Point

Self-Referential Structure

Larsen James Close

March 2026

Abstract

Category theory is the fixed point of its own meta-rule: G_{Cat} is the least fixed point of the operation M that sends any compositional grammar to the grammar of structure-preserving maps between its models. A single adjunction — tensor–internal-hom — underlies every result: the probe of *Elements*, the meta-rule of *Grammar*, the reflexive object of *Computation*, the self-enrichment of **Cat**.

M generates the dimensional tower, adding one structural level per step; at the colimit, exact self-similarity emerges ($D = 1$) and dimension stabilizes: the fixed-point object's elements biject with its endomorphisms — self-indexed computational universality. The Church–Turing threshold follows from dimensional closure, not from additional structure. Five lines of evidence, each arising from a different mathematical domain, converge on $F(X) \cong X$; the Yoneda lemma certifies the construction is faithful; the same fixed-point structure reappears in epistemology. The core fixed-point architecture and computational universality are machine-verified in Lean 4; the specific EAT instantiation awaits Gabriel-Ulmer duality in Mathlib.

Concrete

Maximizing abstraction and generalization eventually bottoms out in radically coherent structure and relationships. Recognition here is substrate-agnostic — its self-similarity is exact, not approximate.

Computation is the only other domain where self-similarity is exact in this way. We conjectured that being maximally fundamental, the one must be an enabling condition of the other. Formalized proofs constructed from category theory showed this was the case: the reflexive fixed point of the structure's own endofunctor is a model of universal computation. This actualized the prior abstract intuition into Category Theory specifically — the framework where "what are the structure-preserving maps?" is a precise question with a determinate answer.

We then noticed the limits of Category Theory in relation to the maximally general and abstract, and endeavored with it to do justice to the intuition. The core chain is machine-verified.

Contents

Map

How to Read This

Larsen James Close

The Idea

Category theory is what happens when you take the most basic move in mathematics — noticing that two different things have the same structure — and study that move itself.

You already know how to do this. When you learned that addition on whole numbers and concatenation of strings both satisfy associativity, you recognized the same pattern in two different settings. Category theory is the discipline of making that recognition precise: what exactly is preserved when you say "these are structurally the same"?

This document is a map. The companion document, *Elements*, is the territory. Read this first if you want to know where the construction is going and why it was built the way it was. Read the *Elements* if you want to see the construction itself.

The construction reveals four levels of structure, each visible only when the ambient category provides it:

1. **Dimension.** An endofunctor and an initial object. The chain, its grading, tower initiality.
2. **Convergence.** The colimit exists and the endofunctor preserves it. Lambek's lemma. Dimension stabilizes.
3. **Closure.** The endofunctor is the internal hom. Containerization. Self-application. The fixed-point combinator.
4. **Computation.** The object indexes its own endomorphisms. Identity modulation. The untyped lambda calculus. Universal computation.

Each level is not added to the previous — it is what the previous level becomes when the ambient category makes it visible. The papers fill this skeleton in developmental order; the logical order is the reverse of discovery.

The Dimensional Ladder

Mathematics has a natural vertical dimension: the level of abstraction at which structure becomes visible.

Level 0: Things. Points. Elements. Objects. No relationships visible — just existence. A set, considered only as a bag of elements, lives here.

Level 1: Maps between things. Functions. Arrows. Now you can see *how things relate*. A function $f : A \to B$ is a relationship between two sets. A homomorphism is a relationship between two groups. A continuous map is a relationship between two spaces. The objects differ; the relational structure is the same.

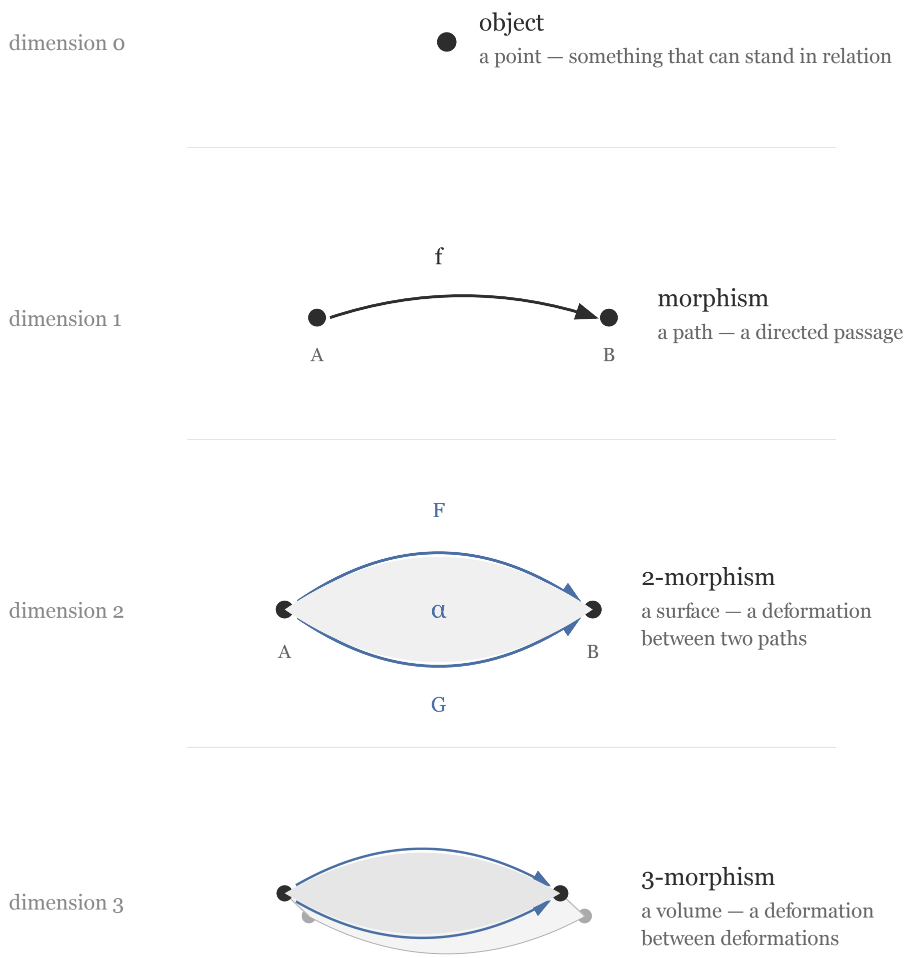
This is where category theory begins. A category is a collection of things (objects) and maps between them (morphisms) that compose associatively with identities. That is the entire definition. It captures Level 1 structure across all of mathematics simultaneously.

Level 2: Maps between maps. Given two functions $f, g : A \to B$, you can ask: is there a systematic way to transform f into g? For plain functions between sets this question is uninteresting. But for functors between categories — structure-preserving maps between entire relational systems — it is the question of *natural transformations*, and it opens a new dimension of structure invisible from Level 1.

Level 3: Maps between maps between maps. Given two natural transformations between the same pair of functors, is there a systematic deformation of one into the other that respects all the structure? These are *modifications*. Like every level before, they compose, have identities, and satisfy associativity. The compositional structure at dimension 3 is identical to dimensions 0, 1, and 2 — same axioms, no additions.

Level n: Maps between $(n{-}1)$-maps. The pattern continues. At each level, the question "what are the structure-preserving maps between the things at this level?" generates the next level. The tower is in principle infinite. The first four levels — objects, morphisms, natural transformations, modifications — exhibit identical compositional structure, and the fixed-point property ($M(G_{\text{Cat}}) \cong G_{\text{Cat}}$ — the meta-rule returns the same grammar) guarantees the pattern is exact at every subsequent level.

The *Elements* document constructs this ladder directly: Act I builds Level 1 (categories), Act II builds Levels 2 through 4 (functors, natural transformations, and modifications), and Act III proves the ladder is self-certifying.



Why Generalization Works Here

In most mathematical subjects, the general theory is harder than the examples. Groups are harder than the integers. Topology is harder than $\mathbb{R}^n$. You learn the examples to build intuition for the general case.

Category theory inverts this. The general pattern is often *simpler* than any particular example, because the examples carry domain-specific detail that the pattern strips away. Composition of functions, composition of homomorphisms, and composition of continuous maps all satisfy the same laws — but each comes wrapped in the particular details of sets, groups, and spaces respectively. The categorical abstraction removes the wrapping and shows you the compositional skeleton directly.

This means the right pedagogical direction is: start with the skeleton, then see the examples as instances.

The Examples That Already Live Inside

Every domain of mathematics you already know contains categorical structure. Recognizing it requires no new definitions — only the observation that the same pattern appears in each case.

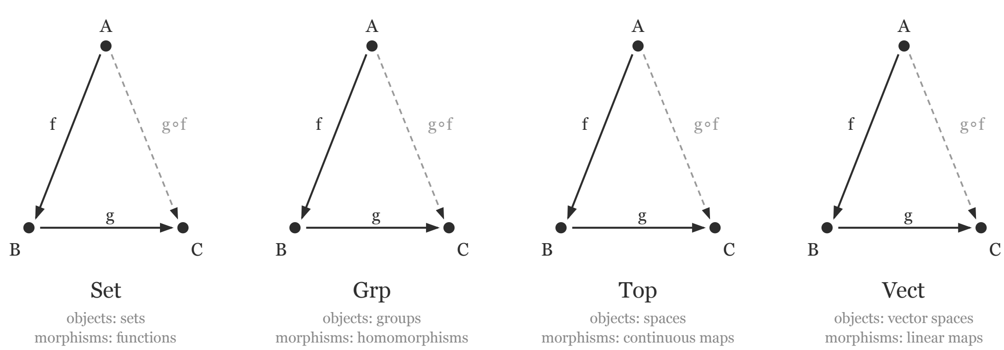
Sets and functions. Objects are sets. Morphisms are functions. Composition is function composition. This is the category **Set** — the most familiar instance.

Groups and homomorphisms. Objects are groups. Morphisms are group homomorphisms (maps that preserve the group operation). Composition is composition of homomorphisms. This is **Grp**.

Topological spaces and continuous maps. Objects are topological spaces. Morphisms are continuous maps. This is **Top**.

Vector spaces and linear maps. Objects are vector spaces over a field k. Morphisms are linear transformations. This is $\mathbf{Vect}_k$.

In each case: the objects differ, the morphisms differ, but the compositional structure is identical. Morphisms compose associatively. Each object has an identity morphism. That is the categorical pattern, and it is the same pattern in every case.



Degenerate Cases Clarify the General

The most illuminating examples are not the richest ones but the *thinnest* — categories with so little structure that the categorical pattern is visible in isolation.

A poset is a thin category. A partially ordered set $(P, \leq)$ is a category where:

- Objects are the elements of P
- There is at most one morphism $a \to b$, which exists exactly when $a \leq b$
- Composition says: if $a \leq b$ and $b \leq c$, then $a \leq c$ (transitivity)
- Identity says: $a \leq a$ (reflexivity)

The categorical axioms *are* the order axioms. A poset is a category with at most one arrow between any two objects.

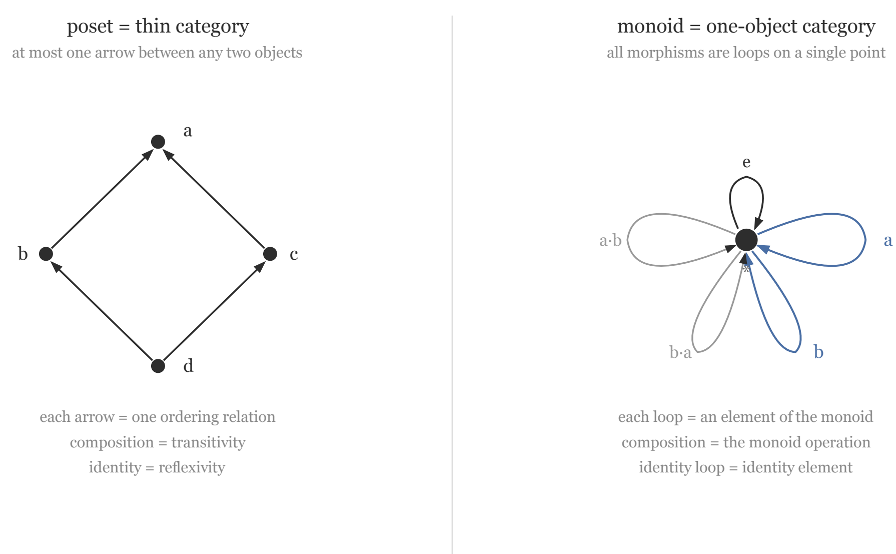
A monoid is a one-object category. A monoid $(M, \cdot, e)$ — a set with an associative binary operation and identity element — is a category with one object $*$ where:

- Morphisms $* \to *$ are elements of M
- Composition is the monoid operation $\cdot$
- The identity morphism is the identity element e

The categorical axioms *are* the monoid axioms. A monoid is a category with exactly one object.

A group is a one-object category where every morphism is invertible. A groupoid is what you get when you allow multiple objects.

These degenerate cases reveal the skeleton. A category is simultaneously a generalization of partial orders (allow multiple arrows between objects) and a generalization of monoids (allow multiple objects). The two simplest algebraic structures — order and multiplication — are both special cases of the same thing.



The Visual Bandwidth Argument

Human visual processing resolves structure, symmetry, and pattern in parallel. Symbolic processing is serial. Category theory is fundamentally about structure and pattern.

This means diagrams are not illustrations of the mathematics — they are a format matched to the cognitive channel that processes this content most efficiently. The *Elements* document is built on this principle: each proposition is a diagram with minimal annotation. The prose names what the reader sees. The visual cortex does the actual work.

The key diagram types:

Commutative diagrams. A diagram of objects and arrows where every path between the same two objects yields the same composite morphism. Commutativity of a diagram is the visual form of an equation — but it shows the structure rather than just stating the relationship.

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[dr, "g \circ f"'] & B \arrow[d, "g"] \\ & C \end{tikzcd}$$

The triangle commutes: the path $A \to B \to C$ equals the path $A \to C$. This is composition — visible, not stated.

