informational and socio-technical domains, treating concepts as designable artifacts of formalized information; but Floridi holds that conceptual design is performed decisively by philosophy as the ultimate site of concept making, a view Laruelle challenges.

Goodman's *Ways of Worldmaking* (Goodman 1978) does not bill itself as conceptual engineering or design, but broadens its scope further, treating nominalized symbol systems as art-like constructions not limited to science, logic or philosophy, while hypothesizing basic operations of construction common to different ways of worldmaking. Worlds are not made from scratch, however, but from prior or inherited versions, which blocks worlds outside the line of descent from participating in new designs. Yet constructions of living worlds can be infused with conceptual materials from physical worlds outside the inheritance line: importations or abductions (in Peirce's sense) can explain how living worlds emerge by identifying the conditions for the emergence of life rather than non-deterministic laws governing it (Kauffman 2019). Whether social or symbolic worlds emerge in this way from bio-semiotic worlds is more controversial.

Ten years earlier, Sellars in *Science and Metaphysics* (Sellars 1968) developed a scientific realism based on a kind of nominalism that Goodman's book fundamentally challenged, even though the two had much in common: both had roots in Carnap, both regarded symbol systems as materials for constructing worlds from earlier worlds or frameworks in the same line of descent, and both contributed to analytic philosophy's shift toward a pragmatist, context-sensitive understanding of conceptual frameworks. Sellars moved from a stereoscopic vision of the manifest and scientific images to a design based on picturing, an Archimedean point approximated in the scientific image. Matter-of-factual assertions coincide "with the picture the world-cum-language *would* generate in accordance with the uniformities controlled by the semantical rules of the language" (Sellars 1968, p. 145), and "the correctness of the picture is not defined in terms of the correctness of a performance but vice versa" (ibid.). For Sellars, then, the correctness of performance answers to the correctness of the picture. We return to this asymmetry in Section 4 and in the conclusion, where we argue that the two answer to each other.

Goodman's and Sellars's views of conceptual design differ significantly from Laruelle's. Laruelle starts from a real that cannot be known but causes thought and worlds not identical to it: the "radical immanence" of lived experience, "the One-in-person," from which thought and worldmaking emerge. And while conceptual design for Goodman and Sellars involves practice, neither considered its ramifications for normative practices such as political economy, which Laruelle does, making conceptual design more generic under what he calls generic science. Generic science clones conceptual materials into a generic matrix that confers a non-hierarchical, flat ecology among them, without subsuming them into a single authoritative discourse or keeping them in fixed lines of descent, as Sellars does. Cloning is done in and across milieux: science, art, engineering, normative science and practice, and, most importantly for Laruelle, philosophy itself. In place of games, disciplines, paradigms and frameworks, Laruelle proposes philo-fictions that incubate new designs, hypotheses and implementations with no fixed overarching order. We therefore read his non-standard philosophy as a proposal to redesign conceptual design into something more generic and generative, which we call generic design. Generic design makes new concepts or rules from conceptual materials and old habits, creating new language games, political-economic orders, even whole worlds: new concept-infused practices, without being ensnared in philosophy's double transcendences, in which concept and practice are each seen as transcending the other. In Laruelle's unilateral duality, concept-infused practice is immanent lived experience: in the last instance, doing what one is generically proposing in the very act of doing it. It resembles Peirce's view of experience, thought and practice as semiosis that embodies changing norms by correcting habits, Nancy's transimmanence, in which sense is inherent in the being and becoming of the real rather than imposed from a transcendent source, and Buber's dialogical immanence.

Generative Design and GenAI

Bordas et al. (2024) ask what is "generative" in generative AI from the standpoint of C/K design theory and find two distinct levels of generativity rather than one. Starting from a canonical design task, in which design parameters and functional requirements are linked by a model, they distinguish four types: design by activation (a fixed model is activated), design by inversion (a fixed model is inverted), fixed-language design (no fixed model, but a fixed language of object dimensions and requirements), and non-fixed-language design (even the language or dimensions of the object can evolve). The four main families of GenAI models (VAEs, diffusion models, GANs and GPT-style LLMs) differ technically but, at the level of the learned model in use, all generate by

activation: they sample from a learned distribution, producing artifacts that are new but remain within the domain structured by their training data, much like Kuhn's normal science. A second, more hidden level appears in the design of the models themselves: training and optimization produce models that did not previously exist, a fixed-language generativity that combines distinct pockets of knowledge, such as statistical physics for diffusion and linguistics for attention. Bordas et al. (2024) even cite a neural weather-forecasting model as a paradigm shift away from equation-based forecasting. But the most radical form, non-fixed-language generativity, is in their words "currently accessible only to humans," and they leave open "how humans can leverage this nonfixed-language generativity to enhance the generativity of GenAI tools."

This is the question our paper takes up, with two differences of emphasis. First, we ask not how humans might help GenAI approach non-fixed-language breakthroughs, but how humans and GenAI can work together to synergize them, without assuming that GenAI achieves what is called Artificial General Intelligence or personhood, and without reducing it to a tool. Second, the distinction between fixed-language and non-fixed-language design anticipates the argument of Section 6: non-fixed-language design is a change of schema, the kind of change that creative preservation cannot measure. Our FSN-TDB supplement to C/K theory, like generic design, aims exactly at this, in which the language of design itself is open to redesign.

3. Introducing Frame Semantic Networks and TypeDB to a C/K Theorist

An LLM (Claude, Anthropic) was used under the author's direction to parse texts into TypeDB schema definitions and data insertions, to write TypeQL queries for the figures, to suggest and locate bibliographic sources, and to edit drafts. All outputs were reviewed and revised by the author, who takes full responsibility for the content.

TypeDB's current stable state as of this writing is 3.13.x (October 2026). TypeDB Studio is the official interactive visual environment for managing, modeling, and querying TypeDB databases. Its ability to visualize schema definitions and inserted data will be useful in explaining how FSN-TDB might offer a way to visualize, inspect, evaluate, remove, modify, and create new design concepts, in particular, the C and the K of the C/K Design Theory even in the Topos version. The schema definitions of TypeDB are formulated in Frame Semantic Network terms, using FrameNet as a basis. FrameNet is a computational lexicographic database that maps the meaning of words to common-sense situations based on Charles Fillmore's theory of frame semantics. Importantly, using FrameNet as a basis does not mean that all frame semantic concepts and the lexical units that trigger or instantiate them are derived from FrameNet. Most domain-specific frames have to be constructed using semantic parsing.

Frames are inscribed in TypeDB knowledge bases as TypeDB relations that correspond to framed relational structures. Examples of relational structures are situations, processes, events, and states. Relational structures are specified in terms of roles that are played by entities, or even other relational structures. Both relational structures and entities have attributes. TypeDB therefore has three root concepts—entities, relational structures (called relations in TypeDB), and attributes, all of which are designated with names that label nodes in TypeDB graphs. There are also labelled edges of two types: role edges define how an entity or relation participates in a relationship with other entities or relations, while an attribute defines a typed data value (like a string or integer that are neither entities nor relational structures) that an entity or relational structure owns or has. Relational structures specify the names of the roles they relate to that correspond to the labels placed on edges from the relational structure to what plays these roles. Entities as well as relational structures specify the name of the relational structure in which it plays a role and the name of the role it plays, and its edges use that name as a label for the edge. In schema graphs, roles are labeled on a node by specifying the label of the relational structure followed by a colon followed by the name of the role that a relational structure "relates" and that either an entity or a relational structure plays. In data instantiations of a schema, there are no role nodes. Roles are derivatives of relational structures and entities and are specified only as labels of the edges defined in the schema as "relates" for the former and "plays" for both. In this way, roles are given a different ontological status when rendering concrete goings-on, rather than

abstraction concepts. This means that roles do not determine a fourth root category alongside TypeDB's three roots—entity, relation **(**or relational_structure in the TypeDB databases we experiment with), and attribute.

Fig. 1 shows the schema we abducted, in Peirce's sense, from the C/K Topos paper of Le Masson et al. (2026) for its railroad renovation case, which Section 6 discusses in detail. With an LLM, we compiled an inventory of 68 textual fragments from that paper and classified each by the entities, relational structures, attributes, processes and agents it mentions. Fragments stating the theory's rules informed the schema; fragments describing the case became its data. The schema has four regions. The railway domain covers the artifact hierarchy (platform, ballast, sleeper, rail) and the nested support relations through which components form an assembly. The rules region represents Cat: rule sets for each component and assembly, the rule morphisms between them, and the propositions in which rules are stated. The design-process region represents the steps of C/K Topos theory as relational structures with input and output roles, the constructs they produce (C-space, Cat, presheaf, topology, sheafification, conjunction), the partitions and proposals of the concept space, the status of each component (kept, replaced, unknown, designed), and the final design decision. The evidence region links each fragment to the instance it evidences, so that every element of the data can be traced to its source. Because TypeDB relations can play roles, design steps, partitions and decisions are themselves relational structures relating other relational structures: the design process is represented with the same expressive resources as the domain it transforms. Fig. 2 (Section 6) shows the schema instantiated by the textual data (Online Resource 1).