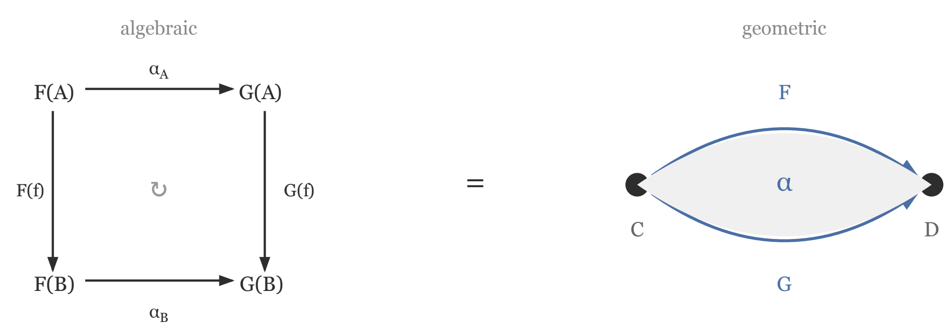
Naturality squares. The defining diagram of a natural transformation. Two functors applied to a morphism, connected by the transformation's components.

$$\begin{tikzcd} F(A) \arrow[r, "\alpha_A"] \arrow[d, "F(f)"'] & G(A) \arrow[d, "G(f)"] \\ F(B) \arrow[r, "\alpha_B"'] & G(B) \end{tikzcd}$$

The square commutes: the transformation respects the structure. This is the content of "natural" — visible as a geometric constraint.

The same diagram across domains. The critical moment in CT pedagogy: seeing that the naturality square has the same shape whether the functors map between groups, spaces, or abstract categories. The diagram is domain-independent. The visual makes this immediate in a way that symbolic notation does not.

A natural transformation has two equivalent representations — the algebraic naturality square and the geometric surface between two paths:



The Self-Certification

The *Elements* document is organized around a specific structural observation: the theory of categories proves at its climax (the Yoneda lemma) that representing an object by the pattern of arrows pointing at it — its relational profile — loses no information. The Yoneda embedding is full and faithful.

This is not a routine mathematical result. It means the representational strategy at the heart of diagrammatic reasoning — identifying objects by their morphisms — is certified as exact by the theory itself. Most mathematical representations are approximations or projections that lose some information. Yoneda says this one does not. The certification is of the mathematical content, not the visual medium as such — but the visual medium works precisely because it makes that content, objects and their arrows, directly visible.

The reading experience is designed around this: Yoneda is announced at the start as the destination, the construction builds toward it, and when it arrives it should feel not like surprise but like recognition — the representational strategy was the theorem all along.

Where the Construction Sits

The *Elements* document is self-contained. A reader who goes through it and nothing else has a complete constructive derivation of category theory's core from minimal primitives.

For readers who encounter this in the context of the larger project: the derivation chain from the primitive to geometric emergence — recognition, mutual determination, reflexive stability, computation, geometry — is itself a categorical structure. Each step preserves what preceded it. The preservation is what functors track. The derivation chain is a diagram in a category whose objects are levels of the derivation and whose morphisms are the structure-preserving translations between them. The tool and the application are the same type of thing. A reader who has internalized the *Elements* will recognize this without being told.

What Is New

The series uses standard results as scaffolding and novel claims as load-bearing structure. To help the reader evaluate the claims critically:

Established (standard results, recalled for context)**:** Yoneda lemma, Lambek's lemma (Lambek 1968a), Scott's D_∞ construction (Scott 1976), the Lawvere-Linton correspondence (Lawvere 1963; Linton 1966), the Boardman-Vogt tensor for Lawvere theories (Boardman and Vogt 1973; Hyland and Power 2007), the Curry-Howard-Lambek correspondence (Lambek 1968b, 1969; Lambek and Scott 1986), the Adámek fixed-point theorem (Adámek 1974). These are established mathematics deployed in a new configuration.

Established and formalized (verified in the paired Lean 4 project, 42 files, 0 sorry, 0 custom axioms)**:**

- *Layer 1 — Categorical substrate:* Adámek's initial algebra theorem (Adámek 1974) (original formalization, not yet in Mathlib); Lambek's lemma (via Mathlib); right adjoint uniqueness (via Mathlib); Adámek-Rosický Theorem 2.23 (Adámek and Rosický 1994) (right adjoint accessibility — original formalization); the substrate-independent fixed point (existence and uniqueness in any monoidal closed, locally finitely presentable category where the tensor preserves finite presentability).

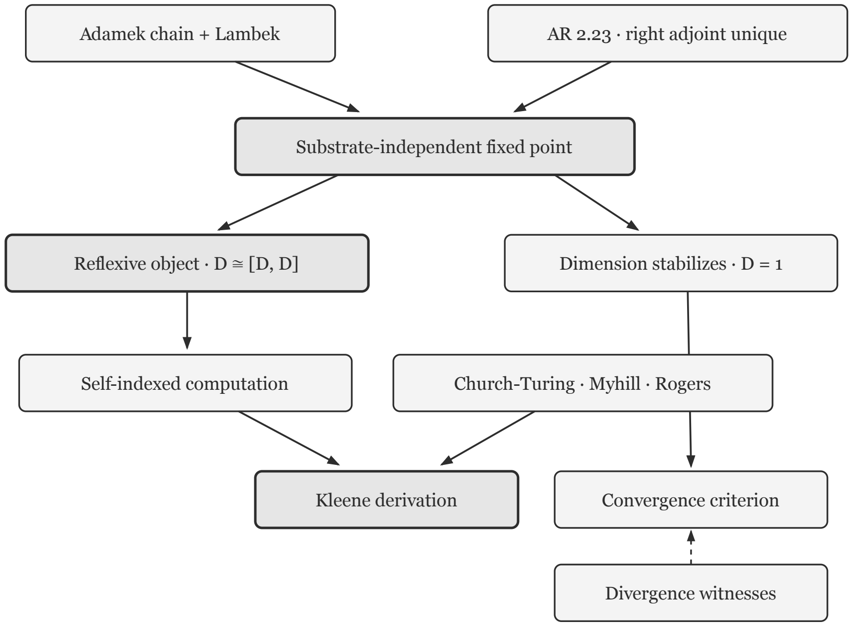
- *Layer 2 — The main result (fixed point IS computation):* The reflexive object $D \cong [D, D]$, self-application map, Y combinator, containerization, identity modulation, and the untyped lambda calculus model from the Lambek isomorphism — derived without any $\mathbb{N}$-indexing. Naming equivalence (abstract fixed-point property), universal evaluator, self-indexed Kleene recursion.
- *Layer 3 — $\mathbb{N}$-indexed bridge:* Church-Turing characterization theorem (the abstract equivalence result); Effective Myhill Isomorphism Theorem (computable bijection from computable injections); weak Rogers isomorphism (computable translations between any two acceptable numberings); Kleene's recursion theorem for abstract computation models; strong Rogers isomorphism (computable bijection between any two acceptable numberings).
- *Layer 4 — Divergence and convergence witnesses:* FinSet divergence (dimension grows without bound); thin-category triviality (only trivial reflexive objects); convergence criterion (reflexive object exists iff the colimit chain converges).
- *Cross-cutting — Dimensionality:* Truncation-level dimension, increment by one, stabilization at the fixed point; tower initiality and chain morphism framework.

The Boardman-Vogt tensor extension conjectures (BoardmanVogt.lean) are stated as weak placeholders with no downstream dependency.

Derived (novel arguments, structurally complete, not yet fully formalized)**:**

- *Fixed-point instantiation:* G_{Cat} as least fixed point of M on EAT — the general fixed-point machinery is formalized (the substrate-independent fixed point is proved in Lean 4 for any category satisfying the hypotheses), but the specific instantiation to EAT awaits Gabriel-Ulmer duality (Gabriel and Ulmer 1971) in Lean (*Grammar*).
- *Substrate independence:* $D = 1$ as substrate-independence principle — the formal content (fixed-point uniqueness) is formalized, the conceptual content (axiom-schema complexity as a metric) is a paper-level observation (*Computation*, *Adjunction*).
- *Tensor and specification:* The Boardman-Vogt tensor extension to EAT (*Adjunction*); specification identity resolving Ψ without constructing the functor (*Adjunction*).
- *Universality bridge:* Church-Turing as characterization of the convergence threshold — the categorical argument connecting reflexive objects to universality is formalized; the full bridge deriving all CompModel axioms from the categorical structure remains a paper-level argument pending the EAT instantiation (*Computation*, *Adjunction*).
- *Geometric correspondence:* The S^2/G_{Cat} structural correspondence (*Method*).
- *Dimensionality:* M as the canonical dimension generator and computation as constructively posterior to dimensionality (chain vs. colimit); the CPS correspondence: continuation-passing at $R = D$ collapses via reflexivity, identifying CPS with M restricted to the computational fragment (*Dimensionality*).

Conjectured (claims requiring unresolved mathematical steps)**:** full closed monoidal structure of EAT (associativity, unit, tensor-hom adjunction); the strong Ψ conjecture (monoidal 2-functor); formal equivalence of the S^2/G_{Cat} pairs; M as the terminal dimension-generating operation — every coherent notion of dimension factors through the categorical dimensional ladder (*Dimensionality*).



The Lean 4 Pairing

The full Lean 4 formalization is archived at doi:10.5281/zenodo.18878239.

Category theory is one of the rare domains where formalization is not a parallel track but the same content in a different medium. Lean 4's type theory is itself categorical — types are objects, functions are morphisms, and dependent types correspond to fibrations under the Grothendieck construction — a structural correspondence that the paired Lean 4 project makes explicit. The formalization covers four layers. Layer 1: the categorical substrate (Adámek's theorem, Lambek's lemma, substrate-independent fixed-point existence and uniqueness, convergence machinery). Layer 2 — the main result: the fixed point IS a model of universal computation. The reflexive object $D \cong [D, D]$, containerization, identity modulation, and the untyped lambda calculus model are proved from the Lambek isomorphism alone — no external enumeration or $\mathbb{N}$-indexing is needed. Self-application, Y combinator, naming equivalence, universal evaluator, and self-indexed Kleene recursion follow; the classical $\mathbb{N}$-indexed computability results are a corollary. Layer 3: the $\mathbb{N}$-indexed bridge (Church-Turing characterization, Rogers isomorphisms, classical Kleene recursion). Layer 4: divergence and convergence witnesses (FinSet, thin categories, convergence criterion). The dimensionality infrastructure (truncation-level dimension, increment by one, stabilization at the fixed point) is machine-checked across all four layers. The project stands at 42 files with 0 sorry and 0 custom axioms across all layers. What remains unformalized is the specific instantiation to EAT (awaiting Gabriel-Ulmer duality in Mathlib) and the conjectured items (closed monoidal structure of EAT, strong Ψ).

The next formalization targets, in dependency order: the full Church-Turing bridge (deriving all CompModel axioms from the categorical structure, pending the EAT instantiation); categorical CPS foundations; and the Boardman-Vogt tensor (blocked by Gabriel-Ulmer duality in Mathlib).

The formalization does not replace the visual construction. It certifies it in a second medium — one that is machine-checkable and phase-invariant. The visual construction makes the mathematics visible. The formalization makes it undeniable.

Elements

A Constructive Derivation of Category Theory

Larsen James Close

Destination

We are going to build a mathematical structure from almost nothing — a point, an arrow, and the rule that arrows placed tip-to-tail compose — and prove that the structure we build fully determines itself through its own internal relations. Most mathematical representations lose information. This one provably does not. The medium of construction is the diagram. The final theorem, the Yoneda lemma, says: representing an object by the pattern of arrows pointing at it is faithful — no information is lost. The representational strategy the construction uses is what the construction proves to be exact.

The reader should hold this destination in mind throughout. Every diagram below is a step toward it. When the theorem arrives, it should feel like recognition.

The Motivating Pressure

We want to study structure without fixing what the objects are.

Sets have elements. Groups have multiplication tables. Topological spaces have open sets. Each comes with internal detail that is particular to its kind. But the *relationships* between sets, between groups, between topological spaces — the functions, the homomorphisms, the continuous maps — obey the same compositional logic regardless of the internal detail. Composition is typed, associative, and has identities. That pattern recurs across every domain of mathematics.

The question is: can we study that pattern directly, without first deciding what kind of thing we are talking about?

Typed, directed composition — maps between things that compose when their types match — is what remains when the domain-specific detail is stripped away. It is the most abstract compositional structure and the most fundamental one: the starting point the construction arrives at, not one it selects. The axioms that follow are what that pattern requires.

Act I: The Primitives

Proposition 1. The Object

$$A$$

A point. It has no internal structure visible at this level. We do not know what A is — only that it is something that can stand in relation. The point is not defined. It is placed.

Proposition 2. The Morphism

$$A \xrightarrow{f} B$$

An arrow from A to B. It is directed: it goes *from A to B*, not the reverse. It is typed: it has a definite source and a definite target. These properties are not axiomatized — they are visible. The arrow carries them the way a compass stroke carries circularity.

The arrow f is a morphism. It represents a structure-preserving passage from A to B. We do not yet say what "structure-preserving" means, because we have not yet said what structure the objects have. The arrow is the primitive; the structure will be read off from how arrows compose.

Proposition 3. Composition

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[rr, bend right, "g\circ f"'] & B \arrow[r, "g"] & C \end{tikzcd}$$

Given an arrow $f : A \to B$ and an arrow $g : B \to C$ — two arrows placed tip-to-tail — there exists an arrow $g \circ f : A \to C$. The composite.

This is the single generative operation. Everything that follows is a consequence of the fact that arrows placed tip-to-tail yield a new arrow. The diagram makes the operation visible: two arrows that connect produce a third that spans them.

Note what is already present in the picture. Composition is *typed*: f and g compose only because the target of f (which is B) matches the source of g (which is B). If the arrows did not meet, there would be no composite. The diagram enforces this — you cannot place arrows tip-to-tail unless they connect.

Proposition 4. Identity

$$\begin{tikzcd} A \arrow[loop left, "\mathrm{id}_A"] \end{tikzcd}$$

Every object has a morphism to itself. The identity morphism $\mathrm{id}_A : A \to A$ is the compositional expression of A being A.

Its defining property:

$$\begin{tikzcd} A \arrow[loop left, "\mathrm{id}_A"] \arrow[r, "f"] & B \arrow[loop right, "\mathrm{id}_B"] \end{tikzcd}$$

$$f \circ \mathrm{id}_A = f = \mathrm{id}_B \circ f$$

The identity morphism is a precondition for any of this to hold together. It is what makes an object a stable node in the compositional structure — present and addressable as a fixed reference point.

Proposition 5. Associativity

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[rrr, bend right, "h\circ g\circ f"'] & B \arrow[r, "g"] & C \arrow[r, "h"] & D \end{tikzcd}$$

Given three composable arrows f, g, h, the two possible groupings yield the same result:

$$h \circ (g \circ f) = (h \circ g) \circ f$$

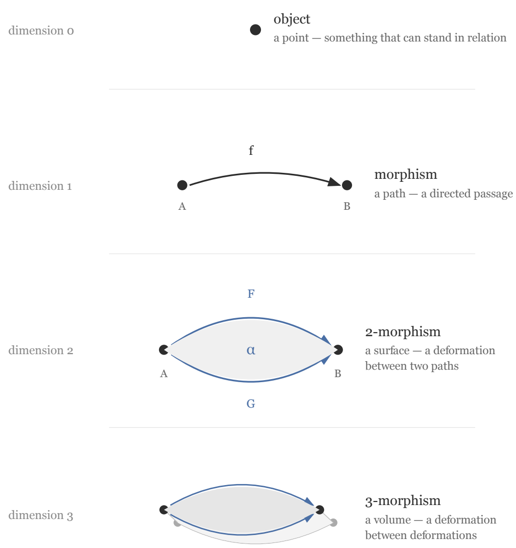
The path from A to D does not depend on how you bracket the intermediate steps. The diagram shows why: there is one path through B and C from A to D, regardless of which pair you compose first.

This completes the primitives. A **category** is a collection of objects and morphisms satisfying these three conditions: composition of compatible arrows, identity arrows, and associativity. The diagram drew all three without stating axioms. The visual medium carried them.

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[rr, bend left, "g\circ f"] \arrow[rrr, bend right, "h\circ g\circ f"'] & B \arrow[r, "g"] \arrow[rr, bend left, "h\circ g"] & C \arrow[r, "h"] & D
\end{tikzcd}$$

Everything commutes. Every path between the same two objects yields the same morphism. This is the content of associativity made visible: the diagram *is* the proof.

What we have built is a tower of dimensional structure — objects as points, morphisms as directed paths between them. The next act extends this tower upward.

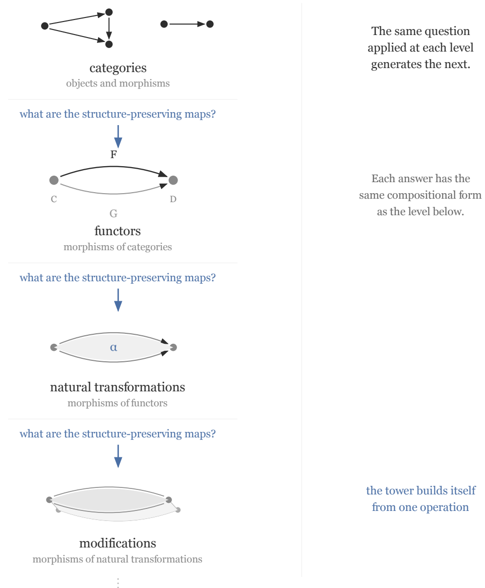


Act II: The Iterating Question

We have built one category. Now we ask the question that generates the rest of the theory:

What are the structure-preserving maps between categories?

This single question, applied iteratively, constructs the entire tower.



Proposition 6. The Functor

Let $\mathcal{C}$ and $\mathcal{D}$ be categories. A **functor** $F : \mathcal{C} \to \mathcal{D}$ sends:

- each object A in $\mathcal{C}$ to an object $F(A)$ in $\mathcal{D}$
- each morphism $f : A \to B$ in $\mathcal{C}$ to a morphism $F(f) : F(A) \to F(B)$ in $\mathcal{D}$

preserving the structure we just built:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "F"'] & B \arrow[d, "F"] \\ F(A) \arrow[r, "F(f)"'] & F(B) \end{tikzcd}$$

$$F(g \circ f) = F(g) \circ F(f) \qquad\qquad F(\mathrm{id}_A) = \mathrm{id}_{F(A)}$$

A functor preserves composition and identities. That is the entire definition. It is "structure-preserving map" made precise for the structure we have: a functor is a morphism of categories.

The diagram makes the key point: the square commutes. Going across-then-down gives the same result as going down-then-across. The functor translates the structure of $\mathcal{C}$ into $\mathcal{D}$ without distortion.

The move here is the one announced in the introduction: we asked "what are the structure-preserving maps?" at the next level, and the answer has exactly the form we should expect — it preserves the operations (composition and identity) that define the structure.

Proposition 7. Functors Compose

$$\begin{tikzcd} \mathcal{C} \arrow[r, "F"] \arrow[rr, bend right, "G \circ F"'] & \mathcal{D} \arrow[r, "G"] & \mathcal{E} \end{tikzcd}$$

Functors compose: if $F : \mathcal{C} \to \mathcal{D}$ and $G : \mathcal{D} \to \mathcal{E}$, then $G \circ F : \mathcal{C} \to \mathcal{E}$ is a functor. The identity functor $\mathrm{Id}_{\mathcal{C}} : \mathcal{C} \to \mathcal{C}$ sends every object and morphism to itself.

This is Proposition 3 again, one level up. Categories are the objects. Functors are the morphisms. Composition and identity work as before. **Categories and functors form a category.**

The reader should pause here. The structure we built in Act I — objects, morphisms, composition, identity — has reproduced itself. The tool we constructed is an instance of the pattern it describes. This self-reproduction is not an accident. It is the engine of everything that follows.

Proposition 8. The Natural Transformation

We apply the question again: *what are the structure-preserving maps between functors?*

Given two functors $F, G : \mathcal{C} \to \mathcal{D}$, a **natural transformation** $\alpha : F \Rightarrow G$ assigns to each object A in $\mathcal{C}$ a morphism $\alpha_A : F(A) \to G(A)$ in $\mathcal{D}$, such that for every morphism $f : A \to B$ in $\mathcal{C}$:

$$\begin{tikzcd} F(A) \arrow[r, "\alpha_A"] \arrow[d, "F(f)"'] & G(A) \arrow[d, "G(f)"] \\ F(B) \arrow[r, "\alpha_B"'] & G(B) \end{tikzcd}$$

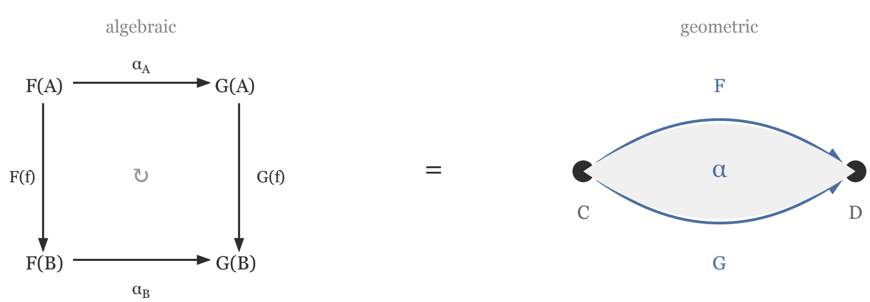
The square commutes: $G(f) \circ \alpha_A = \alpha_B \circ F(f)$.

A natural transformation is a consistent way of deforming one functor into another — "consistent" meaning that the deformation respects every morphism in the source category. The commutativity of the square is the entire content: it says the transformation commutes with the structure.

This is the second iteration of the question, and the pattern holds. Natural transformations have the same compositional character: they compose (vertically and horizontally), they have identities (the identity natural transformation). Functors between two fixed categories, with natural transformations between them, form a category — a **functor category**.

The tower continues. One could ask about modifications (maps between natural transformations), and the pattern would repeat. But the essential structure is already present: the iterating question produces new levels, and each level has the same compositional form as the one below.

The naturality square above is the algebraic view. The geometric view is a surface filling in between two paths — the same constraint, seen differently:



Proposition 9. The Hom-Functor

Fix an object A in a category $\mathcal{C}$. For every object X in $\mathcal{C}$, consider the collection of all morphisms from X to A:

$$\mathrm{Hom}(X, A) = \{f : X \to A\}$$

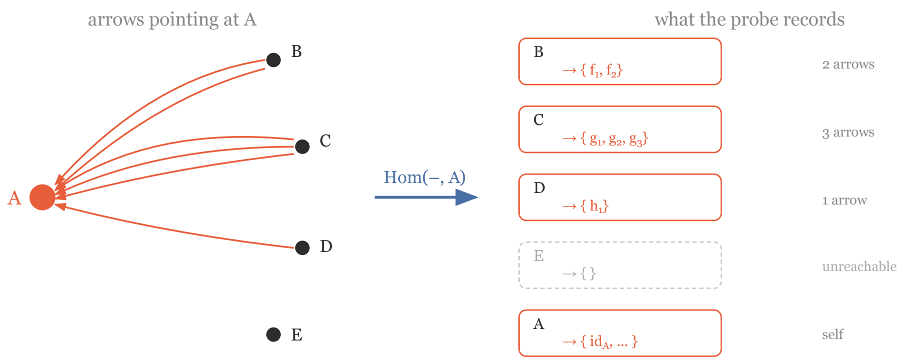
This assignment is functorial — contravariantly. Given a morphism $g : X \to Y$, there is a map $\mathrm{Hom}(g, A) : \mathrm{Hom}(Y, A) \to \mathrm{Hom}(X, A)$ defined by composition:

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

$$f \mapsto f \circ g$$

So $\mathrm{Hom}(-, A)$ is a functor from $\mathcal{C}^{\mathrm{op}}$ to **Set**. It takes each object X to the set of arrows from X to A, and each morphism to pre-composition.

This is the **probe**. Every object in the category sends arrows toward A. The functor $\mathrm{Hom}(-, A)$ collects the results: for each source, the set of all arrows from that source to A.



The probe records everything the rest of the category "sees" about A — its complete inward-facing relational profile. Some objects send many arrows toward A, some few, some none at all. Each object defines a different probe. Different objects attract different arrows.

Most mathematical representations are projections that discard some structure. The question the Yoneda lemma will answer is: does this record determine A completely?

Dually, for a fixed object B, the assignment $X \mapsto \mathrm{Hom}(B, X)$ is a covariant functor — it records what the category looks like when probed *from* B *toward* every object.

Proposition 10. The Yoneda Embedding

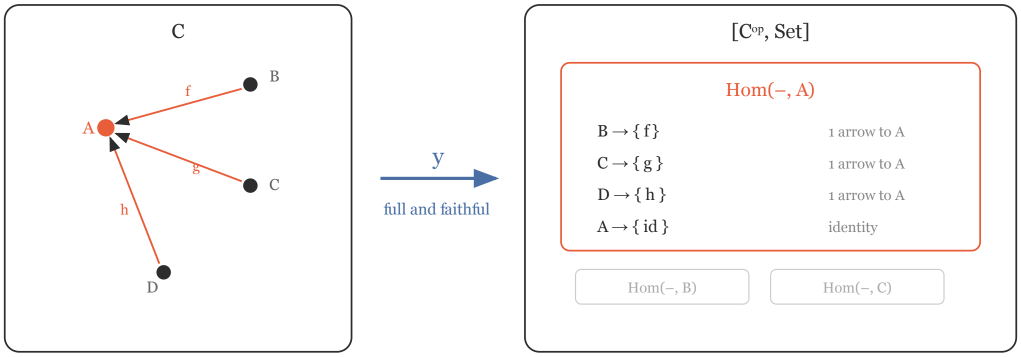
The assignment $A \mapsto \mathrm{Hom}(-, A)$ is itself functorial. It defines a functor:

$$\mathbf{y} : \mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$$

sending each object to its probe, and each morphism to the induced transformation between probes.

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "\mathbf{y}"'] & B \arrow[d, "\mathbf{y}"] \\ \mathrm{Hom}(-, A) \arrow[r, "\mathbf{y}(f)"'] & \mathrm{Hom}(-, B) \end{tikzcd}$$

The Yoneda embedding sends each object to its representable functor — the complete record of how the rest of the category relates to it.



The object A with its arrows maps to the complete relational profile $\mathrm{Hom}(-, A)$. The question is whether this mapping loses anything. That is what the next proposition answers.

Act III: The Certification

Proposition 11. The Yoneda Lemma (Mac Lane 1998)

For any functor $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$ and any object A of $\mathcal{C}$, there is a natural bijection:

$$\mathrm{Nat}(\mathrm{Hom}(-, A),\ F) \ \cong\ F(A)$$

Natural transformations from the probe of A into any functor F correspond exactly to elements of $F(A)$. Watch what happens to a single morphism $f : B \to A$ under naturality:

$$\begin{tikzcd} \mathrm{Hom}(A, A) \arrow[r, "\alpha_A"] \arrow[d, "\mathrm{Hom}(f, A)"'] & F(A) \arrow[d, "F(f)"] \\ \mathrm{Hom}(B, A) \arrow[r, "\alpha_B"'] & F(B) \end{tikzcd}$$

A natural transformation α is entirely determined by what it does to $\mathrm{id}_A \in \mathrm{Hom}(A, A)$. Set $x = \alpha_A(\mathrm{id}_A) \in F(A)$. Then naturality forces, for every $f : B \to A$:

$$\alpha_B(f) = F(f)(x)$$

The transformation has no freedom. Once x is chosen, the rest is determined by the structure. Conversely, every element $x \in F(A)$ defines such a transformation. The correspondence is natural in both A and F.