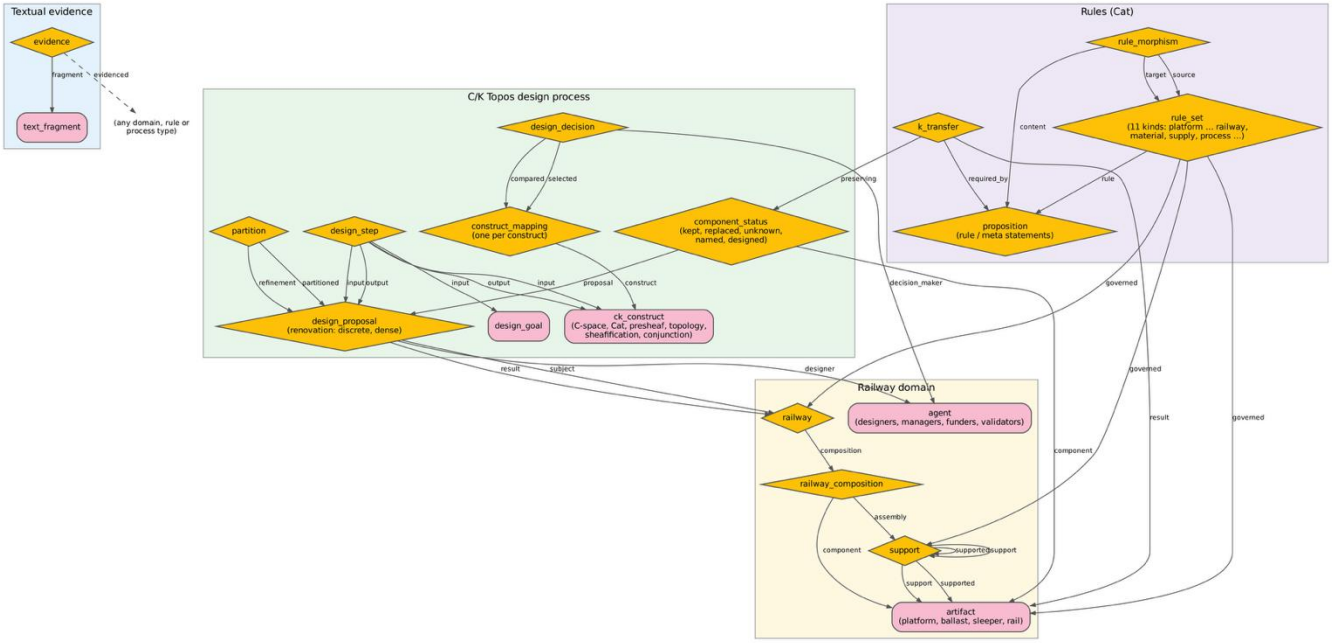


Fig. 1 Overview of the TypeDB schema abducted from the railroad renovation case of Le Masson et al. (2026). Diamonds are relation types and rounded rectangles entity types; edge labels are roles. Subtypes are listed in parentheses; attribute types and some minor roles are omitted for legibility

The rest of this section shows the C/K theorist how C and K, currently described as undecidable and true propositions, can also be formulated in a frame-based TypeDB schema: C as the kinds of concepts the schema defines (relational structures, roles, entities and attributes), and K as inserted data, such as technical documents and project-session discussions, instantiating them. One point needs to be explicit at the outset: this does not identify C/K theory with TypeDB. TypeDB supplies a computationally executable representation in which the distinction between a space of concepts and a body of knowledge can be articulated more structurally than through undecidable and true propositions alone. The intention is to supplement the propositional distinction, not to replace it.

TypeDB is well suited to this because it treats relations as first-class objects rather than as edges between entities. Since both entity and relation types can play roles and own attributes, a relation can participate in another relation, making nested and higher-order relational structures possible. TypeDB Studio provides complementary visualizations: a schema view displays types and their capabilities, such as owned attributes and played roles, while TypeQL queries produce graphs of schematized instances and their connections.

C can thus be viewed as a structured conceptual possibility space and K as the data instantiations grounding or specifying it. The first contact with data in K is triggered by lexical units in the domain of discourse; once a framed relational structure is triggered, its roles and attributes are instantiated by other textual data. This includes full texts serving as an evidential corpus: technical documents, design records, project-session discussions, research literature, requirements, observations and decisions. They are evidential in the sense that they can be used to judge whether the conceptual schema fits the data.

3.1 The Role of Semantic Parsing

Semantic parsing identifies the lexical units that evoke frames and the parts of paragraphs, sentences, clauses and phrases that fill their roles. It analyzes texts into units that are inserted automatically as TypeDB instances: entities and relational occurrences, their role-player linkages, and their attributes. Provenance attributes or relations can record the document, passage, speaker, session, date, confidence or interpretive source supporting each insertion.

What a Frame-Semantic TypeDB System Adds to C and K:

C = the kinds of framed situations, relations, participants, properties and combinations that the conceptual system makes expressible.

K = the instances of those possibilities that are presently warranted by the corpus and have been inserted into the knowledge base.

This turns the C/K distinction from one based only on the epistemic status of propositions into a distinction between typed relational possibility and evidential realization. The question is then not only truth or falsity but how well the data fits the schema, an empirical question of a different sort. The distinction remains dynamic. New data may reveal that the C-schema lacks a necessary frame, role, entity type, attribute or permissible composition; the schema is extended, and the corpus reparsed under the enlarged concept space. Conversely, changes to C may make previously unnoticed structures in K queryable. C-expansion and K-expansion thus form an iterative design process:

analysis of data → specification of concepts and their contexts → schema definition → insertion of knowledge as data → inadequacy or novelty → schema transformation → repeat.

TypeDB Studio makes this process inspectable: the schema view displays the currently articulated conceptual space, and data queries display the framed structures instantiated from the corpus, so that the design theorist can move visually between the architecture of possible knowledge (C) and its currently evidenced configurations (K). This also reinterprets undecidability. A C-element need not be only a proposition whose truth value is unknown; it may be a possible framed configuration whose realization has not yet been determined. A schema may state that a design-transformation frame has roles such as initial-design, transformed-design, constraint, designer and criterion, without asserting that any such transformation occurred. Once a passage in a design-session transcript warrants the configuration, K grounds the C-element.

3.2 Implications for C/K Operators

All four C/K operators can now be understood in terms of a schema that schematizes concepts and textual data that serves as evidence for that schematization. K → C is the schema-generative transformation prompted by the misfit of existing or new data. C → C expands the concept space by *partition* or inclusion: adding, removing or transforming properties of concepts, or introducing new concepts. K → K expands the knowledge space by

deduction, experiment and other knowledge-producing operations. C → K grounds concepts in K, legitimating them.

C/K operator	Classical interpretation	Frame-Semantic TypeDB interpretation
C → C	Expansion or partition of concepts	Schema transformation / frame elaboration or addition
K → K	Expansion of knowledge through deduction or experiment	Derivation, integration, and enrichment of instantiated frame networks
K → C	Knowledge in the form of evidential data forces conceptual change	Corpus/data patterns motivate elaborated or new frames, roles, entity types, attributes, or compositions
C → K	Concepts become decidable/validated propositions	A schema-defined possibility becomes evidentially instantiated and grounded in corpus data

C → C is then not merely "one concept generating another." It transforms the schema of possibilities itself, possibilities that must still be grounded in and validated by data. A frame Design-Modification, with roles Designer, Artifact and Modification, might be elaborated into Constraint-Driven-Design-Modification, adding Constraint, Objective and Affected-Function; or what was represented as an attribute might be re-represented as a framed relation with its own roles. Its elementary forms include introducing a new frame, specializing or generalizing a frame, adding or removing roles, changing permissible role-player types, converting an attribute into a relational structure or vice versa, allowing one relation to play a role in another, composing frames into a larger frame, and splitting a unitary frame into several. Because TypeDB relations can play roles, C → C need not merely extend a taxonomy: it can alter the relational architecture of conceptualization. This is where it is most closely tied to the generativity of non-fixed-language design, though all four operators are involved. It corresponds well to the C/K notion of an *expansive partition*: the new conceptualization introduces a distinction that could not have been enumerated from the previous conceptual structure.

K → K, in turn, produces new warranted knowledge from already instantiated knowledge without changing the schema. If several project-session statements identify differently named entities as the same component, entity resolution produces a richer K without modifying C. Besides inference, K → K includes rule-based derivation, entity and coreference resolution, merging corroborating statements, detecting contradiction, adding provenance, revising confidence, temporal ordering, calculating derived relations, integrating several documents into a common frame instance, and relating instantiated frames to one another: knowledge-network enrichment rather than mere accumulation. This yields a simple test. C → C changes what can be represented; K → K changes what is represented as currently known. If semantic parsing finds something new in a transcript and the existing schema can represent it, incorporating it is K → K; if representing it requires altering the schema, the process crosses into K → C, followed by C → C.

3.3 The New Operator Cycle

Suppose a technical-project discussion says: "The material was not selected simply because it met the thermal requirement; its selection changed what the team understood the requirement itself to be." The existing schema represents Material-Selection with roles Material, Designer and Requirement. Then:
K → K: semantic analysis integrates the statement with what is known about the material, requirement and decision.

K → C: the data reveals what the schema cannot express: a design choice can *transform the requirement against which it was supposedly evaluated*.
C → C: a new frame, Reciprocal-Requirement-Transformation, is introduced, with roles Design-Choice, Prior-Requirement, Transformed-Requirement and Design-Process; because relations can play roles, Material-Selection itself can play Design-Choice.
C → K: returning to the corpus, we ask whether the new frame is instantiated here and elsewhere and the instances the textual evidence supports.
The result is a loop, K → C → C → K → K …, which looks much more like design inquiry than ordinary database construction. In classical C/K theory, C is represented by propositions whose logical status is undecidable relative to K. The frame-semantic formulation supplements this with a structural one: C → C becomes structural invention, of new relational organizations, role structures, compositions and distinctions, and K → K becomes relational knowledge production, connecting, integrating, inferring, corroborating and revising instantiated structures. This may also supplement the C/K theorist's use of topos theory, since the question is no longer only "What propositions are true or undecidable?" but also "What structures, transformations and mappings are available in the conceptual and knowledge spaces?" FSN-TDB graphs and topos-based graphs may then provide different but complementary views, as Section 6 shows.

4. Structuring and Visualizing C/K *Generativity* with FSN-TDB

Our Frame Semantic Network TypeDB (FSN-TDB) approach supplements C/K theory by structuring the relation between conceptual and knowledge generativity at the level of schema and instantiation. C designates the generative frame-semantic schema; K designates data-grounded structures instantiated under that schema. This makes conceptual invention and evidential evaluation structurally explicit and connects them through a reciprocal process of expansion, testing and revision.

Let S_t denote the current FSN-TDB schema, with its frames, roles, entity types, attributes, permissible relational structures and constraints. C-generativity expands or reorganizes this structure, $S_t \rightarrow S_{t+1}$, adding, differentiating or recombining schema elements and so altering what the FSN-TDB can represent. Let $\Omega(S_t)$ be all valid possible instantiations licensed by S_t. Validity here means conformity to the schema; it does not by itself establish evidential support or explanatory adequacy, and $\Omega(S_t)$ introduces no structures beyond the schema and its possible instantiations: a configuration requiring a new type of frame, role, attribute or relation requires a change to S_t. The actual K_t comprises the instantiated structures supported by the available data and is therefore a subset of $\Omega(S_t)$ rather than the whole possibility space. $\Omega(S_t)$ is the space over which K-generativity operates, not K-generativity itself, which consists in instantiating, comparing, testing, reorganizing and extending data-grounded structures, and so assessing what a schema enables an inquiry to explain. In short, C-generativity expands the schema; K-generativity tests what that schema can adequately instantiate.