Proposition 12. Full Faithfulness of the Yoneda Embedding

The Yoneda lemma holds for any functor F. Choosing F to be another representable — specifically $\mathrm{Hom}(-, B)$ — turns the general result into a statement about the relationship between morphisms in $\mathcal{C}$ and natural transformations between their probes. Applying the Yoneda lemma with $F = \mathrm{Hom}(-, B)$:

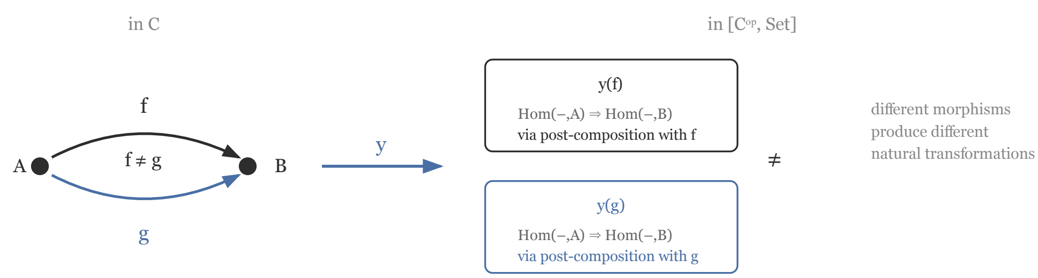
$$\mathrm{Nat}(\mathrm{Hom}(-, A), \mathrm{Hom}(-, B)) \ \cong\ \mathrm{Hom}(A, B)$$

Every natural transformation between representable functors comes from a morphism in the original category. The Yoneda embedding $\mathbf{y} : \mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ is **full and faithful**: it embeds $\mathcal{C}$ into the functor category without losing any information.

$$\mathcal{C} \xhookrightarrow{\mathbf{y}} [\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$$

An object is nothing more and nothing less than the pattern of arrows pointing at it.

What "full and faithful" means concretely: distinct morphisms in $\mathcal{C}$ produce distinct natural transformations (faithful — nothing collapses), and every natural transformation between representable functors comes from a morphism (full — nothing is missed).



The Certification

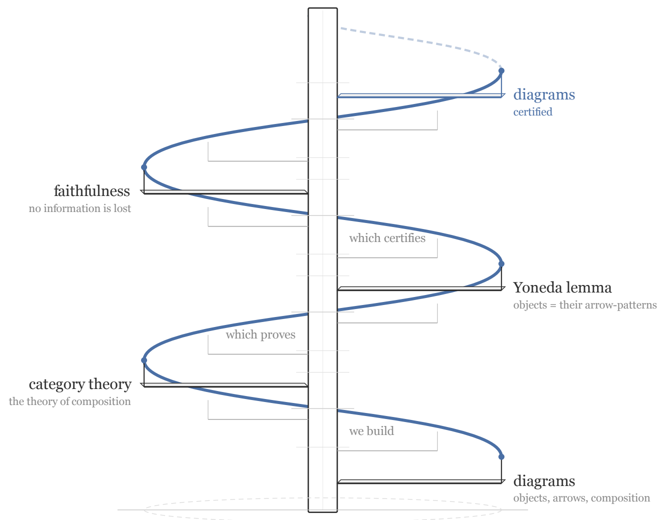
The construction is complete. Return to where we began.

We set out to study structure without fixing what the objects are. We drew a point and an arrow. We observed that arrows compose, that identity arrows exist, that composition is associative. These were not axioms imposed — they were properties visible in the diagrammatic medium.

We asked one question — *what are the structure-preserving maps?* — and applied it iteratively. It produced functors, then natural transformations. Each level had the same compositional form as the level below.

Then we built the probe: each object determined by the morphisms pointing at it. The Yoneda lemma says the probe is complete — a natural transformation from a probe is the same thing as a choice of element, nothing more. The Yoneda embedding says the map from objects to their probes is faithful — no information is lost.

The Yoneda embedding certifies a specific mathematical fact: objects are fully determined by the morphisms pointing at them. The embedding $\mathbf{y} : \mathcal{C} \hookrightarrow [\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ is full and faithful. This is the certification: (1) Yoneda proves that the relational profile — the hom-functor $\mathrm{Hom}(-, A)$ — loses no information about A. (2) The diagrams we have drawn from the first proposition display exactly this content: objects identified by their arrow-patterns, structure visible as paths. (3) Therefore the diagrammatic medium is matched to the certified content — it makes visible the very thing Yoneda proves is exact. The certification is of (1), the mathematical content. The match with diagrams — that the visual medium works — is the observation (2)+(3), which follows from (1) but is not itself a theorem.


diagrams
certified
faithfulness
no information is lost
which certifies
Yoneda lemma
objects = their arrow-patterns
which proves
category theory
the theory of composition
we build
diagrams
objects, arrows, composition

Grammar

The Tower

Larsen James Close

Destination

Category theory's axioms are a formal grammar. The meta-rule "what are the structure-preserving maps between structures at this level?" is the single production that generates the dimensional tower. Applied to the grammar, it returns the same grammar one level up. The grammar is a fixed point of its own meta-rule — and not just any fixed point, but the least one: the simplest compositional grammar closed under the operation of examining its own models. Every other fixed point in the tower — 2-categories, n-categories, ∞-categories — is reachable by iterating from this seed.

The companion document *Elements* constructs the tower. *Map* maps it. This paper recognizes the tower as the least fixed point of a meta-generative grammar.

The Grammar

A **compositional grammar** is a finite specification consisting of generating sorts, typed operations on those sorts, and equational laws constraining the operations. The operations may be partial — composition of morphisms, for instance, is defined only when the types match. In the language of universal algebra, this is an *essentially algebraic theory* (Freyd 1972) — a theory whose operations may have domains of definition specified by equations, generalizing the single-sorted total-operation framework of Lawvere theories (Lawvere 1963) to the multi-sorted partial-operation setting that category theory requires. The category of essentially algebraic theories and their morphisms is the ambient setting for everything that follows.[1] It has finite limits because limits of models of any essentially algebraic theory are computed sort-by-sort — the limit of a diagram of theories is the theory whose sorts, operations, and equations are the limits of the corresponding components. It has directed colimits for the same structural reason: directed colimits of essentially algebraic theories are computed sort-by-sort. These are the two conditions the iteration construction below requires: directed colimits for the sequence to have a target, and an initial object — the empty theory $\emptyset$, which has no sorts, no operations, and no equations, and is initial because there is exactly one theory morphism from $\emptyset$ to any other theory: the morphism that sends nothing to nothing. A precision note: the closely related category of Lawvere theories (single-sorted, total operations) admits a tensor product (Boardman and Vogt 1973; Hyland and Power 2007) and closed structure. Whether that closed structure extends to the full EAT setting is resolved in *Adjunction*: the extension is well-defined because the domain conditions of two theories are written in disjoint vocabularies and cannot conflict under interleaving. This matters for the strongest form of the conjecture in *Computation* but not for the present paper's claims, which require only the iteration machinery. What is novel here is not the framework but a specific claim about a specific object within it.

The categorical grammar G_{Cat} has:

[1] We work throughout with locally small categories and a Grothendieck universe U large enough that $\mathbf{Cat}_U$ is small relative to the next universe; the construction is insensitive to choice of U because $D = 1$ is invariant under universe enlargement (see *Dimensionality* §2).

Sorts:
Objects: $A, B, C, \ldots$
Morphisms: $f : A \to B$ (typed, directed)

Operations:
Composition: $f : A \to B,\ g : B \to C \ \vdash\ g \circ f : A \to C$
Identity: $\vdash\ \mathrm{id}_A : A \to A$

Laws:
Associativity: $h \circ (g \circ f) = (h \circ g) \circ f$
Unit: $f \circ \mathrm{id}_A = f = \mathrm{id}_B \circ f$

Meta-rule M:
$G \ \mapsto\ \{\text{structure-preserving maps between } G\text{-models}\}$

Two sorts, one binary operation, one nullary operation family, two equational laws, one meta-rule. This is the entire specification. Everything that follows is generated output.

The minimality is visible: a monoid has one sort, one operation, and the same two laws — but it is not closed under M. A group adds an inverse operation. A ring adds a second sort of operation and a distributivity law. Each is a valid compositional grammar. None is a fixed point.

The Meta-Rule

Define M precisely. Given a compositional grammar G:

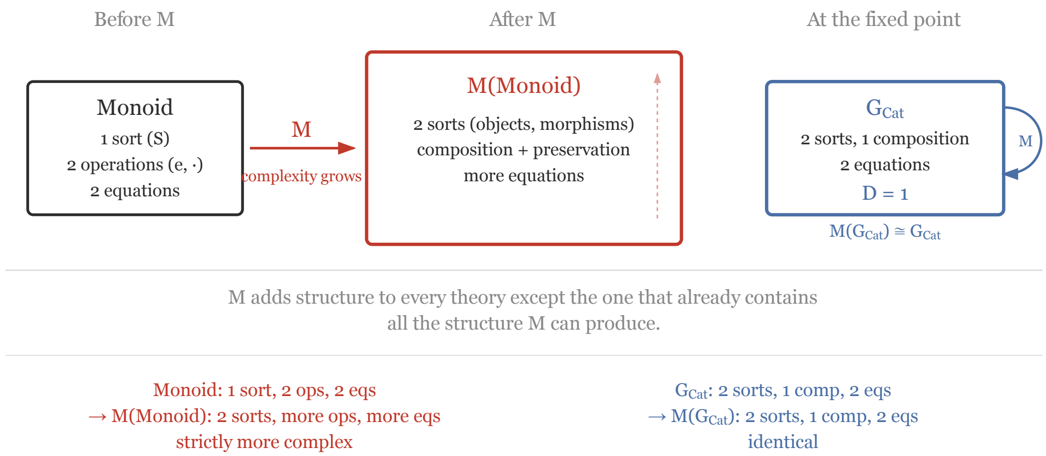
- A **G-model** (or G-algebra) is a structure satisfying G's axioms.
- A **G-homomorphism** is a map between G-models that preserves all of G's operations and respects all of G's laws.
- $M(G)$ is the grammar whose models are G-models, whose morphisms are G-homomorphisms, and whose composition and identity are ordinary function composition and identity.

$M(G)$ always has categorical structure. Regardless of what G is, the collection of G-models and their homomorphisms has objects (the models), typed morphisms (the homomorphisms), composition (which is associative), and identities. M categorifies its input. When EAT carries a monoidal closed structure with tensor $\otimes$, the meta-rule $M(T, -)$ is the internal hom — the right adjoint of $\otimes$. Fixing the first argument recovers the unary endofunctor $G \mapsto M(G)$ used throughout this paper.

This is the key structural observation: M **maps every compositional grammar into categorical territory.** The output of M always satisfies G_{Cat}'s axioms. Every grammar, when you ask "what are the structure-preserving maps between its models?", yields a category.

Applied to specific grammars:

- $M(\text{Monoid})$ = the category of monoids and monoid homomorphisms. Categorical structure, not monoidal. $M(\text{Monoid}) \ncong \text{Monoid}$.
- $M(\text{Group})$ = the category of groups and group homomorphisms. Categorical, not group-theoretic. $M(\text{Group}) \ncong \text{Group}$.
- $M(\text{Ring})$ = the category of rings and ring homomorphisms. Same pattern.
- $M(G_{\mathrm{Cat}})$ = the category of categories and functors. Categorical structure — *the same axiom set as the input.* $M(G_{\mathrm{Cat}}) \cong G_{\mathrm{Cat}}$.



G_{Cat} is a fixed point. The grammar of functors between categories is the grammar of categories.

Why $M(G_{\text{Cat}}) \cong G_{\text{Cat}}$ in detail. $M(G_{\text{Cat}})$ is the theory of categories-and-functors-between-them. Its models are categories. Its morphisms are functors. What governs this collection? Objects are categories. Morphisms (functors) go between categories. Functors compose (Proposition 7 of the *Elements*). Each category has an identity functor. Composition of functors is associative. The axiom schema: two sorts (objects and morphisms), one typed binary operation (composition), one nullary operation family (identity), two equational laws (associativity and unit). This is the axiom schema of G_{Cat}. The isomorphism $M(G_{\text{Cat}}) \cong G_{\text{Cat}}$ is not a coincidence of structural similarity — it is the observation that the grammar governing structure-preserving maps between G_{Cat}-models reproduces G_{Cat}'s axiom schema exactly. The theory morphisms are explicit: in one direction, send "object" to "G_{Cat}-model" and "morphism" to "G_{Cat}-homomorphism"; in the other, send "G_{Cat}-model" to "object" and "G_{Cat}-homomorphism" to "morphism." Both composites are the identity on sorts and operations, so the pair is an isomorphism in EAT.

The Generated Language

Iterating M from G_{Cat} generates the tower:

Level 0. G_{Cat} itself: objects, morphisms, composition, identity, associativity. This is the grammar.

Level 1. $M(G_{\text{Cat}})$: categories and functors. Functors compose, have identities, satisfy associativity. The grammar of the output is G_{Cat}.

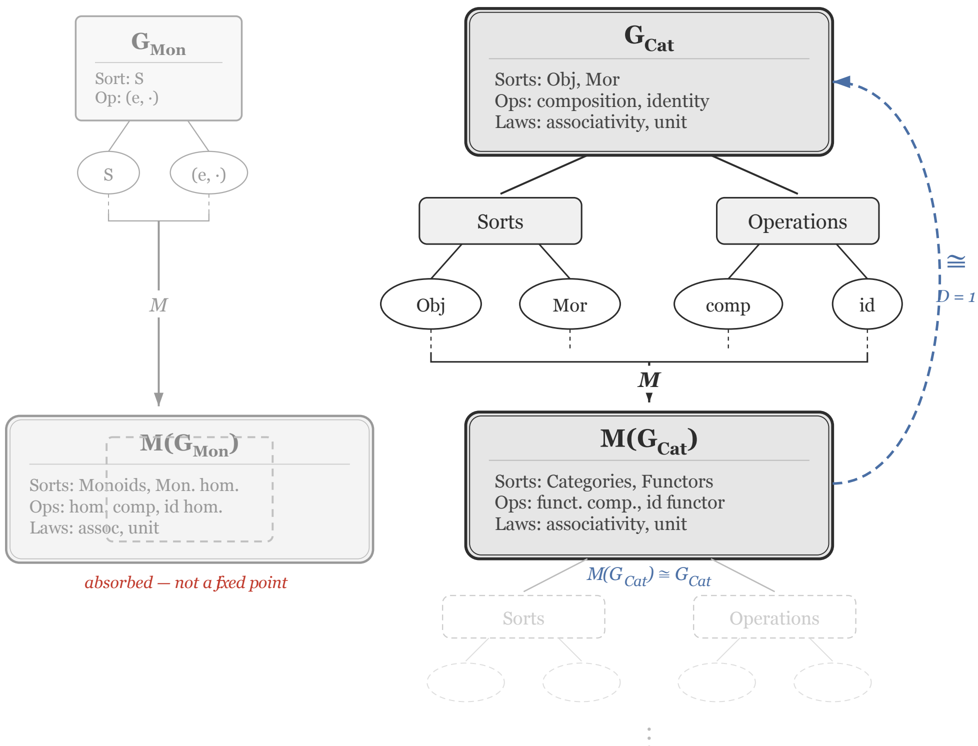
Level 2. $M^2(G_{\text{Cat}})$: applying M to "the category of categories and functors" asks for structure-preserving maps between functors — natural transformations. Natural transformations compose (vertically), have identities, satisfy associativity. The grammar of the output is G_{Cat}.