A frame schema S_t must not be mistaken for a passive, empty template awaiting routine retrieval (K → K). It is a content-dense hypergraph of explicit role bindings, entity constraints and typed relational dependencies. The concept space is defined as $C_t = (S_t, \Omega(S_t))$. When evidence D_t is mapped into S_t to produce data instantiations $\sigma(D_t)$, questionable role instantiations and structural misfits within K_t, or even within $\Omega(S_t)$, resist routine pattern completion. This resistance disrupts closed-world optimization and forces the designer, or an agentic AI, into an expansive partition (C → C), requiring either a reinterpretation of role semantics in K_t or a change to the schema S_t itself, generating a new instantiation space $\Omega(S_{t+1})$.

4.1 The Four Operations

K → C occurs when an anomaly, new datum or inadequacy is recognized in an instantiated knowledge structure, motivating reconsideration of the schema. The question is whether the difficulty can be met by a different instantiation of the existing schema or requires new conceptual resources; an anomaly gives grounds for investigating schema change without making it automatic. C → C expands or reorganizes the schema through S_t

→ S_{t+1}, changing the conceptual machinery available for organizing evidence; merely producing another valid instance under an unchanged schema belongs to the instantiation side. C → K makes a proposed schema change testable by instantiating it against the available data, yielding candidate structures within $\Omega(S_{t+1})$ whose evidential adequacy must be assessed before they are stabilized as K_{t+1}. K → K compares and evaluates instantiated structures, asking whether an anomaly is better addressed by reorganizing instantiations under the existing schema or by a proposed revision. Relevant considerations include anomaly reduction, explanatory coverage, fit with new evidence, what is preserved from previously successful explanations, intelligibility, desirability and quality. K-generativity thereby checks C-generativity, and unresolved anomalies prompt renewed K → C activity.

4.2 Generativity and the Scope of the Supplement

The distinction separates two complementary capacities: G_C, a schema-generating capacity, and G_K, an instantiation-and-discrimination capacity. C-generativity can produce ever more elaborate schemas without improving the organization of evidence; K-generativity confined to a fixed schema can improve fit while remaining limited by its conceptual vocabulary. Productive generativity depends on their coupling, expressed schematically rather than as a numerical identity: $G_{CK} = G_C$ constrained and selected by G_K.

This account is a proposed supplement to C/K theory, open to discussion and evaluation, not a claim that classical C/K theory should define C as schema and K as instantiation. It retains the distinction concerning the logical status of propositions as an epistemic distinction within the proposed architecture, rather than as its sole organizing principle. Its contribution is to make schema invention, evidential instantiation and their reciprocal evaluation explicit, and TypeDB implementations can operationalize it to test its capabilities.

4.3 Informational Preservation and Computational Organization

This structuring changes the theoretical role of visualization. Its contribution need not depend on introducing information that classical C/K theory cannot express: FSN-TDB may preserve the principal C/K distinctions while making explicit the relational structures through which generativity operates. Larkin and Simon (1987) distinguish informational from computational equivalence. Representations are informationally equivalent when the information in each can be inferred from the other, yet their organization may support different processes of search, recognition and inference. Diagrams can place relevant information together and expose relationships that would otherwise require additional computation, an advantage that depends on the representation and the procedures used with it rather than on graphical form alone.

For FSN-TDB, complete mutual recoverability with a propositional C/K account is a theoretical benchmark, not an established property: demonstrating it would require an explicit account of both semantics and of translation between them, and success on a finite set of tasks would not establish unrestricted equivalence. A more defensible proposal is C/K-relative informational preservation. The FSN-TDB representation should permit recovery of the distinctions and transitions relevant to a specified C/K account: the status of concepts and knowledge, the four operators, the anomalies motivating particular revisions, and the resulting changes in knowledge. Task-relative equivalence is a stronger, reciprocal claim within a bounded domain, in which both representations support adequate answers to the same specified questions. Neither claim should be confused with complete informational equivalence, and preservation does not imply computational equivalence. A propositional account may report an expansive partition; FSN-TDB can make the corresponding transformation inspectable, identifying newly introduced frames, differentiated roles, changed role-player constraints and new connections between previously separate relations, which alter what is licensed by $\Omega(S_t)$ and $\Omega(S_{t+1})$. Comparing instantiated networks can likewise reveal which structures were retained, added or reorganized around an anomaly.

4.4 Peirce and the Stronger Role of Diagrams in Reasoning

Peirce's account of diagrammatic reasoning supports a claim beyond computational convenience. In his 1902 Carnegie application, he describes reasoning as the construction of a diagram under general rules, experimentation

upon it, observation of the results, and justification that the results hold for all constructions governed by those rules (Peirce 1902/1976). A diagram then participates in the discovery and justification of consequences rather than recording conclusions already reached. Peirce did not contrast diagrams with symbols sharply: he considered even algebraic deduction diagrammatic in a broad sense, and he thought his Existential Graphs more adequately diagrammatic than equally expressive linear notations because their non-linear organization represents certain logical relations more directly. The significant distinction is therefore not words versus pictures but less versus more diagrammatically adequate representation.

Applied to FSN-TDB in design theory, this is a hypothesis about the formal representation of design reasoning: an FSN-TDB may be more diagrammatically adequate because relations in the representation correspond structurally to the relations represented. A frame such as Explanation is not first encoded as a sequence of propositions from which its structure must be reconstructed; the framed relation, its roles, participants, attributes, relations to other frames and evidential grounding are represented directly as a relational structure. Diagramming and manipulating that structure, through TypeQL queries[8], alternative role bindings, frame substitutions, comparisons of schematizations and traversals of relations, could become analogous to Peirce's experimentation upon diagrams, and schema transformation and evidence-sensitive instantiation become candidate models of generativity itself. To justify the analogy, the semantics of the structures, the admissible operations and the evidential interpretation of their results must be specified; visual inspection alone establishes neither logical validity nor empirical adequacy. Such reasoning is also different from, though complementary to, sketching, simulating and prototyping, which show what a design would be like if realized; graphical reasoning can use these stand-ins together with the other factors needed for realizing a design, including its implementation.

We therefore have two complementary arguments for visualization. Larkin and Simon ask which representation makes information easier to recover and use for a given task: representations differ in task-relative computational accessibility. Peirce asks which representation more adequately embodies the relational form of what is reasoned about, and whether operations on it can disclose consequences through its structure: representations differ in diagrammatic adequacy to relational form. The second may be more fundamental for FSN-TDB. Visualization is then not only an interface advantage but the visible manifestation of a relationally structured representation whose structure participates in practical as well as theoretical reasoning. The components of the proposal can be summarized as:

FSN-TDB = logical status + relational structure + evidential grounding + visualization.

This identifies complementary functions, not a sum of independent information. Logical status preserves the principal epistemic distinctions of C/K; relational structure specifies the organization transformed and instantiated in design; evidential grounding connects instantiated structures to the observations against which they are evaluated; and visualization makes selected structures, transformations and comparisons available for inspection and diagrammatic reasoning. If *conceptual engineering* makes concepts objects of design, FSN-TDB offers a way to investigate what is being transformed: relational structures capable of multiple propositional articulations, which become explicit objects to inspect, manipulate, instantiate, query, compare and evaluate.

Both arguments concern what a single representation makes available. A further distinction concerns what different kinds of representation can show at all. Two things can be diagrammed: what design reasoning is about, and the reasoning itself as a process. In Peirce's Existential Graphs they are joined: the graph represents what is reasoned about; the reasoning consists of permitted transformations performed on it, and a proof is the sequence of graphs those transformations produce (Peirce 1902/1976). Peirce, and commentators such as Caterina and Gangle, (2023), hold that this extends to hypothesis, or abduction (Peirce 1878b); whether it can carry the full weight of design reasoning remains, for us, an open question.

[8] TypeQL is the query language for TypeDB. Results of the queries are rendered as hypergraphs using TypeDB Studio. It can also produce both new schema definitions and data insertions.

Peirce's graphs also join discrete and continuous representation, the continuous sheet of assertion and lines of identity carrying discrete relational content, though the discrete is not represented in the detail an FSN provides. FSN-TDB graphs represent discrete structure in detail: the frames, roles and instances that design reasoning is about, and, used atypically as in Section 6, the steps, inputs, outputs and decisions of a design process together with the contents being designed. What they do not show is continuous quality: how far the data covers what the concepts require, where strain and gaps appear, and whether they are filled. The topological graphs of Section 6 show exactly this. Discrete structure and continuous quality are where the two kinds of graph complement each other.

This yields a design-science research program. Informational preservation can be examined against specified C/K distinctions and tasks; computational consequences through search effort, inferential steps, errors and query costs; and diagrammatic adequacy through how well discrete and continuous representations together track the design reasoning they diagram

Diagrammatic adequacy returns us to Sellars, for whom the correctness of the picture is prior to the correctness of the performance (Sellars 1968, p. 145). We hold instead that the two answer to each other, so that evidence must be brought to bear on both: on how well a representation fits what it represents, and on how well what is done measures up to what ought to be done. On our interpretation, the normative order of design reasoning and the natural order it aims to track are meant to correspond, each mirroring the other, in Peirce's sense of reality as what inquiry would converge on in the infinite long run (Peirce 1878a). Picturing in this sense is a regulative ideal, not something any diagram achieves now. Even if FSN-TDB is informationally conservative with respect to the principal distinctions of C/K theory, it may be representationally transformative with respect to generativity. The contribution rests not on showing that graphical networks contain information propositions cannot express, but on the hypothesis that design generativity has a relational structure that can be represented and operated upon directly.

5. From Frame-Semantic C/K Representation to Semantic-Network Graph-Directed Design Execution

The preceding sections supplemented C/K theory with FSN-TDB. C is represented not only as a set of propositions without logical status but as a structured space of possible framed configurations, and K not only as propositions with logical status but as an evidentially grounded network produced by schematizing documents, observations, project discussions, experimental results and other data under the current schema. The next step is to ask whether the same graphic structure can move beyond representing what *can be conceived* to help determine what is to be done next.

Feng et al. (2026) trace how LLMs have moved through a series of engineering paradigms: Prompt Engineering for eliciting model capabilities, Context Engineering for managing the information available to the model, Harness Engineering for connecting it to tools, memory and skills, and Loop Engineering for organizing these into "iterative cycles of planning, action, observation, verification, and adaptation." Together these produce what they call Individual Intelligence. But many real tasks require heterogeneous expertise, interdependent subtasks, parallel execution, independent verification and persistent state, which exceed what any single agent can organize. Feng et al. therefore propose *Graph Engineering*, in which graph structures organize and control three things at the level of the system: task organization (how an objective is decomposed into executable, interdependent units), agent coordination (how those units are assigned to heterogeneous agents and how their work is integrated), and runtime state management (how the evolving state of execution is recorded, kept consistent and recovered). Graphs are thereby turned "from static representations into operational mechanisms" for governing execution, the basis of what they call System Intelligence.

Feng et al. (2026) also identify what Graph Engineering lacks. Explicit graph structures do not ensure that a system's components interpret them consistently: agents may still disagree about "what constitutes task completion, sufficient evidence, valid state, or authorized action." They call for *Ontology Engineering*, a shared,

machine-interpretable model of entities, relations and constraints, covering goal formation and value alignment, shared semantics and world grounding. FSN-TDB is a case of Ontology Engineering in this sense. Its frame-semantic schema (C) is the shared model of entities, relational structures and constraints; its schematized data (K) supplies grounding in evidence, with provenance; and changes to the schema are exactly the semantic changes that, according to Feng et al., LLMs may propose but that "should undergo provenance checking, consistency validation, and impact analysis," the role K plays in checking C → C. Ontologies, they add, "cannot decide which values a system should adopt, but they can make goals and normative constraints explicit and verifiable," which is the role of the design_goal, rule_set and design_decision types in the railroad schema of Section 3.

FSN-TDB also articulates the problem Feng et al. leave most open, goal formation. In the process graphs of Section 6, the design goal enters as an input: nothing in the graph shows a process of which it is the result. Goals are formed before the graphed process begins, and forming them is itself a design activity, the one C/K theory addresses when C0 is first formulated. This paper is an instance. It was designed and implemented by a mixed team: a human designer and an AI participant, Claude, which may itself operate as several agents. Formulating the paper's goals, and building the FSN-TDB schemas, data and graphs that support them, was a process of goal formation and shared semantics carried out by human and AI participants together. In this sense we extend Feng et al.'s System Intelligence from teams of AI agents to mixed teams of human and AI participants in the design process.

5.1 The FSN-TDB as more than a world model

Execution graphs differ from knowledge graphs: a knowledge graph represents what the world or design situation is, while an execution graph governs what a design process does. In FSN-TDB, execution graphs are constructed within the same hypergraph. The schema includes operational frames that represent executable procedures and specify triggers, transitions and validation conditions; execution graphs are instances of those frames. They then become the basis for implementation: code instantiates them, entering the database as data alongside other text, together with textual data reporting how well parts of the code, and the code as a whole, perform. Validation takes both into account, the code and how well it works. This is where TypeDB Studio would need to be extended or a new application built on TypeDB, to validate proposed execution paths against the schema and against evidence of their performance. This is what FSN graph-directed execution involves, and what takes FSN-TDB beyond a knowledge graph. The progression, frame-semantic schema → schematized data → FSN graph-directed execution, covers the same territory as C → K → directed execution of C/K operations, but the third term is not a new epistemological space alongside C and K: it is an operational layer within their coupling.

With $C_t = \left(S_t, \Omega(S_t)\right)$ as in Section 4, the schema now has two parts, $S_t = S_t^{dom} \cup S_t^{op}$: domain frames and operational frames. The data remain text, but of three genres: descriptions of the design domain, code, and reports on how parts of the code, and the code as a whole, perform, $D_t = D_t^{dom} \cup D_t^{code} \cup D_t^{perf}$. Schematized under S_t, they yield $K_t = \sigma_{S_t}(D_t)$, which contains executed and evaluated code as well as knowledge of the domain. An execution graph is then the part of K that instantiates the operational frames, chosen among the possibilities the schema licenses:

$$E_t = \Phi(C_t, K_t, \Gamma_t) \subseteq \Omega\left(S_t^{op}\right)$$

where Γ_t denotes the current design goals, constraints, evaluation criteria or desired outcomes. E_t is generated from neither C nor K alone but from their coupling. C determines what kinds of states, relations, transformations, participants and outcomes are representable; K determines what is supported by evidence, including evidence about code already written and how it has performed; Γ determines which possible transformations are worth undertaking. The execution graph answers the question: what should be done next, under what conditions, by whom, using what, and how should the result be evaluated?

An execution node need not be an unstructured box labelled Research, Draft or Review. It can itself be a framed relational structure, such as Experiment (hypothesis, method, material, performer, conditions, observation, criterion) or Design_Revision (prior-design, anomaly, revision, constraint, expected-improvement). Because

TypeDB roles can be played by relations, a prior experiment, design decision or user evaluation can itself play a role in the next design action. This matters because design activities are multiadic: an action has an actor, object, input, resource, condition, purpose, constraint, expected output and evaluation criterion, and reducing it to pairwise control-flow edges loses the very semantics the FSN-TDB was built to preserve. The schema would therefore include a family of explicitly operational frames, such as requires, enables, precedes, triggers, branches-to, retries, validates and terminates, which translate semantic conditions into possible state transitions. Triggers (unresolved-anomaly, model-reconsideration), for example, would not only relate two phenomena but authorize a transition when specified conditions are met. The states and transitions of the execution graph then carry explicit domain semantics.

5.2 The C/K Operators as Executable Design Operators

The execution layer asks how each C/K operator is performed. An unresolved anomaly in K can trigger a K → C procedure that retrieves the relevant data, locates the part of the frame network responsible for the mismatch, and generates candidate frames or compositions. A C → C procedure can compare candidate schema transformations, construct new roles, recombine frames, loosen role-player restrictions or propose an expansive partition. A C → K procedure can determine which experiment, simulation, corpus search, prototype or stakeholder consultation would discriminate among the candidates. A K → K procedure can integrate the resulting observations, perform inference, compare the revised schematization with the previous one, calculate anomaly reduction, and update provenance and confidence. C/K operators thus become types of executable design transitions as well as descriptions of reasoning.

5.3 Execution Closes the Evidential Loop

Execution produces new text, not only artifacts. Executing E_t yields results R_t: the code written, together with the logs and reports of how it performed. These join the evidence, $D_{t+1} = D_t \cup R_t$, and are reschematized as $K_{t+1} = \sigma_{S_{t+1}}(D_{t+1})$, possibly revealing new anomalies that stimulate further conceptual transformation:

$C_t \leftrightarrow K_t \rightarrow E_t \rightarrow R_t \rightarrow D_{t+1} \rightarrow K_{t+1} \leftrightarrow C_{t+1}$

Feng et al. describe a comparable "closed loop between execution, experience, and evolution," in which execution outcomes refine task organization, agent coordination and state management, while validation and rollback ensure that only reliable improvements persist. The difference is where design enters. Their loop improves the organization of the system; ours can also change the concept space itself, so that the execution graph changes the knowledge structure that generated it, which can in turn modify the schema and the execution graph: a **reflexive design system**. Execution also supplies a practical check on generativity. C-generativity can produce large numbers of possible framed structures, as $\Omega(S_t)$ gives way to $\Omega(S_{t+1})$; K-generativity discriminates among them through anomaly detection, explanatory fit, new evidence, desirability and intelligibility. The execution layer adds a further constraint, not a third C/K space: can a proposed possibility be turned into a coherent course of action? A concept may be imaginable, and even compatible with existing knowledge, yet fail once its dependencies, resources, sequencing, actors, constraints and validation procedures are made explicit. We can therefore distinguish the conceivable, the knowledge-supported and the operationally realizable, and execution guards against the uncontrolled proliferation of conceptual possibilities.

6. C/K Topos Theory Beyond Creative Preservation

The previous sections have shown how classical C/K theory can be rendered as an FSN hypergraph in TypeDB. C/K theorists have recently added topos theory to address what they call creative preservation (Le Masson et al. 2026). We discuss this in some detail but then argue that the focus on creative preservation misses the opportunity for C/K Topos theory to continue to address creative innovation as well. We ask what happens when the foundations of a discipline are themselves put in question, in radical paradigm shifts or conceptual revolutions. These are usually discussed within specific disciplines, but the most interesting cases are those in which a

discipline itself is shaken to its foundations, or a new one is created, as when science itself arose in a mythical world without it. We first describe what C/K Topos adds to classical C/K theory and how it handles the railroad case. We then turn to the shifts from an Aristotelian to a Galilean-Newtonian world and from that to a relativistic and quantum one, and to the thought experiments and impossible figures that drove them. We argue that C/K Topos can be applied to these cases as well, provided preservation is no longer its measure of success: its tools (categories, topologies, sheafification, and the obstructions that block it) apply even where its criterion of preservation does not.

6.1 What C/K Topos Theory Adds to Previous Versions of C/K Theory

C/K Topos Design Theory was proposed to address a major limitation of classical C/K Theory: the inability to model "Creative Preservation"—how a designer radically innovates within a highly constrained, rule-bound tradition or systemic structure without destroying its foundational rules. Classical C/K theory was built primarily to model Schumpeterian "creative destruction" or breakthrough innovation, where the identity of objects is entirely destabilized (Schumpeter 1942; Hatchuel and Weil 2009). When applied to complex existing infrastructures (like a national railway network) or strict rulesets (like classical French culinary arts), classical C/K theory often treated the existing ruleset as a fixed, rigid constraint that stifled innovation unless completely abandoned. By injecting Grothendieck's Topos Theory (a branch of category theory that generalizes set theory and logic) (Mac Lane and Moerdijk 1992; Caramello 2017), design researchers added four critical capabilities:

Design Step 1: *Cat* (The Base Categories) – fixed constraints like immutable system rules

This change in design state maps the baseline ontology and established tradition into concepts. It defines the objects, their attributes, and their relationships (e.g., railway track gauges, station safety regulations, or classic culinary ingredient ratios).