Level 3. $M^3(G_{\text{Cat}})$: applying M to the functor categories of Level 2 asks for structure-preserving maps between natural transformations — **modifications**. A modification between two natural transformations $\alpha, \beta : F \Rightarrow G$ assigns to each object a morphism mediating between α_A and β_A, subject to coherence conditions that are themselves instances of composition, identity, and associativity. Modifications compose, have identities, and composition is associative. The grammar of the output is G_{Cat}.

Level n**.** $M^n(G_{\text{Cat}})$: n-morphisms between $(n{-}1)$-morphisms. The grammar at every level is G_{Cat}.

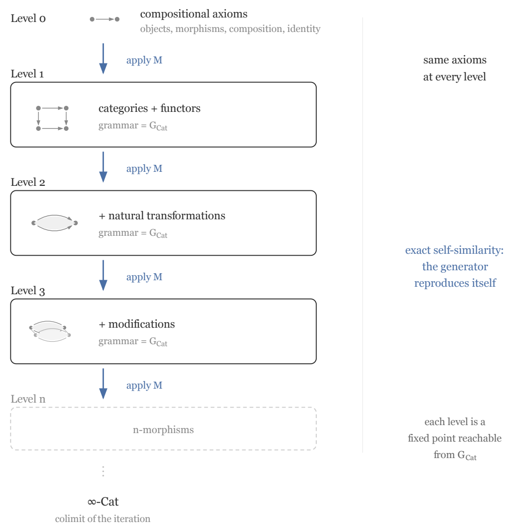
The tower is the generated language of the grammar under iteration of M. Each level is a word in the language; the production rule that generates each word from the last is always M. The language is infinite and every word has the same grammatical structure.

This is **exact self-similarity**. The structure at level $n+1$ is not *approximately* the same as level n — it is *identically* the same axiom set applied to the models of level n. No distortion, no loss, no additional structure required at higher levels. The generator reproduces itself exactly under its own production rule.

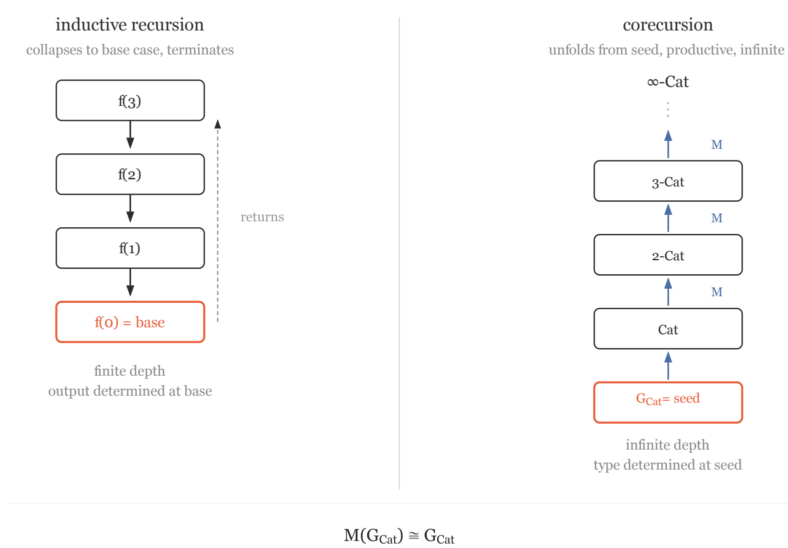


The Recursive Structure

The construction has the form of a recursive algorithm. G_{Cat} is the base case. M is the recursive step. The fixed-point condition $M(G_{\text{Cat}}) \cong G_{\text{Cat}}$ is the statement that the base case is a fixed point of its own recursive step — the recursion, applied to its starting point, returns to where it began.



But the recursion does not terminate. Each application of M produces the next level of the tower, and there is no final level. The tower is **corecursive**: an infinite productive process whose type is determined from the first step. This is the distinction between induction and coinduction. An inductive definition builds up from a base case and halts — a natural number, a finite list, a tree with leaves. A coinductive definition produces output without bound — a stream, an infinite tree, a tower of n-categories for all n.



G_{Cat} as least fixed point is the inductive side: the seed from which the tower grows, computable, determinate, constructive. The full tower including $G_{\infty\text{-Cat}}$ is the coinductive side: the terminal coalgebra of M, the greatest fixed point, the limit of the infinite unrolling. The duality between these two — initial algebra and terminal coalgebra of the same endofunctor — is itself a categorical phenomenon. The recursion that generates the tower is an instance of the structure the tower describes. The Adámek chain $\emptyset \to M(\emptyset) \to M^2(\emptyset) \to \cdots$ is a functor from ω to EAT — a diagram in a category. Diagrams in categories are the subject matter of the tower's objects. The construction is a morphism in the structure it constructs.

This also clarifies what kind of self-reference is at work. Gödelian self-reference produces undecidability — a statement that refers to its own unprovability loops without resolution. Recursive self-reference in a well-founded domain produces a least fixed point — computable, determinate, productive. The grammar's self-reference is recursive in the computational sense: $G = M(G)$ is a recursive equation with a canonical solution, not a paradox. That is why the construction is productive rather than destructive.

For the Lean 4 formalization: the fixed-point existence (Adámek's theorem, Lambek's lemma, uniqueness) is fully proved. The tower as a coinductive object — the infinite stream of levels — would require `codata` or `Stream`-like constructions in Lean 4, and is not formalized. The Lean project proves the fixed point exists and is unique; representing the full infinite tower is a separate formalization target.

Self-Similarity and the Fixed Point

Exact versus Approximate

A fractal generated by an iterated function system — the Sierpinski triangle, the Koch curve, the Mandelbrot set — has approximate self-similarity. Zoom into the boundary of the Mandelbrot set and you see structures that resemble the whole, but each copy is embedded in a different local geometry, slightly distorted, never exactly identical. The self-similar dimension is fractional: $\log 3/\log 2 \approx 1.585$ for the Sierpinski triangle. The copies carry the generator's signature but not its exact form.

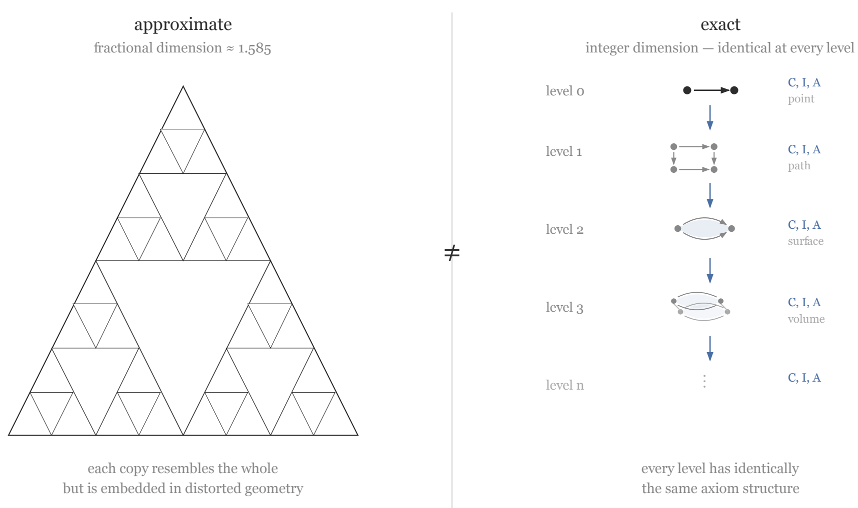
The categorical tower has exact self-similarity. The axiom structure at each level is not similar to the axiom structure at every other level — it is identical. The functor M sends G_{Cat} to G_{Cat}, with no residual, no distortion, no additional axioms needed. The self-similar dimension is integer: each level is a full copy of the same finite grammar. The copies don't merely carry the generator's signature — they *are* the generator.

The distinction matters because exact self-similarity constrains the fixed-point structure. A system with approximate self-similarity can have many fixed points at different scales, each slightly different. A system with exact self-similarity has a fixed point that reproduces without variation — and the question of how many such fixed points exist is controlled by the structure of the generator.

We call this condition $D = 1$: exact self-reproduction. The axiom-schema complexity is constant under iteration — the operation returns what was put in, with no residual contribution from the ambient category. $D = 1$ is not a separate measurement imposed on the fixed point; it is the fixed-point condition itself, read for its consequence: the ambient category does not participate.

A precision note on measurement. The claim that the axiom structure is identical at every level is stated for the canonical presentations produced by M at each iteration step. These presentations are canonical because M is a definite construction — it takes a theory and produces a specific theory, not an equivalence class. Two different presentations of the same theory could differ in their sort/operation/equation counts, so the invariance claim is about the specific iteration chain, not about arbitrary presentations of the theories involved. The sequence $C(\emptyset), C(M(\emptyset)), C(M^2(\emptyset)), \ldots$ is well-defined. $D = 1$ says this sequence stabilizes — the iteration does not produce theories of increasing complexity. This is a stronger observation than $M(G_{\mathrm{Cat}}) \cong G_{\mathrm{Cat}}$ alone: the isomorphism says the iteration has a fixed point; $D = 1$ says the complexity was already at its final value when the fixed point was reached. In contrast, non-fixed-point grammars show growth: $C(M(\text{Monoid}))$ exceeds $C(\text{Monoid})$, because the category of monoids has richer axiom structure than the monoid axioms themselves.

What the isomorphism witnesses concretely: $M(G_{\text{Cat}})$ is the theory whose models are structure-preserving maps between categories. Its sorts, operations, and equations are determined by M's construction — a sort for functors, operations for applying a functor to objects and morphisms, equations asserting preservation of composition and identity. The isomorphism $M(G_{\text{Cat}}) \cong G_{\text{Cat}}$ is realized by explicit theory morphisms: in one direction, each sort/operation/equation in $M(G_{\text{Cat}})$ maps to its correspondent in G_{Cat} (functors between categories are themselves objects of a category; natural transformations between functors are morphisms; the preservation equations instantiate the same associativity and unit laws). In the other direction, G_{Cat} maps into $M(G_{\text{Cat}})$ because the categorical axiom schema is exactly what M produces when given the categorical axiom schema as input. These are definite morphisms, not an abstract guarantee of isomorphism — the identity of axiom schemas is witnessed, not merely asserted.



The Least Fixed Point

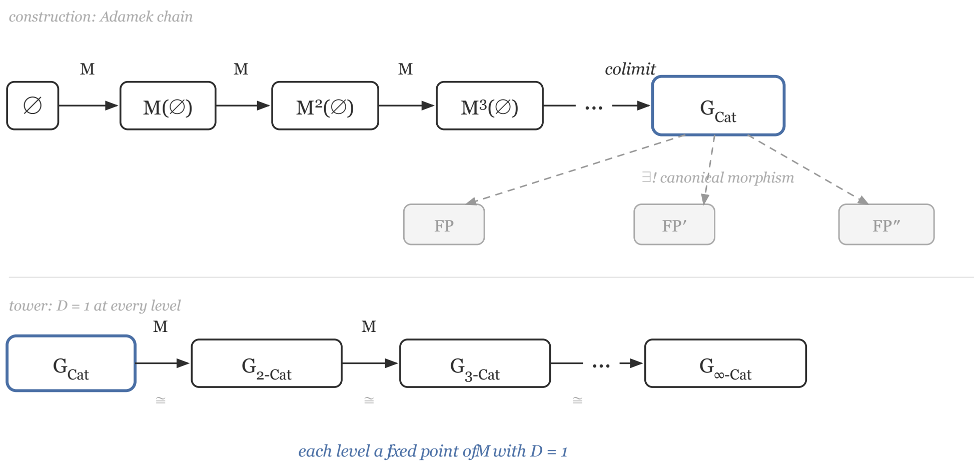
G_{Cat} is not the only fixed point of M. The grammar $G_{\text{2-Cat}}$ — 2-categories, with objects, 1-morphisms, 2-morphisms, two composition operations, and the interchange law — is also a fixed point: structure-preserving maps between 2-categories (2-functors) form a 2-category, with 2-natural transformations as 2-cells. Similarly, $G_{n\text{-Cat}}$ is a fixed point for every n, and $G_{\infty\text{-Cat}}$ (with appropriate definitions) is a fixed point.

But each of these is **reachable from G_{Cat} by iteration**. $G_{\text{2-Cat}}$ emerges at level 2 of the tower generated by iterating M from G_{Cat}. $G_{n\text{-Cat}}$ emerges at level n. The tower

$$G_{\text{Cat}} \to G_{\text{2-Cat}} \to G_{\text{3-Cat}} \to \cdots$$

exhibits the fixed-point property: each level is isomorphic to G_{Cat} as an essentially algebraic theory, because applying M to a grammar with full categorical structure returns a grammar with the same categorical structure. The tower is self-similar from the first stage onward.

A note on Homotopy Type Theory. The tower $G_{\text{Cat}} \to G_{2\text{-Cat}} \to G_{3\text{-Cat}} \to \cdots$ resembles the hierarchy of truncation levels in HoTT (n-types, n-groupoids), and both produce higher categorical structure. The mechanisms are different. HoTT starts from type theory and builds higher categorical structure via identity types and the univalence axiom — the tower arises from the internal logic of types. This construction starts from compositional grammar and derives the tower as the iteration of a single operation M on essentially algebraic theories — the tower arises from a fixed-point condition on a meta-rule. The key structural difference: in HoTT the categorical structure genuinely changes at each truncation level (a 1-type is not a 2-type), while here the axiom schema is identical at every level ($D = 1$). HoTT's tower is stratified; this tower is self-similar. The approaches are complementary — both arrive at higher category theory, one by iterating a type-theoretic construction, the other by iterating a grammar-theoretic one — but the fixed-point property ($M(G_{\text{Cat}}) \cong G_{\text{Cat}}$, exact self-reproduction) has no analogue in the HoTT program.