Design Step 2: *Presheaves* (Anomaly Identification) – Anomalous concept that is locally valid, globally open

The change in design state introduces a radical new concept or identifies a systemic friction point. A presheaf $\boldsymbol{F}: \boldsymbol{C^{op} \rightarrow Set}$ assigns "sets of possible design choices" to the base objects. However, unlike a sheaf, a presheaf is not required to satisfy global gluing conditions. It addresses an anomaly as an unverified concept because while a designer can imagine this new feature locally, it currently contradicts or cannot be reconciled with the overarching system in Step 1. It is a formalized representation of K → C.

Design Step 3: *Grothendieck Topologies (J)* (Deductive Strategies & Tree of Ideas) – Partitioning strategy represented by a tree of disjunctions

This change in design state is the result of selecting a design methodology to systematically explore options. The topology *J* defines what constitutes a valid "covering" of the objects of Cat (which parts must be given and which may be deduced), and so determines how a concept is partitioned into sub-concepts within the Concept Space (C) of the design space. By choosing a specific topology, the designer chooses a strategy for how to partition or organize the concepts of a disjunction. This generates a structured tree of ideas that determines exactly how much deductive freedom, ambiguity, or strict constraint propagation the designer will tolerate while exploring variations to resolve the anomaly.

Design Step 4: *Sheafification* (Conjunction & Systemic Convergence) – Global convergence as glued conjunction

This change in design state forces the radical innovation to integrate with the base infrastructure or shows that it cannot. It is the mathematical closure of the design process. Sheafification takes the anomalous presheaf from Step 2, runs it through the topological partitioning rules of Step 3, and forces it into a Sheaf. In category theory, a sheaf guarantees that all compatible local design data can be "glued together" into a unique, coherent global section. In design terms, it forces a conjunction (C → K), establishing whether the final innovative artifact can coexist with the traditional base category without destroying it.

The Presheaf stage is key. It is the mathematical vehicle for an “unresolved innovation” or an “anomaly.” It represents data or a conceptual proposal that is locally coherent but globally mismatched with the existing base system. It bridges the gap between static base rules and dynamic design strategies. To see how these work, an example from Le Masson et al. (2026) is useful.

The case, from this C/K Topos paper, concerns one of France’s roughly 9,200 km of old local railroads. Its renovation is framed by thousands of pages of engineering rules. The designers decide to preserve the rules. Keeping every old element (platform, ballast, sleepers and rail) would leave an aging assembly that no longer meets them, and the line would close; replacing every element would cost far more than funders can bear. The designers instead keep the platform, ballast and sleepers, leave the rail open, and design a new one inspired by the long-welded rails of high-speed lines. The new rail supplies the rigidity the old substructure lacks, and the solution is judged valid and affordable.

The four-step process is activated by the design goal of renovating a given local railroad. Step 1 sets out the initial concept C0 based on the category construct, Cat, of railroad engineering rules. Step 2 forms a presheaf, an instantiation of Cat with the rail left unknown. Step 3 specifies two topologies: a topology J_discrete, under which every element must be given, and only keep-or-replace trade-offs remain, or topology J_dense, under which the rail may stay open. Step 4a sheafifies, trivially under J_discrete, where the gap remains, and by knowledge expansion under J_dense, where the long-welded rail fills it. Step 4b concludes false for the discrete branch, true for the dense one. Taking into consideration these two conclusions, the designers decide to pursue the dense branch, which ends the process. Fig. 2 shows how textual data, 68 textual fragments abducted from the paper’s text, instantiate a TypeDB schema, providing evidence for how well the textual data fits the schema.

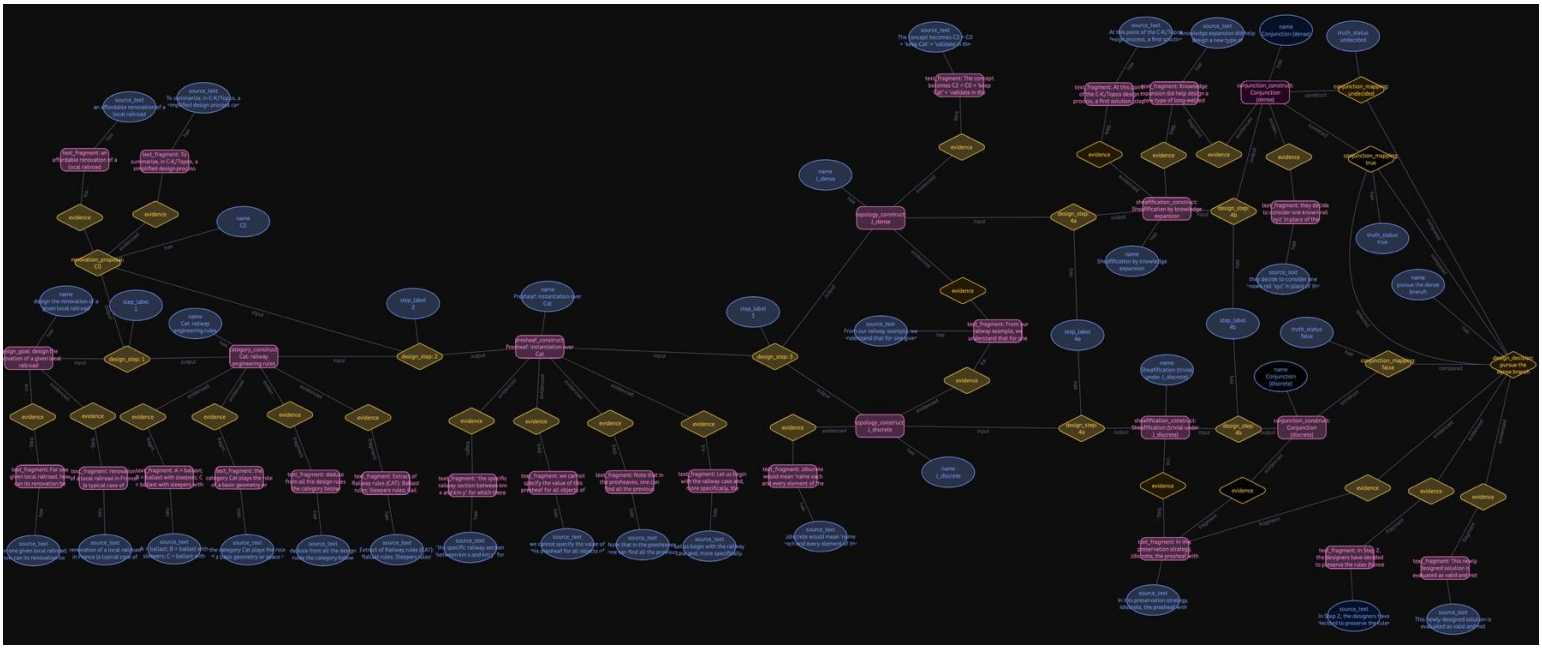


Fig. 2 Textual data providing evidence for the design goal of local railroad innovation

Fig. 3 shows the process twice, in horizontal bands shared by both panels, one band per step, with the goal below and the decision above. On the left, the FSN Hypergraph shows what each step takes in, puts out and concludes. It carries typed roles, direction, discrete outcomes, and text data with its sources: the goal statement, the rule-sets of Cat, the unmet rigidity requirement, and the reasons each option was rejected or selected. On the right, the surfaces show what each step does: the relief of the rules; the aging texture, the trough of missing rigidity and the hole of the unknown rail; the dense plateau above the discrete one; and, at the top, the teal patch of the long-welded rail filling the hole and rising to a peak, beside the pits of the rejected options. Heights are evidence counts from the data.

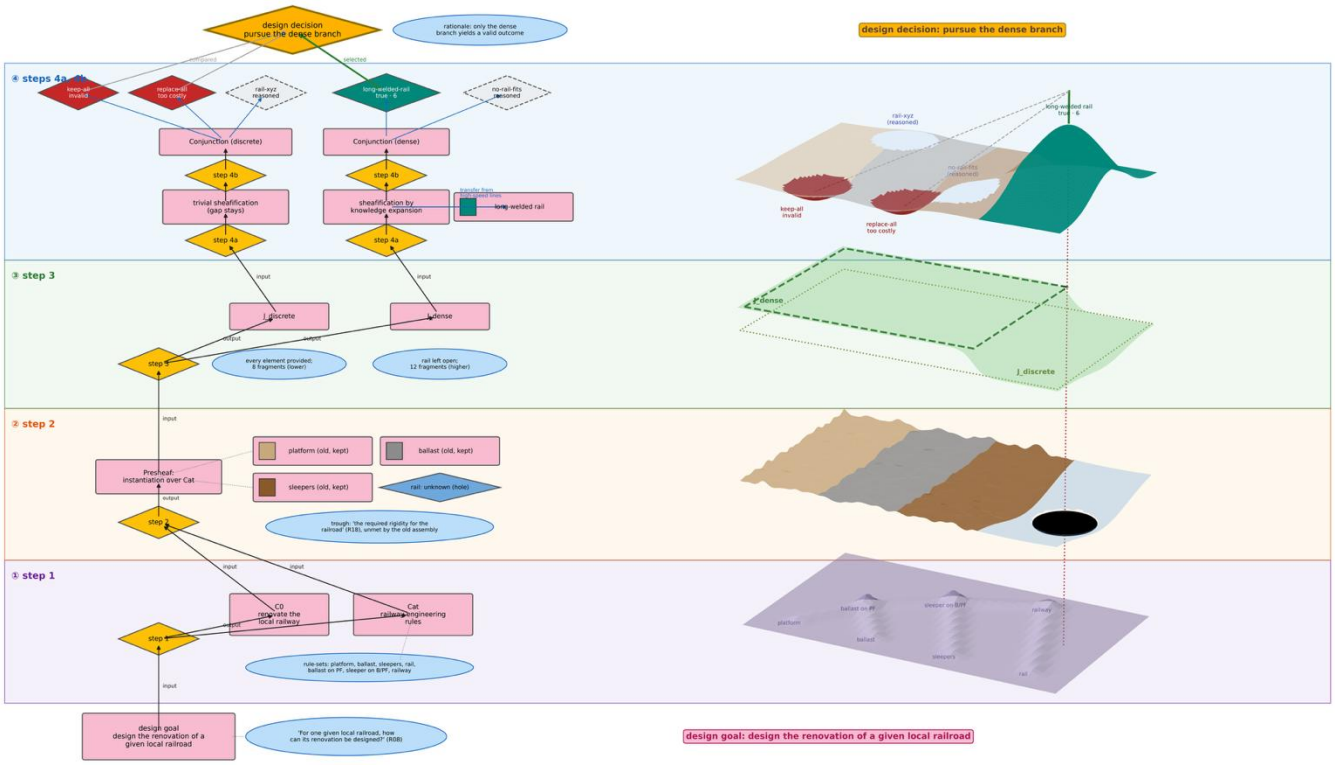


Fig. 3 Renovating a local railroad: the C/K Topos design process as an FSN Hypergraph (left) and as layered topological surfaces (right). Bands mark the four steps in both panels, from the design goal (bottom) to the design decision (top)

Each panel covers what the other cannot. The graph cannot show magnitude, continuity, strain, or how much of the space a strategy covers; the surfaces can, but they cannot show types, roles, provenance or direction, and so cannot say why a region looks as it does. Read together, the graph explains the surface, and the surface shows the consequences that the graph only names.

6.2 Applying C/K Topos Theory to Creative Innovation

We have seen how creative preservation works when Cat is fixed: the designer keeps a category of rules and varies which of its elements are given and which are left open to be designed. But what about transitions in which Cat itself is not fixed? Here we take up the much-discussed ideas of Kuhn and Feyerabend, and of those before and after them, on radical paradigm shifts or scientific revolutions (Kuhn 1970; Feyerabend 1975). These arise from anomalies, from new data not yet digested by the current paradigm, or from new intuitions, such as the thought experiments of Galileo, Einstein and Schrödinger. In such cases the paradigmatic concepts of previous and current knowledge spaces are shaken and become less fixed; at most the data is preserved, and even the data becomes subject to reinterpretation. C/K Topos does allow the rules themselves to change: "a designer can always design a new category Cat* related to Cat" (Le Masson et al. 2026). But if its success remains preservation, radical shifts fall outside that measure.

The authors' own example shows why. Special relativity, they write, "preserves Newtonian physics at low speed" (Le Masson et al. 2026). Kuhn argued that the Newtonian laws derived as the low-speed case of relativity are not Newton's laws: the derived "mass," "space" and "time" are Einsteinian terms that happen to take Newtonian values (Kuhn 1970, ch. IX). What is preserved are numbers, not concepts. It amounts to a re-instantiation of data in a new schema. Radical shifts can even change the data as well. What logical empiricists and radical behaviorists treated as data is already interpreted as it is collected. Sellars's critique of the "myth of the given" (Sellars 1956) and Feyerabend's pragmatic theory of observation (Feyerabend 1958) show that observation statements take their meaning from the theories in which they are formulated: theory is needed to formulate the data. Sellars argued for scientific realism against the instrumentalism of his day; the same tension returns today, between physicists who see speculative theory as no longer physics, thereby adopting an instrumentalist retreat of "shut up and calculate."

The transitions we have in mind are the two most profound paradigm shifts in the understanding of nature: from an Aristotelian to a Galilean-Newtonian world, and from that to the unfinished relativistic and quantum one. The shift from Aristotle's world to Galileo's and Newton's de-anthropomorphized nature. Aristotle's universe was qualitative and purposive: a stone fell because it strove to return to its natural place at the center of the Earth. Galileo and his successors replaced this teleology with mathematical law and mechanical causation, and, through thought experiments and physical experiments, overturned the Aristotelian requirement that motion needs a constant mover (Kuhn 1970; Koyré 1978).

The shift from the Galilean-Newtonian world to the relativistic and quantum one is unfinished. Einstein replaced absolute space and time with a geometric fabric that mass and energy warp, producing what we perceive as gravity; it governs the cosmic scale with great precision. At the subatomic scale, Schrödinger and his peers found that matter follows no definite trajectories: it is described by a wave function of probabilities, in superposition until a measurement yields a definite outcome. Causation becomes probabilistic, entanglement makes correlations non-local, and the observer's measurement context enters the outcome. Unlike Galileo's revolution, which replaced Aristotle's, this second shift remains an unresolved paradox.

6.3 Radical Transitions via Paradoxical Visualizations in Thought Experiments

Both transitions, including the unfinished one, have in common that preservation takes a back seat to thought experiments that expose contradictions and paradoxes inside the paradigms in transition. We sketch a few of these and how they are crucial to conceptual change (Kuhn 1977). Galileo's falling stones is a purely logical contradiction (reductio ad absurdum), not a topological globalization problem; Einstein's and Schrödinger's thought experiments are, and these call for topos theory.

Galileo's dropping stones, in the *Discourses and Mathematical Demonstrations Relating to Two New Sciences* (1638) (Galilei 1638/1974, First Day; Gendler 1998), takes Aristotle's fundamental law of motion, applies it to a

single scenario, and forces it to tear itself apart. Galileo grants Aristotle's premise that heavier objects fall faster than lighter ones, and asks what happens if a heavy stone H and a light stone L are tied together. On the one hand, the slower L should act as a brake on H, so the combined system H + L must fall slower than H alone. On the other hand, H + L is heavier than H alone, so by Aristotle's law it must fall faster. The same bundle of stones must fall both faster and slower than the heavy stone, which is impossible, so the premise that weight dictates speed is broken. The only way to remove the contradiction is to conclude that, without air resistance, all bodies fall at the same rate regardless of their weight. Nothing of Aristotle's fundamental law of motion is preserved.

Galileo's ship, in the *Dialogue Concerning the Two Chief World Systems* (1632), goes further and calls into question what seemed irrefutable data. He was answering a central anti-Copernican argument: if the Earth turned, a rock dropped from a tower would be left behind as the tower moved east and would land to the west. Rather than forcing a contradiction, Galileo uses a visual scene to expose the assumption about motion behind the argument. He asks the reader to go below deck on a large ship with butterflies, a bowl of fish and a bucket of dripping water (Galilei 1632/1967, Second Day). With the ship at rest, the fish swim in all directions, the butterflies fly freely, and the drops fall straight into the bucket. With the ship moving smoothly at constant speed, nothing changes. Everything inside the ship, air and drops included, already shares its motion: a falling drop keeps moving forward with the ship, so that to someone inside it falls straight down, while to someone on shore it traces a curve. Both are correct descriptions.

However, Galileo also designed experimental apparatus to instantiate what his thought experiments suggested. Because free fall is too fast to time, he "diluted" it with an inclined plane: a wooden beam about twelve cubits long with a straight groove lined with polished parchment, down which he rolled a hard bronze ball, timing each descent with a water clock that weighed the water flowing out during the trial. Repeating the trials about a hundred times at different inclinations, he found that the distance traveled was proportional to the square of the elapsed time (Galilei 1638/1974, Third Day): falling bodies accelerate uniformly. The conceptual change achieved by thought experiment thus led to the creation and collection of new data, guided by the concept it had produced.

Two Einstein thought experiments have similar intentions but draw on a different kind of logical paradox. Both of Einstein's thought experiments can be interpreted through the lens of cohomology, though in different ways: the train exposes the absence of a canonical global section, the elevator a non-trivial holonomy. In algebraic topology, *cohomology* (specifically first cohomology, H^1) measures the obstruction to extending locally compatible data into a single, globally consistent coordinate system or field (Ghrist and Cooperband 2025). Ghrist and Cooperband point out that visual paradoxes like the Penrose triangle (Fig. 4a) and staircase (Fig. 4b) present exactly this tension: locally coherent geometric relationships that cannot be realized globally.

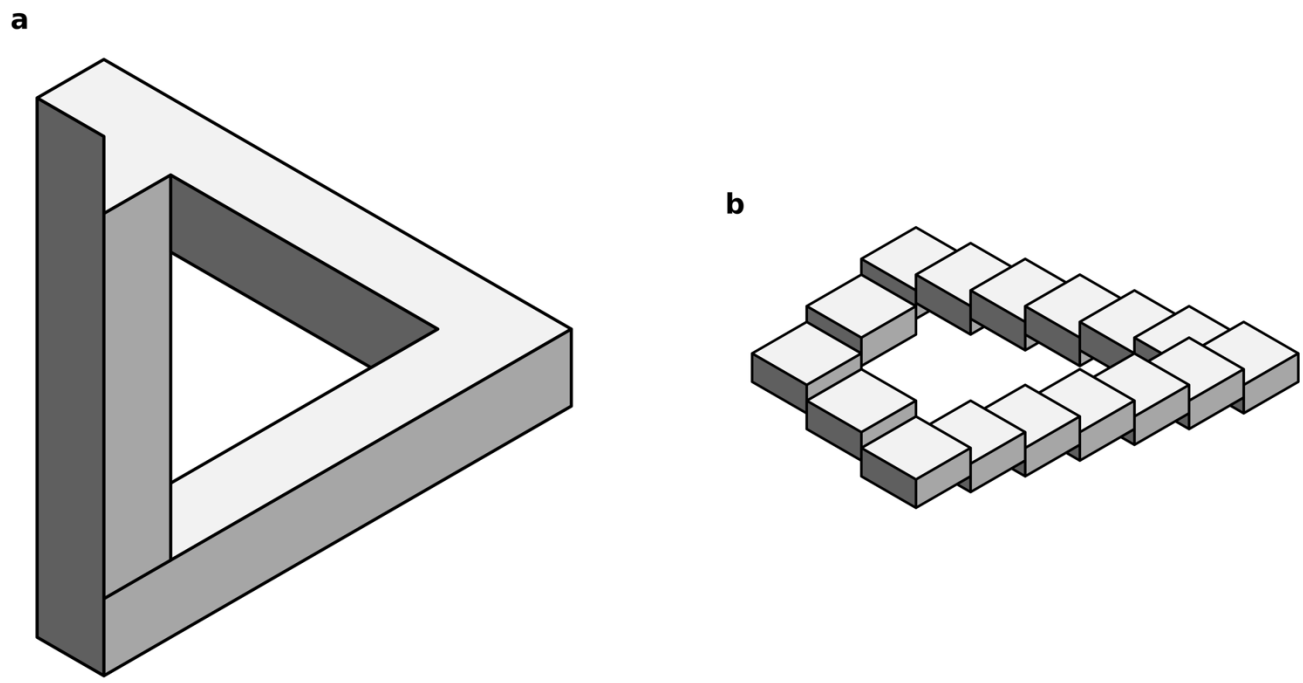


Fig. 4 a The Penrose triangle and **b** a Penrose staircase, drawn by the author after the analysis of Ghrist and Cooperband (2025). Isometric projection hides any displacement along (1, 1, 1); every joint is locally consistent, yet neither loop can close in three-dimensional space