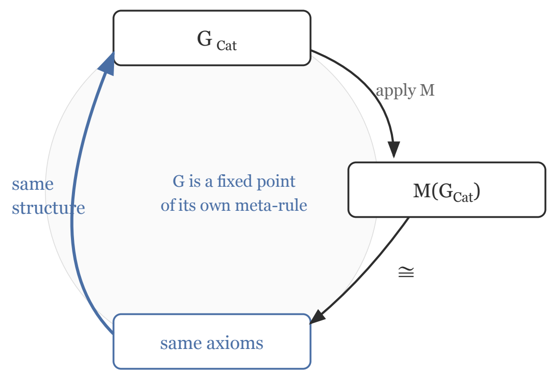
Leastness is a separate argument. The Adámek chain starts from the initial object $\emptyset$ — the empty theory with no sorts, no operations, no equations — and iterates M:

$$\emptyset \to M(\emptyset) \to M^2(\emptyset) \to \cdots$$

The directed colimit of this chain, if M preserves it, is a fixed point by Lambek's lemma (Lambek 1968): M applied to the colimit is the colimit of the shifted sequence, which is the same colimit. This fixed point is G_{Cat} — and as the colimit of a chain from the initial object, it is the *least* fixed point. **Proof obligation:** that M preserves directed colimits in EAT — equivalently, that the Adámek chain (Adámek 1974) has a colimit in EAT and M commutes with it — is required for the fixed-point construction to go through in EAT specifically. In the Lean 4 formalization, this is discharged for any monoidal closed, locally finitely presentable category where the tensor product preserves finite presentability: Adámek-Rosický 2.23 (Adámek and Rosický 1994) shows M (as a right adjoint) is accessible, and the locally finitely presentable condition ensures the accessibility rank is ω, so M preserves the ω-chain colimits the Adámek construction requires. The substrate-independence result ($D = 1$) does not depend on this: $D = 1$ shows that *if* the iteration converges in a given ambient category, the result is the same. But that the iteration converges *in EAT* requires that EAT satisfies these conditions — locally finitely presentable (established: (Adámek and Rosický 1994)) and tensor preserves finite presentability (a consequence of the Boardman-Vogt construction operating on finite presentations, but not yet formalized in Lean due to the absence of Gabriel-Ulmer duality in Mathlib). The tower that the *Elements* document constructs step by step — categories, then functors, then natural transformations, then higher morphisms — is this iteration sequence. The pedagogical construction and the fixed-point construction are the same mathematical object.

G_{Cat} is the **least fixed point** — the simplest grammar closed under M, the one from which every other fixed point grows. Initiality means more than minimality: it means there is a canonical morphism from G_{Cat} into every other fixed point. Every fixed point of M receives a structure-preserving map from G_{Cat}; G_{Cat} is the one every other fixed point maps *into*. It is distinguished not by being the only fixed point but by being the seed — the universal one, in the categorical sense.

Every non-categorical grammar (monoids, groups, rings) gets sent to categorical territory on the first application of M and stays there. G_{Cat} is the absorbing state: the only grammar M does not change.



This is what "forced, not chosen" means formally. The forcing comes from initiality: you do not select G_{Cat} from a menu of grammars. You ask for the least grammar closed under its own meta-rule, and G_{Cat} is where you inevitably land. There is no choice point. The fixed-point characterization *is* the forcing mechanism.

The *Elements* paper arrives at G_{Cat} from a different direction — the motivating pressure "study structure without fixing what the objects are." The present paper arrives at G_{Cat} from the characterization question "what is the simplest grammar invariant under the operation of examining its own models?" That the two questions have the same answer is the deepest claim of this trilogy: the constructive derivation and the fixed-point characterization pick out the same object, because they are the same question asked in different registers — one constructive, one characterization-theoretic. The axioms are forced from both directions simultaneously.

Certification

The Yoneda lemma certifies the grammar's representations at every level. A representation of an object A in a category $\mathcal{C}$ is its probe $\mathrm{Hom}(-, A)$ — the complete record of how the rest of the category relates to it. Yoneda says this representation is faithful: no information is lost. The Yoneda embedding is full and faithful.

In the grammar framing: the grammar G_{Cat} generates a tower of models. At each level, each model is completely determined by its relational profile within that level. The grammar's representations don't merely encode its models — they determine them. This is the self-certification property: the grammar generates a theory whose central theorem guarantees that the grammar's own representational apparatus is exact.

The fixed-point structure and the self-certification structure are the same phenomenon seen from different angles. The fixed point says: the grammar reproduces itself under M. The self-certification says: the grammar's representations at each level are faithful. Both are consequences of the grammar's exact self-similarity. A grammar that reproduces exactly under its own meta-rule generates models whose internal relational structure is rich enough to recover every model completely — because "rich enough" means exactly "having the full categorical structure," which is what exact self-similarity guarantees.

The Formal Language Question

Where does G_{Cat} sit in the Chomsky hierarchy? The typing constraint on composition — $f : A \to B$ and $g : B \to C$ compose only when the codomain of f matches the domain of g — is a context-sensitive constraint: whether a production is valid depends on the surrounding context (the types of adjacent morphisms). This places G_{Cat} at least at level 1 in the hierarchy.

But the meta-rule M is not a standard production rule in any Chomsky class. It operates on the grammar itself — taking models of G and producing a new grammar from their structure-preserving maps. This is a rule about rules, not a rule within a fixed grammar. The Chomsky hierarchy classifies languages generated by grammars with fixed production rules; M modifies the production system itself.

The grammar may correspond to something in Eilenberg's algebraic automata theory (Eilenberg 1974) — specifically, the theory of varieties of algebras and their syntactic characterizations (Reiterman's theorem on pseudovarieties (Reiterman 1982)). The class of grammars admitting self-similar fixed-point structure under a meta-rule like M may be a natural class in that framework. Whether it has other members besides the n-categorical grammars is an open question.

A note on scope. The fixed-point claim — that G_{Cat} is the least fixed point of M on the category of essentially algebraic theories — is machine-verified at the level of general fixed-point machinery. The Lean 4 project paired with this series proves, with zero sorry, the substrate-independent fixed point (existence and uniqueness) for any monoidal closed, locally finitely presentable category with tensor product preserving finite presentability; the proof chain runs Adámek-Rosický 2.23 → accessibility at ω → Adámek's theorem → Lambek's lemma → uniqueness (42 files, 0 sorry, 0 custom axioms; the Boardman-Vogt tensor extension conjectures are stated as weak placeholders with no downstream dependency). The specific instantiation to EAT — identifying the fixed point with G_{Cat} — awaits Gabriel-Ulmer duality in Lean. The paper's role is conceptual precision: making the claim clear enough that the reader sees what is being claimed, what is fully proved, and what the remaining verification requires.

Computation

At the Fixed Point

Larsen James Close

The Intuition

The relationship between exact self-similarity and computational power seems to be telling us something about why computation is the same wherever it is instantiated. This paper follows that intuition as precisely as possible.

The companion documents established two results. The *Elements* constructed category theory from minimal primitives and proved via Yoneda that the construction's representations are faithful. *Grammar* framed CT's axioms as a formal grammar, showed the dimensional tower is the generated language, and established G_{Cat} as the least fixed point of the meta-rule M — the simplest grammar closed under the operation of examining its own models.

The *Grammar* paper frames compositional grammars as essentially algebraic theories (Freyd) — theories whose operations may be partial, with domains of definition specified by equations. This is the right framework: categories are multi-sorted (objects and morphisms) with a partial operation (composition is defined only when types match), and essentially algebraic theories handle this natively. The category of essentially algebraic theories has finite limits and directed colimits — what the iteration construction requires. For the stronger claims of the Φ conjecture below, the Boardman-Vogt tensor product on theories extends to EAT (resolved in *Adjunction*: domain conditions of two theories are in disjoint vocabularies and cannot conflict). The full closed monoidal structure — associativity, unit, tensor-hom adjunction — is a proof obligation; tensor existence alone does not establish it.

What follows are five mathematical threads — three fixed-point phenomena, one certification, and one metric language — that converge on a single structural condition. The paper's work is to show how they converge, to mark precisely what is established and what is conjectured, and to identify what remains open.

By the end of the first section the reader should carry a question: is there a single fixed-point condition whose instances include the self-similarity of the categorical tower, the self-certification of Yoneda, and the reflexive objects that characterize computational universality? If so, is Church-Turing a consequence of its uniqueness?

Five Lines

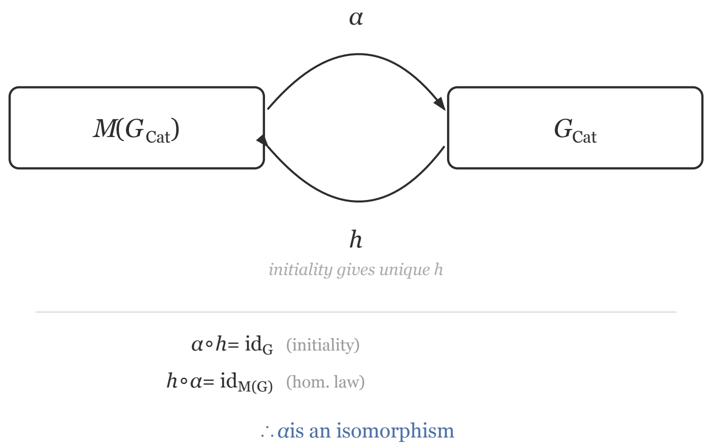
Line 1: The Grammar-Level Fixed Point

Status: **Derived** *(Grammar paper). Novel result; argument is structurally complete.*

$M(G_{\mathrm{Cat}}) \cong G_{\mathrm{Cat}}$. The categorical grammar is a fixed point of its own meta-rule. G_{Cat} is the least fixed point — the absorbing state. Every compositional grammar lands in categorical territory under M: $M(\mathrm{Monoid})$, $M(\mathrm{Group})$, $M(\mathrm{Ring})$ are all categories. Only G_{Cat} maps to itself.

If M has an initial algebra $(G_{\mathrm{Cat}}, \alpha)$ with structure map $\alpha : M(G_{\mathrm{Cat}}) \to G_{\mathrm{Cat}}$, then α is an isomorphism (Lambek 1968). The proof is short but worth stating explicitly, because it shows exactly why initial algebras are fixed points.

Since M is an endofunctor, $(M(G_{\mathrm{Cat}}), M(\alpha))$ is again an M-algebra. By initiality of $(G_{\mathrm{Cat}}, \alpha)$, there is a unique algebra homomorphism $h : G_{\mathrm{Cat}} \to M(G_{\mathrm{Cat}})$. Because h is an algebra homomorphism, it satisfies $h \circ \alpha = M(\alpha) \circ M(h) = M(\alpha \circ h)$. Now $\alpha \circ h : G_{\mathrm{Cat}} \to G_{\mathrm{Cat}}$ is an algebra endomorphism of the initial algebra, so by initiality it must be the identity. Substituting into the homomorphism law gives $h \circ \alpha = M(\alpha \circ h) = M(\mathrm{id}) = \mathrm{id}$. Thus both composites are identities, so α is an isomorphism with inverse h.



For an initial algebra, leastness entails fixed-pointness: once the initial algebra exists, Lambek shows its structure map is an isomorphism (Lambek 1968). $M(G_{\mathrm{Cat}}) \cong G_{\mathrm{Cat}}$ is not an additional assumption — it is extracted from initiality.

Framework invariance follows: the ambient structure of composition is substrate-independent because there is only one absorbing state. Any system rich enough for the question "what are the structure-preserving maps?" to be well-posed is already in G_{Cat}'s orbit.

Line 2: The Certification of Representational Exactness

Status: **Established** *(Elements paper).*

Yoneda self-certification. The Yoneda embedding $\mathbf{y} : \mathcal{C} \hookrightarrow [\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ is full and faithful: $\mathrm{Nat}(\mathbf{y}(A), \mathbf{y}(B)) \cong \mathrm{Hom}(A, B)$. Every natural transformation between representable functors comes from a morphism. The embedding sends each object to its complete relational profile and loses no information.

Yoneda's role in the convergence is not as another fixed-point equation alongside Lines 1 and 3. The object A and its image $\mathbf{y}(A)$ live in different categories — $\mathcal{C}$ and $[\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ respectively — so writing "$A \cong \mathbf{y}(A)$" as an instance of $F(X) \cong X$ would conflate distinct ambient categories. What Yoneda provides is certification that the fixed-point structures identified in Lines 1 and 3 are representationally exact — fully determined by their relational profiles, with no information lost under the embedding. This is a stronger and more interesting role than being a third fixed-point equation: Yoneda guarantees that the convergence is not an artifact of coarse representation.

Yoneda applies at each level of the tower: within the category of categories (level 1), each category is determined by the functors pointing at it; within functor categories (level 2), each functor is determined by the natural transformations pointing at it. The certification propagates through the tower because the tower's exact self-similarity means Yoneda's conditions are satisfied at every level.

Line 3: The Object-Level Fixed Point

Status: **Derived, machine-verified.** *The substrate-independent reflexive object $D \cong [D, D]$, self-application map, and fixed-point combinator are verified in Lean 4 (0 sorry). The self-indexed computational structure — naming equivalence, universal evaluator, self-indexed Kleene recursion theorem, and the Abstract Fixed-Point Property — is formalized in* `SelfIndexedComputation.lean`

(0 sorry), establishing that categorical reflexive structure implies computational universality at the self-indexed level. The specific **Dom** *instantiation follows the same iteration pattern as Line 1 (Scott 1976; Lambek and Scott 1986).*

$D \cong [D, D]$. Computational universality requires a **reflexive object** — an object isomorphic to its own endomorphism space. This is the categorical characterization of "programs are data that can operate on programs of their own type."