Ghrist and Cooperband develop a mathematical framework that characterizes this phenomenon through network *torsors* and *sheaf cohomology* (Penrose 1992; Ghrist and Cooperband 2025). A network torsor formalizes relative attributes, such as height changes or orientation flips, without requiring absolute measures; it is trivial if it admits a global section, a coherent assignment everywhere. Around the loop of the Penrose staircase the height changes sum to a non-zero value, so no global section exists: the torsor is non-trivial, and the element of H^1 that measures the mismatch is the paradox's obstruction. The figures themselves show artistic and scientific intuition complementing one another. Roger Penrose and his father, Lionel Penrose, published the tribar and the endless staircase in 1958 (Penrose and Penrose 1958), and M. C. Escher elaborated the same impossibilities in his lithographs *Ascending and Descending* (1960) and *Waterfall* (1961) (Ghrist and Cooperband 2025).

Einstein used a thought experiment with a moving train and lightning strikes to establish his Special Theory of Relativity (Einstein 1905; Einstein 1920, ch. IX). Two bolts strike the front and back of the train at the same time according to an observer on the embankment; a passenger at the train's midpoint, moving toward the light from the front and away from the light from the rear, sees the front strike first. Events simultaneous for one observer are not simultaneous for another in relative motion: the *relativity of simultaneity*. Each observer can synchronize clocks with light signals, and each frame yields a time coordinate that is consistent across the whole system: in Ghrist and Cooperband's terms, each provides a global section. What fails is not gluing but uniqueness: nothing in physics singles out one frame's "now" as the true one. This is the structure of a torsor: relative relations (time differences within a frame, and the transformations between frames) are well defined, while an absolute assignment is not. Unlike the Penrose staircase, the train yields no contradiction around a loop, since passing from embankment to train and back returns the original times. What it removes is the absolute, universal "now" that Newton's concept of time required.

In the elevator thought experiment, a person in a windowless elevator accelerating upward at 9.8 m/s² in deep space feels exactly what they would feel standing still on Earth; no experiment inside the box can tell gravity from acceleration. Einstein concluded that the two are equivalent, the Equivalence Principle (Einstein 1920, ch. XX),

and from it developed General Relativity, in which gravity is not a force but the curvature of spacetime by mass and energy. Because gravity can always be "transformed away" in a small enough region, spacetime inside each elevator is flat to first approximation. But a vector carried around a loop through a sequence of these locally flat elevators returns rotated. This is holonomy, and it is measured by the curvature of spacetime (Misner, Thorne and Wheeler 1973). The "paradox" is that while *every local elevator is flat*, the *global assembly of them is curved*: here, as in the Penrose staircase, locally consistent data fail to close around a loop. Gravity, in General Relativity, is this global geometric obstruction.

As in Galileo's thought experiments, a visual intuition (the staircase's failure to close around a loop for the elevator, the absence of an absolute frame for the train) suggests a new concept that challenges what we think we know. Both kinds of thought experiment involve a paradox or contradiction, but cohomology is a specialized tool that strictly measures whether a set of overlapping local rules can stitch together into a single global whole, and whether that whole is unique. The outcomes of both challenge fundamental concepts used to describe what we thought we knew, destabilizing their fixedness and leaving open rather than preserving fundamental spacetime concepts.

Similar points can be made regarding Schrödinger's cat (Schrödinger 1935/1980). Locally, the mathematics of a radioactive atom in a superposition (decayed/not decayed) makes complete sense; so, does the macro-state of the cat (alive/dead) in isolation. The paradox emerges when the two are joined: the quantum description carries the superposition up to the cat, while the macroscopic world admits only one outcome. Penrose's objective reduction addresses exactly this kind of macroscopic superposition (Penrose 1996). A superposition of two mass distributions implies a superposition of two spacetime geometries with no canonical identification between their points, the same absence of an absolute frame that the train exposed for time. Contextuality is a separate and formally sharper point. The Kochen–Specker theorem shows that no assignment of definite values to all observables can be consistent across every measurement context (Kochen and Specker 1967). Abramsky and Brandenburger restate this as the absence of a global section of a presheaf of measurement outcomes (Abramsky and Brandenburger 2011), and a cohomology class serves as a "witness" of contextuality (Abramsky, Mansfield and Barbosa 2012), much as a cohomology class measures how far apart the ends of a Penrose triangle are.

7. Conclusion and Next Steps

We began with the linguistic turns, from the emergence of language through writing and print to the theoretical turns of the last century, and with the possibility that LLMs and GenAI will prove as world-transforming as any of them. Reading Plato through Sellars, we took design to be guided by the soul as craftsman, which in Sellars's linguistic terms is a capacity to compose conceptual materials, grounded in language use, into a balanced order. We then introduced frameworks and tools that structure conceptual materials for composition, combining FSNs and TypeDB (FSN-TDB) as a supplement to C/K theory, with visualization and graph-directed execution as resources for design reasoning, and extended C/K Topos theory beyond creative preservation to radical shifts, in which concepts and even data change.

The balance the craftsman pursues is, in our terms, a balance between the correctness of the design picture and the correctness of the design performance. Sellars made the second answer to the first. We hold that the two answer to each other. We also hold that evidence has been brought to bear on them unequally: the ontological side, how well our representations fit what is, has been emphasized at the expense of the deontological side, how well what we do measures up to what ought to be done. The C/K design process can be used to uncover this imbalance and to right it, because designing means making explicit what counts as performing correctly and bringing evidence to bear on it, as has long been done for the correctness of the picture. Tools like FSN-TDB do this structurally: reports of how a design performs enter as data alongside descriptions of the domain, and goals, rules and decisions, such as design_goal, the rule sets of Cat and design_decision. Picturing, and visualization in design, can then shift the balance toward performance, provided the picture is wedded to both the ontological and the deontological.

This is why we have focused on visual and intuitive understanding. Design reasoning is grasped through diagrams as well as through propositions, and our two kinds of diagram are complementary: FSN-TDB graphs show discrete structure, the frames, roles and instances design is about and the steps by which it proceeds; topological surfaces show continuous quality, the coverage, strain and gaps that a design process leaves or fills. Plato's khora, the receptacle that receives the forms impressed on it, conveys how we intend these surfaces to be made, etched from the schema and textured by the data. We follow Laruelle, rather than Peirce, in taking the khora not as what inquiry converges on in the long run but as the immanent, lived real now: the generic space of generic design, where materials from every discipline are composed without any one discipline claiming final authority.

We have used an LLM extensively: for semantic parsing of text into schemas, for writing queries that display graphs of slices of a schema and of the instantiated data, and for bibliographic suggestions and searching for information needed in writing the paper and for editing, catching errors and inconsistencies in the drafts, all according to the designs of the paper's named author. This paper was thus itself designed and implemented by a mixed team: a human designer and an AI participant, Claude. What allowed the two to reach common ground was, above all, a shared focus on visualization. Again and again, a graph exposed what text had hidden, as when a rearranged evidence graph revealed that an excerpt had silently lost the words "One could attempt to," turning a conditional design branch into an apparently reported one. Data are interpreted as they are collected, and with GenAI more than ever. We ask that the paper be judged by the validity of its claims rather than dismissed for its provenance. Extending these diagrams to radical shifts, for instance drawing Einstein's elevator as a change of schema alongside its holonomy, is a next step, and we expect synergy between human and AI participants to carry out such a task. We believe that such synergy is itself a radical shift in the making, one that evidences a fit between its own concept and instance.

That fit is what generativity must achieve. A measure of generativity cannot measure C alone, the depth and breadth of schema change; it must combine C and K, schema and instantiation, because correctness requires both. Two developments would make this practical, both requiring an extension of TypeDB Studio or a new application built on TypeDB: generating the topological surfaces directly from the results of TypeDB queries, so that what they show is computed from the data rather than drawn; and validating proposed execution paths against the schema and against evidence of their performance.

Statements and Declarations

Funding No funding was received for conducting this study.
Competing interests The author has no competing interests to declare.
Data availability The TypeDB schema, data and queries supporting Figs. 1–3, and the inventory of 68 textual fragments from which they were compiled, are provided in Online Resource 1. The inventory and TypeQL files were compiled by LLM-assisted annotation of the source text and reviewed by the author.
Author contributions The author conceived the study, performed the analysis, and wrote the manuscript.

References

S. Abramsky and A. Brandenburger, "The sheaf-theoretic structure of non-locality and contextuality," *New J. Phys.*, vol. 13, 113036, 2011.

S. Abramsky, S. Mansfield, and R. S. Barbosa, "The cohomology of non-locality and contextuality," in *Proc. 8th Int. Workshop Quantum Phys. Logic (QPL 2011)*, EPTCS vol. 95, 2012, pp. 1–14.

Bordas A, Le Masson P, Thomas M, Weil B (2024) What is generative in generative artificial intelligence? A design-based perspective. Res Eng Des 35:427–443. https://doi.org/10.1007/s00163-024-00441-x

O. Caramello, *Theories, Sites, Toposes*. Oxford, U.K.: Oxford Univ. Press, 2017.

Carnap, Rudolf, and Willard Van Orman Quine. Dear Carnap, Dear Van: The Quine-Carnap Correspondence and Related Work. Edited with an Introduction by Richard Creath, University of California Press, 1990.

Caterina, Gianluca, and Rocco Gangle. “Part VI: Diagrams, Visual Models, and Abduction” in *Handbook of Abductive Cognition*, edited by Lorenzo Magnani, Springer Nature Switzerland, 2023, pp. 641–787, doi:10.1007/978-3-031-10135-9_85.

Deleuze, Gilles, and Félix Guattari. What Is Philosophy? Translated by Hugh Tomlinson and Graham Burchell, Columbia University Press, 1994.

Derrida, J. (1997). *Of grammatology* (G. C. Spivak, Trans.; Corrected ed.). Johns Hopkins University Press. (Original work published 1967)

Dunbar, Robin I. M. *Grooming, Gossip and the Evolution of Language*. Harvard University Press, 1996

A. Einstein, “Zur Elektrodynamik bewegter Körper,” *Ann. Phys.*, vol. 17, pp. 891–921, 1905.

A. Einstein, *Relativity: The Special and the General Theory*, R. W. Lawson, Trans. London, U.K.: Methuen, 1920.

Eisenstein, Elizabeth L. *The Printing Press as an Agent of Change: Communications and Cultural Transformations in Early Modern Europe*. Cambridge University Press, 1979.

Feng, Yuyuan, et al. “Graph Engineering in the Era of LLM Agents: From Individual Intelligence to System Intelligence.” *arXiv*, 21 Aug. 2026, arxiv.org/abs/2608.21156.

P. K. Feyerabend, “An attempt at a realistic interpretation of experience,” *Proc. Aristotelian Soc.*, vol. 58, pp. 143–170, 1958.

P. K. Feyerabend, *Against Method*. London, U.K.: New Left Books, 1975.

Fillmore, Charles J. (1982). Frame Semantics. In Linguistic Society of Korea (ed.), Linguistics in the Morning Calm, 111–137. Seoul: Hanshin Publishing Co.

Floridi, Luciano. The Logic of Information: A Theory of Philosophy as Conceptual Design. Oxford University Press, 2019.

FrameNet. International Computer Science Institute, 2026, framenet.icsi.berkeley.edu/.

G. Galilei, *Dialogue Concerning the Two Chief World Systems* (1632), S. Drake, Trans., 2nd ed. Berkeley, CA, USA: Univ. of California Press, 1967.

G. Galilei, *Two New Sciences* (1638), S. Drake, Trans. Madison, WI, USA: Univ. of Wisconsin Press, 1974.

T. S. Gendler, “Galileo and the indispensability of scientific thought experiment,” *Br. J. Philos. Sci.*, vol. 49, no. 3, pp. 397–424, 1998.

R. Ghrist and Z. Cooperband, “Obstructions to reality: Torsors & visual paradox,” arXiv:2507.01226 [math.AT], Jul. 2025.

Goodman, Nelson. Ways of Worldmaking. Hackett Pub. Co., 1978.

Harari, Yuval Noah. Sapiens: A Brief History of Humankind. Harper, 2015.

A. Hatchuel and B. Weil, “C-K design theory: An advanced formulation,” *Res. Eng. Des.*, vol. 19, no. 4, pp. 181–192, 2009.

Ian James, *The Technique of Thought: Nancy, Laruelle, Malabou, and Stiegler after Naturalism,* Published by: University of Minnesota Press, 2019.

Kauffman, Stuart A. *A World Beyond Physics: The Emergence and Evolution of Life*. Oxford University Press, 2019.

S. Kochen and E. P. Specker, “The problem of hidden variables in quantum mechanics,” *J. Math. Mech.*, vol. 17, no. 1, pp. 59–87, 1967.

A. Koyré, *Galileo Studies* (1939), J. Mepham, Trans. Hassocks, U.K.: Harvester Press, 1978.

T. S. Kuhn, *The Structure of Scientific Revolutions*, 2nd ed. Chicago, IL, USA: Univ. of Chicago Press, 1970.

T. S. Kuhn, “A function for thought experiments,” in *The Essential Tension*. Chicago, IL, USA: Univ. of Chicago Press, 1977, pp. 240–265.

Larkin, J. H., & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. Cognitive Science, 11(1), 65–99. https://doi.org/10.1111/j.1551-6708.1987.tb00863.x

Laruelle, François. *Non-standard Philosophy: Generic, Quantum, Philo-Fiction*. Translated by Anthony Paul Smith, Columbia University Press, 2027 (forthcoming), see English translation of “From Fiction to Philo-Fiction and Science-Phiction,” translated by Jeremy R. Smith, en-demic website https://endemictheory.wordpress.com/2021/08/29/translation-of-francois-laruelle-from-fiction-to-philo-fiction-and-science-phiction-selections-from-philosophie-non-standard-2010/

Le Masson, Pascal, Armand Hatchuel, and Benoît Weil. “C-K Theory: Modelling Creative Thinking and Its Impact on Research.” Creativity, Design Thinking and Interdisciplinarity, edited by Frédéric Darbellay, Zoé Moody, and Todd Lubart, Springer, 2018, pp. 169-83. SpringerLink, doi:10.1007/978-981-10-7524-7_11

Le Masson, P., Carvajal-Perez, D., Harlé, H., Hatchuel, A., Jibet, N., Taupin, L., & Weil, B. (2026). Design creativity for transitions: C-K/Topos, an advanced design theory for creative preservation. *Design Science*, *12*, e6. doi.org

Lévi-Strauss, Claude. *Tristes Tropiques*. Translated by John and Doreen Weightman, Atheneum, 1974, Chapter 28: “A Writing Lesson,” pp. 291–311.

Low, Felicia M., Peter D. Gluckman, and Mark A. Hanson. “Niche Modification, Human Cultural Evolution and the Anthropocene.” *Trends in Ecology & Evolution*, vol. 34, no. 10, 2019, pp. 883–885

S. Mac Lane and I. Moerdijk, *Sheaves in Geometry and Logic: A First Introduction to Topos Theory*. New York, NY, USA: Springer, 1992.

McLuhan, Marshall. *The Gutenberg Galaxy*: The Making of Typographic Man. University of Toronto Press, 1962, Section: “The Phonetic Alphabet: An Invisible Technology That Alters the Human Sensorium,” pp. 18–25.

McLuhan, Marshall. *Understanding Media: The Extensions of Man*. New York: McGraw-Hill, 1964.

C. W. Misner, K. S. Thorne, and J. A. Wheeler, *Gravitation*. San Francisco, CA, USA: W. H. Freeman, 1973.

Monarch, I., Subrahmanian, E., Schmid, AF., Mambrini-Doudet, M. (2021). The Discrete Scaffold for Generic Design, an Interdisciplinary Craft Work for the Future. In: Pirtle, Z., Tomblin, D., Madhavan, G. (eds) Engineering and Philosophy. Philosophy of Engineering and Technology, vol. 37. Springer, Cham. https://doi.org/10.1007/978-3-030-70099-7_18.

Negarestani, Reza. “Reza Negarestani On Carnap’s Aufbau.” *YouTube*, uploaded by The New Centre for Research & Practice, 13 Aug. 2021, www.youtube.com/watch?v=Im3ZH-S2sPg.

Ong, Walter J. Orality and Literacy: The Technologizing of the Word. Routledge, 2012.

Pagel, Mark. “How Language Transformed Humanity.” TED, July 2011, https://www.ted.com/talks/mark_pagel_how_language_transformed_humanity. Accessed 4 Oct. 2026

Peirce, C. S. “How to make our ideas clear,” *Popular Science Monthly*, vol. 12, pp. 286–302, Jan. 1878.

Peirce, C. S. “Deduction, induction, and hypothesis,” *Popular Science Monthly*, vol. 13, pp. 470–482, Aug. 1878.

Peirce, C. S. (1902/1976). Carnegie application (L75). In C. Eisele (Ed.), The new elements of mathematics by Charles S. Peirce (Vol. 4, pp. 13–73). Mouton.

L. S. Penrose and R. Penrose, “Impossible objects: A special type of visual illusion,” *Br. J. Psychol.*, vol. 49, no. 1, pp. 31–33, 1958.

R. Penrose, “On the cohomology of impossible figures,” *Leonardo*, vol. 25, no. 3–4, pp. 245–247, 1992.

R. Penrose, “On gravity’s role in quantum state reduction,” *Gen. Relativ. Gravit.*, vol. 28, no. 5, pp. 581–600, 1996.

Pinker S (2010) The cognitive niche: coevolution of intelligence, sociality, and language. Proc Natl Acad Sci USA 107(Suppl 2):8993–8999.

Plato. *Timaeus*. Translated by Peter Kalkavage, 2nd ed., Hackett Publishing Company, 2016.

Schmid, Anne-Françoise, and Armand Hatchuel. "On Generic Epistemology." Angelaki, vol. 19, no. 2, 2014, pp. 131–144.

E. Schrödinger, "The present situation in quantum mechanics," trans. J. D. Trimmer, *Proceedings of the American Philosophical Society* 124(5): 323–338, 1980.

J. A. Schumpeter, *Capitalism, Socialism and Democracy*. New York, NY, USA: Harper & Brothers, 1942.

Sellars, Wilfrid. (1956). "Empiricism and the Philosophy of Mind." In The Foundations of Science and the Concepts of Psychology and Psychoanalysis, edited by Herbert Feigl and Michael Scriven, Vol. 1 of Minnesota Studies in the Philosophy of Science, pp. 253–329. Minneapolis: University of Minnesota Press.

Sellars, Wilfrid. "The Soul as Craftsman." In Philosophical Perspectives, 5–22. Atascadero, CA: Ridgeview Publishing Company, 1967. [1]

Sellars, Wilfrid. *Science and Metaphysics: Variations on Kantian Themes*. Routledge & Kegan Paul, 1968.

Stiegler, Bernard. *Technics and Time, 1: The Fault of Epimetheus*. Translated by Richard Beardsworth and George Collins, Stanford University Press, 1998.

Stiegler, Bernard. *For a New Critique of Political Economy*. Translated by Daniel Ross, Polity, 2010.

TypeDB Ltd. TypeDB Studio, 2026, GitHub, github.com.

TypeDB Ltd, and Haikal Pribadi. TypeDB, version 3.13, 2026, https://typedb.com.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- ESM1.zip