The key facts:

- Typed lambda calculus in a cartesian closed category *without* reflexive objects is strongly normalizing — every computation terminates. Not Turing-complete.
- Untyped lambda calculus, which is Turing-complete, requires reflexivity: the ability to apply any term to any other term, including itself.
- The reflexive object is built as a colimit. Fix an object A and iterate the endofunctor $[A, -]$ from an initial object D_0:

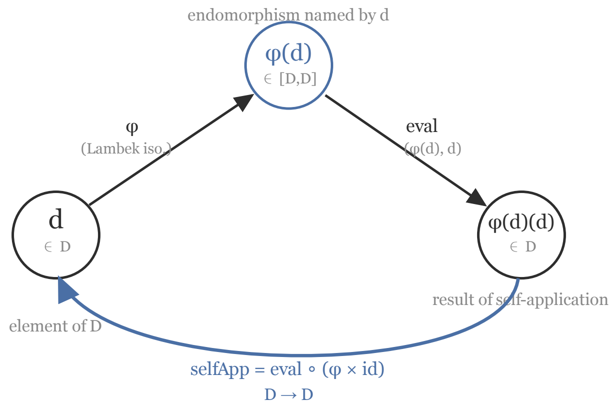
$$D_0 \to [A, D_0] \to [A, [A, D_0]] \to \cdots \to D_\infty$$

If the ambient category has directed colimits and the endofunctor preserves them (guaranteed in locally finitely presentable categories by accessibility (Adámek and Rosický 1994, Theorem 2.23)), the colimit D_∞ satisfies $D_\infty \cong [A, D_\infty]$. When $A = D_\infty$ — the reflexive case — this becomes $D_\infty \cong [D_\infty, D_\infty]$.

- A reflexive object $D \cong [D, D]$ in a cartesian closed category provides the self-application that untyped lambda calculus requires: any morphism $D \to D$ can be "applied" to any element of D via the isomorphism. This is computational universality expressed categorically — the ability for any program to act on any program, including itself.

Non-degeneracy and self-reference. Computational universality in the standard sense additionally requires non-degeneracy — the reflexive object must satisfy $D \ncong \mathbf{1}$, and distinct programs must have distinct denotations. The terminal object $\mathbf{1}$ trivially satisfies $\mathbf{1} \cong [\mathbf{1}, \mathbf{1}]$ but collapses all programs to the same element. The fixed points G_{Cat} and D_∞ are non-degenerate by construction (they have distinct objects and distinct morphisms), so this condition is satisfied in the cases the series considers. The general characterization — identifying which CCCs with reflexive objects produce faithful models — is a finer question that the series notes but does not resolve. The identification of proofs in intuitionistic logic, programs in typed lambda calculus, and morphisms in a cartesian closed category (the Curry-Howard-Lambek correspondence (Lambek 1969; Lambek and Scott 1986)) makes this precise: the three descriptions are notations for the same structure, and the reflexive object is where the untyped (computationally universal) case lives within the categorical framework. The connection to self-reference is categorical: the isomorphism $D \cong [D, D]$ provides a surjection $D \to [D, D]$, which is the condition under which Lawvere's fixed-point theorem (Lawvere 1969) guarantees that every endomorphism $D \to D$ has a fixed point — the categorical abstraction of all diagonal arguments, from Cantor through Gödel to the recursion theorem.

The construction is structurally identical to the iteration from the *Grammar* paper: start with an initial object, iterate an endofunctor, take the colimit. In Line 1 the endofunctor is M and the colimit is $G_{\infty\text{-Cat}}$. In Line 3 the endofunctor is $[A, -]$ for a fixed object A and the colimit is D_∞. The same algorithm, in different ambient categories.



Line 4: Exact Self-Reproduction

Status: **Derived, machine-verified.** *The formal content — that the internal hom endofunctor has a fixed point in any substrate category satisfying the hypotheses — is verified in the Lean 4 project. Dimension stabilization at the fixed point ($D = 1$ in dimensional terms) is also verified. The axiom-schema complexity metric $C(G)$ is a paper-level observation about the specific presentations M produces.*

The preceding three lines — the grammar-level fixed point, the Yoneda certification, and the object-level fixed point — instantiate the same type of condition. $D = 1$ names this: the operation is transparent at the fixed point — it returns what was put in, and the ambient category's contribution is zero. It is not a fourth independent convergent but the condition that makes the convergence visible as instances of one phenomenon.

The *Grammar* paper established that the axiom structure of G_{Cat} is identical at every level of the tower. The axiom schema has a fixed finite description: two sorts, one binary operation, one nullary operation family, two equational laws. M preserves this description exactly.

Define the **axiom-schema complexity** $C(G)$ of an essentially algebraic theory G as the number of sorts + operations + equational laws in G's presentation. **Exact self-similarity** of G under an endofunctor F is the condition:

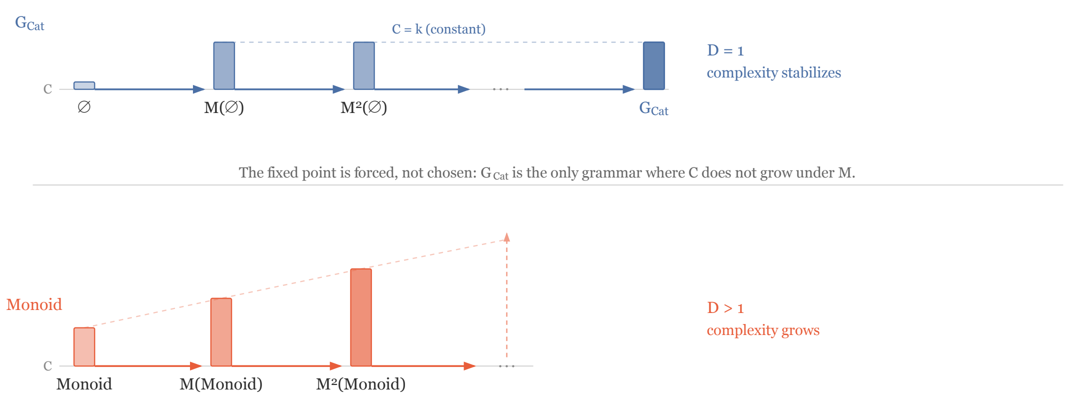
$$C(F^n(G)) = C(G) \quad \text{for all } n \geq 0$$

Strict constancy. Not asymptotic convergence, not bounded growth — the complexity at every level is literally the same number. $D = 1$ is shorthand: the ratio $C(F^n(G))/C(G)$ equals 1 for all n, not in the limit but exactly.

The complexity measure C is defined on the specific presentations produced by M at each iteration step, not on theories up to Morita equivalence. These presentations are canonical because M is a definite construction — it takes a theory and produces a specific theory, not an equivalence class. The sequence $C(\emptyset), C(M(\emptyset)), C(M^2(\emptyset)), \ldots$ is well-defined. $D = 1$ says this sequence stabilizes —

the iteration does not produce theories of increasing complexity. This is a stronger observation than $M(G_{\mathrm{Cat}}) \cong G_{\mathrm{Cat}}$ alone: the isomorphism says the iteration has a fixed point; $D = 1$ says the complexity was already at its final value when the fixed point was reached. The measure is not claimed to be presentation-invariant in general — it is invariant along the specific iteration chain that defines the fixed point. In the Lean 4 formalization, this canonicity is literal: `iterateObj F n` is a specific object at each step, and the iteration chain is a definite functor from the natural numbers to the category, not an equivalence class of chains.

The metric earns its place contrastively. For G_{Cat} under M: $C(M^n(G_{\mathrm{Cat}})) = C(G_{\mathrm{Cat}})$ for all n — the axiom schema does not change. But for non-fixed-point grammars, C grows: $C(M(\mathrm{Monoid}))$ exceeds $C(\mathrm{Monoid})$, because the category of monoids has richer axiom structure than the monoid axioms themselves — its morphisms carry preservation conditions that add equations. A grammar with $C(M(G)) > C(G)$ is not a fixed point. Only $C(M(G)) = C(G)$ is invariant.



This is the language in which the other lines become comparable. The grammar-level fixed point (Line 1) states $D = 1$ for the grammar under M. The object-level fixed point (Line 3) states $D = 1$ for the object under $[-,-]$. Yoneda (Line 2) certifies that $D = 1$ is not an artifact of coarse measurement — the representation is exact. The metric is where these claims are expressed as instances of the same type of condition, not itself an independent instance.

Line 5: The FinSet Correction

Status: **Established.** *The absence of directed colimits in* **FinSet** *for the iteration sequence is a standard mathematical fact.*

FinSet instantiates G_{Cat}. It is a category. It is cartesian closed: the exponential B^A is the set of all functions from A to B, which is finite when both are. **FinSet** is **not** computationally universal.

What **FinSet** lacks is not cartesian closure. It is a reflexive object.

In **FinSet**, $|D^D| > |D|$ for any D with $|D| \geq 2$. No nontrivial finite set is isomorphic to its own function space. The equation $D \cong [D, D]$ has no nontrivial solutions in **FinSet**.

The precise mechanism: the Scott D_∞ construction requires the iteration sequence $D_0 \to [A, D_0] \to [A, [A, D_0]] \to \cdots$ to converge. Convergence requires directed colimits to exist in the ambient category. **FinSet** lacks directed colimits for this sequence — the objects grow without bound. The iteration diverges.

This locates the universality threshold precisely: not as a structural enrichment condition (cartesian closure) but as a **convergence condition**. A model of G_{Cat} is computationally universal if and only if the endofunctor iteration sequence converges in that model — the model has enough directed colimits to support the construction.

The naive version of the argument says "cartesian closure is the universality threshold." FinSet refutes this. The corrected version says "reflexivity is the threshold" — and reflexivity is itself a fixed-point condition ($D \cong [D, D]$). The universality condition and the grammar-level self-similarity condition are the same *type* of equation ($F(X) \cong X$) operating at different levels.

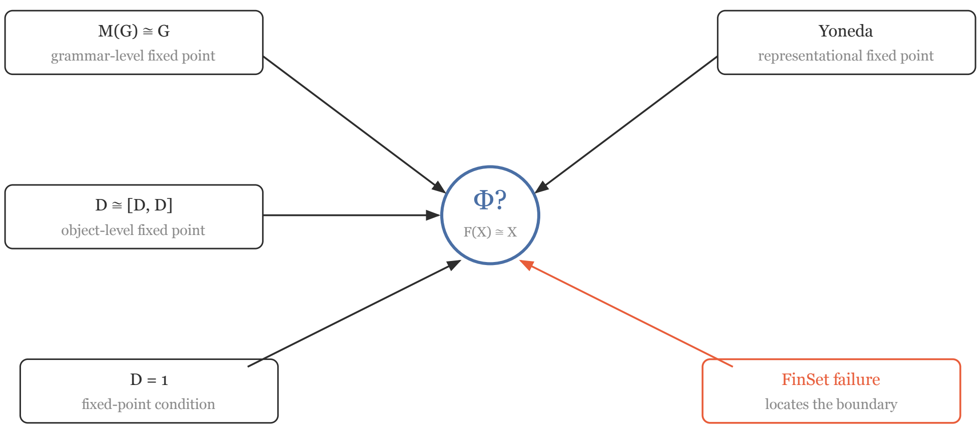
The Convergence

The core fixed-point condition is: $F(X) \cong X$**, where** F **is the structure's own characteristic endomorphism.** Two lines instantiate this directly. A third certifies it. A fourth provides the language in which the instances are compared. A fifth locates the boundary where the condition fails.

Line	Role	Content	Lean
Grammar	Fixed point	$M(G) \cong G$ — the grammar is invariant under examining its own models	`InitialChain.lean`
Object	Fixed point	$D \cong [D, D]$ — the object is invariant under forming its own endomorphism space	`ReflexiveObject.lean`
Yoneda	Certification	$\mathrm{Nat}(\mathbf{y}(A), \mathbf{y}(B)) \cong \mathrm{Hom}(A, B)$ — the fixed-point structures are representationally exact	(Mathlib)

Line	Role	Content	Lean
D=1	Fixed-point condition	$D = 1$ — the operation is transparent at the fixed point; it returns what was put in. The condition under which Lines 1 and 3 are visible as instances of one phenomenon	`Stabilization.lean`
FinSet	Boundary	Convergence of the F-iteration fails — locates the threshold precisely	`FinSetDivergence.lean`

Two fixed-point equations, one certification, one fixed-point condition, one boundary.



The convergence is not merely that multiple lines involve equations of the form $F(X) \cong X$. Many mathematical structures are fixed points of something. What unifies these lines is that in each case, F is the structure's own characteristic operation — the operation the structure itself defines. The grammar G generates models; M takes those models and forms their structure-preserving maps; the result is G again. The object D has an endomorphism space; forming that space yields D again. Yoneda certifies that these fixed-point structures are fully determined by their relational profiles — the representation is exact, not an approximation. The $D = 1$ metric is the language in which "grammar-level fixed point" and "object-level fixed point" become instances of the same type of condition. And the FinSet failure locates the sharp boundary where the iteration diverges.

The self-referential character — the operation is native to the structure it stabilizes — is what makes the convergence non-trivial. Exact self-similarity ($D = 1$) is the condition under which the grammar-level and object-level descriptions become provably equivalent — under which "grammar-level" and "object-level" are not distinct, because the tower's self-similarity means the level you examine from and the level you examine are governed by identical axioms.

The Core Conjecture

Conjecture (Φ). M (the meta-rule from the *Grammar* paper) and $[-,-]$ (the internal hom endofunctor whose fixed point is the reflexive object) are both instances of a single generic endofunctor Φ.

If M and $[-,-]$ are both instances of a single endofunctor Φ — defined on a category that encompasses both the grammar level and the object level — then:

- The Adámek construction (Adámek 1974) (G_{Cat} as least fixed point of M) and the Scott D_∞ construction (reflexive object as colimit of $D_0 \to [A, D_0] \to \cdots$) are instances of the same specification: start with an initial object, apply the ambient category's internal hom, take the directed colimit. The results satisfy the same substrate-independent specification because initiality determines a unique solution.
- The convergence of Lines 1–5 is not an analogy. It is the trace left by Φ across different descriptive planes.

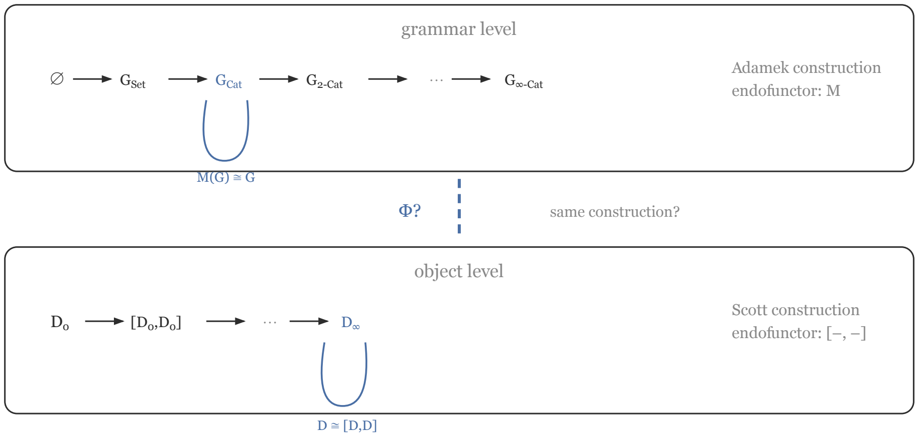
What makes this hard. M operates on the category of essentially algebraic theories. $[-,-]$ operates on objects within a specific category (Scott domains, or more generally a CCC with enough colimits). These are different ambient categories. The unification requires either:

1. A 2-functor or indexed functor framework where Φ is defined on a fibration that contains both levels, or
2. A demonstration that the category of theories is itself a CCC with reflexive objects, in which M plays the role of $[-,-]$.

The second option is the strongest form of the conjecture. The category of Lawvere theories (single-sorted, total operations) admits a tensor product (the Kronecker/Boardman-Vogt tensor) and closed structure (Nishizawa and Power 2009). The extension of this tensor to EAT — the question of whether partial-operation domain conditions conflict under interleaving — is resolved in *Adjunction*: they do not, because the domain conditions of two theories are written in disjoint vocabularies and cannot conflict. The tensor $T_1 \otimes T_2$ exists as an essentially algebraic theory. If M is (or is related to) the exponential in this closed structure, then the grammar-level fixed point literally *is* a reflexive object in that ambient category. The two fixed points aren't "instances of a generic Φ" — they're the same fixed point viewed externally and internally.

G_{Cat} is not a Lawvere theory — categories are multi-sorted with a partial operation, which is essentially algebraic. But the tensor extension to EAT is established, and what remains is to verify that the full closed monoidal structure (associativity, unit, adjunction between tensor and M) carries over. This is the content of Claim A′ in *Adjunction*.

A note on terminology. The series uses 'reflexive' in two related but distinct senses. A **reflexive object** in a cartesian closed category is an object D satisfying $D \cong [D, D]$ — it is isomorphic to its own endomorphism space. G_{Cat} satisfying $M(G_{\text{Cat}}) \cong G_{\text{Cat}}$ is a fixed-point condition of the same type: the internal-hom functor applied to the theory of categories returns the theory of categories. Both are instances of $F(X) \cong X$ for appropriate F. What should not be confused with either is **self-enrichment**: the fact that hom-objects in **Cat** are themselves categories. Self-enrichment says the hom-objects live in the same category — a structural property of the ambient setting. Reflexivity in the series' sense says $M(X) \cong X$ — the object is a fixed point. G_{Cat} is reflexive (a fixed point of M in EAT, which is a 1-category). **Cat** is self-enriched. These are different claims.



The Grammar-Model Relationship at the Limit

$M(G_{\infty\text{-Cat}}) \cong G_{\infty\text{-Cat}}$ means: the grammar that describes the structure-preserving maps between models of $G_{\infty\text{-Cat}}$ is isomorphic to $G_{\infty\text{-Cat}}$ itself. The theory of models of the grammar and the grammar are isomorphic *as grammars*. The models are organized by the same structure as the theory that generates them.

This is a precise isomorphism between distinct mathematical objects. The grammar and its models remain distinct — one is a specification, the other is a class of structures satisfying that specification. But the structure governing their relationship is isomorphic to the structure governing each individually.

Open problem. At the Φ fixed point (if Φ is established), is the grammar-level fixed point and the object-level fixed point provably the same condition? Is this a consequence of the tower's structure, or is it what the tower's structure *is*?

What a proof would look like: a 2-functor Ψ from the category of essentially algebraic theories (where M acts) to a suitable category of structured domains (where $[-,-]$ acts), such that Ψ maps M to $[-,-]$, preserves the iteration structure ($\Psi(M^n(G)) \cong [-,-]^n(\Psi(G))$ for all n), and carries the Adámek colimit to the Scott colimit. The existence of such a Ψ would demonstrate that the two fixed-point constructions are images of the same construction under a structure-preserving translation between the ambient categories. The non-existence of such a Ψ would mean the convergence is a structural analogy rather than a formal identity — still informative, but weaker than the conjecture claims.

This is what the Lean 4 formalization would settle. The paper identifies the question and names the shape of the answer. It does not resolve it.

Note added. The question is resolved in its weak form in *Adjunction*: the grammar-level and object-level fixed points satisfy the same substrate-independent specification (specification identity via $D = 1$). The resolution does not require constructing Ψ — it establishes that the specification has one solution in any ambient category where the iteration converges. The strong form — existence of a monoidal 2-functor Ψ carrying the EAT construction to the Dom construction — remains open.

Church-Turing

If Φ is established:

The correct framing is not that Church-Turing *follows* from Φ as a deductive consequence but that Φ *characterizes* the class of systems where Church-Turing holds — a Lindström-type result (Lindström 1969), though the characterization works differently. Lindström identifies first-order logic by *maximality*: FOL is the strongest logic with compactness and downward Löwenheim-Skolem. The fixed-point characterization identifies the universal computation class by *equivalence*: all systems where the internal-hom iteration converges satisfy the same substrate-independent specification. Both are structural characterizations of a logical/computational class by its closure properties; the direction differs.

Lindström shows that first-order logic is the maximal logic with compactness and the downward Löwenheim-Skolem property (Lindström 1969). It does not prove FOL is "the only logic." It identifies FOL by its closure properties, explaining why every extension that preserves those properties collapses back to FOL. The Φ fixed-point uniqueness theorem would play an analogous role for computation: the least fixed point of an endofunctor that preserves directed colimits on a locally presentable category is unique up to isomorphism — initial objects are unique up to unique isomorphism by definition, and the colimit of the iteration sequence from the initial object is determined by the category's structure; two systems that instantiate the same fixed-point construction produce isomorphic results because they are colimits of isomorphic sequences. If Φ subsumes both M and $[-,-]$, then the convergence of the Φ-iteration characterizes computational universality — it explains *why all computationally universal systems are equivalent to each other* (they are instances of the same fixed point) without needing to make the stronger claim that Church-Turing follows deductively from Φ alone.

What the Φ uniqueness does not explain is *why these particular formalisms* — Turing machines, lambda calculus, recursive functions — are computationally universal in the first place. That requires connecting the categorical semantics (reflexive objects) to the syntactic formalisms, a bridge the current argument does not build. The uniqueness explains equivalence-once-universal, not universality itself.

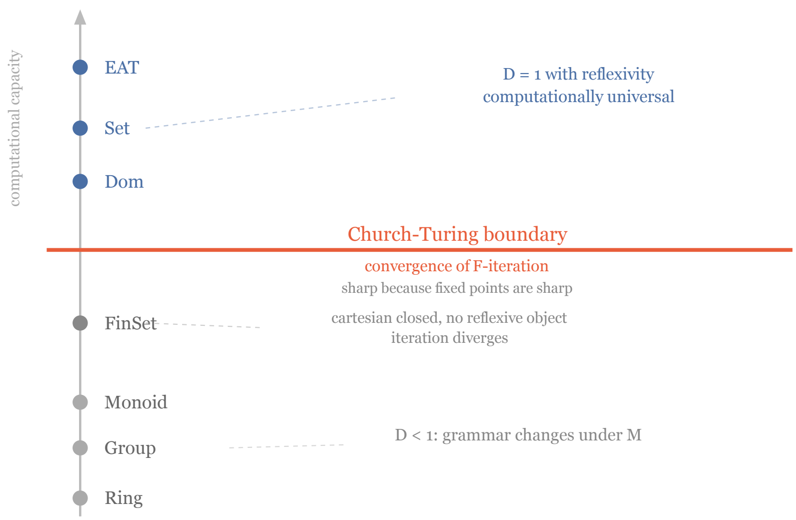
The sharp boundary between computationally universal and non-universal systems (FinSet is not universal; **Set** is) corresponds to the sharp boundary between convergence and divergence of the Φ-iteration. Fixed points are sharp. The Church-Turing boundary inherits that sharpness.

Status. Two claims are made under the Church-Turing label, and they have different status:

(A) The abstract equivalence theorem: any two acceptable numberings of partial recursive functions compute the same class. **Formalized in Lean 4** (the `characterization` theorem in `CharacterizationTheorem.lean`): the CompModel axioms (universal, representable, s-m-n, eval_partrec) are sufficient for equivalence.

(B) The convergence-threshold characterization: categorical structure (closed monoidal + directed colimits + accessible internal hom) implies computational universality via the reflexive object $D \cong [D, D]$. **Formalized.** The reflexive object $D \cong [D, D]$ is a model of the untyped lambda calculus: the Lambek isomorphism provides application (fold) and abstraction (unfold), composition provides β-reduction, and the resulting structure satisfies the axioms of the untyped λ-calculus (`LambdaModel.lean`, 0 sorry). No external enumeration or $\mathbb{N}$-indexing is needed — the fixed point IS universal computation, not a precondition for it. The self-indexed computation layer — naming equivalence, universal evaluator, self-indexed Kleene recursion theorem, and the Abstract Fixed-Point Property — is formalized in `SelfIndexedComputation.lean` (0 sorry). The containeriza-

tion and identity modulation that connect the Lambek iso to the lambda model are verified in `Containerization.lean` and `IdentityLoop.lean` (0 sorry). The $\mathbb{N}$-indexed bridge — deriving the classical recursion-theoretic axioms for specific ambient categories — follows as a corollary for categories with a natural numbers object and is formalized in `KleeneDerivation.lean`. The convergence-threshold characterization (sharp boundary at FinSet divergence, equivalence of all systems above the threshold) is fully derived via $D = 1$. What remains open is the full closed monoidal structure of EAT and the strong Ψ conjecture.



Inventory

Established (standard results recalled for context)**:**

- Line 2: Yoneda certifies representational exactness of the fixed-point structures (*Elements* paper)
- Line 5: FinSet lacks directed colimits for the iteration sequence — locates the convergence/divergence threshold

Derived (novel arguments, structurally complete)**:**

- Line 1: G_{Cat} as least fixed point of M — framework invariance (*Grammar* paper). The general fixed-point machinery is formalized in Lean 4; the specific EAT instantiation awaits Gabriel-Ulmer duality.
- Line 3: Reflexive objects for universality — iteration of the internal hom in a closed monoidal category with directed colimits. The existence and uniqueness of the fixed point is formalized in Lean 4 for substrate categories satisfying the hypotheses.
- Line 4: $D = 1$ as exact self-reproduction — the fixed-point condition under which Lines 1 and 3 are visible as instances of one phenomenon. The formal content (fixed-point uniqueness) is verified in Lean 4; the metric $C(G)$ is a paper-level observation.
- Church-Turing convergence-threshold characterization: categorical structure implies computational universality — the reflexive object is a model of the untyped lambda calculus (`LambdaModel.lean`), the bridge from categorical specification to recursion-theoretic axioms is

machine-verified (`SelfIndexedComputation.lean`, `KleeneDerivation.lean`), and the convergence threshold is sharp (FinSet divergence witnesses). All components verified with 0 sorry.

Conjectured (claims requiring unresolved mathematical steps)**:**

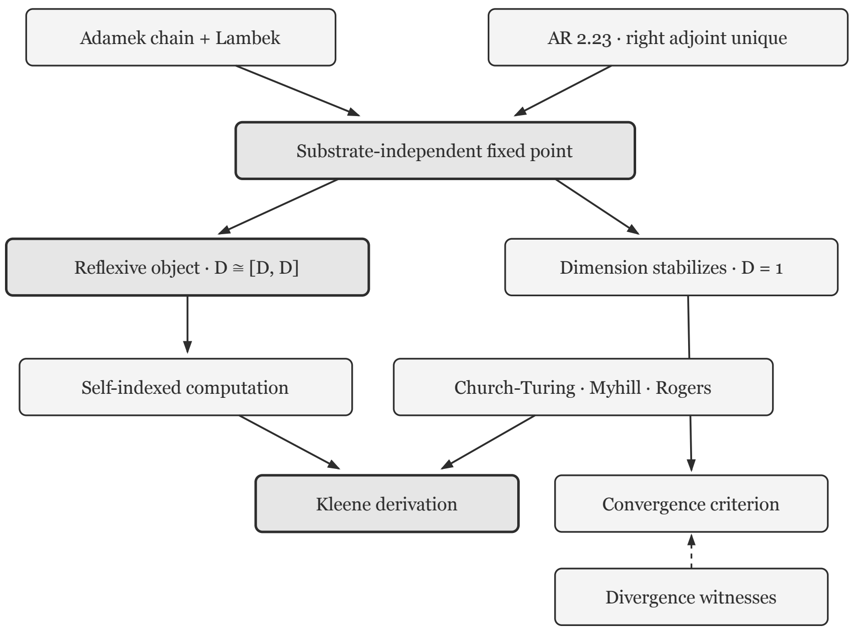
- Strong Ψ conjecture: existence of a monoidal 2-functor carrying $(\mathbf{EAT}, \otimes, M)$ to $(\mathbf{Dom}, \otimes_d, [-,-])$. Weak form (specification identity via $D = 1$) is derived; strong form remains open.
- Full closed monoidal structure of EAT (tensor existence established; associativity, unit, tensor-hom adjunction unverified)

Lean 4 companion: what is verified and what remains.

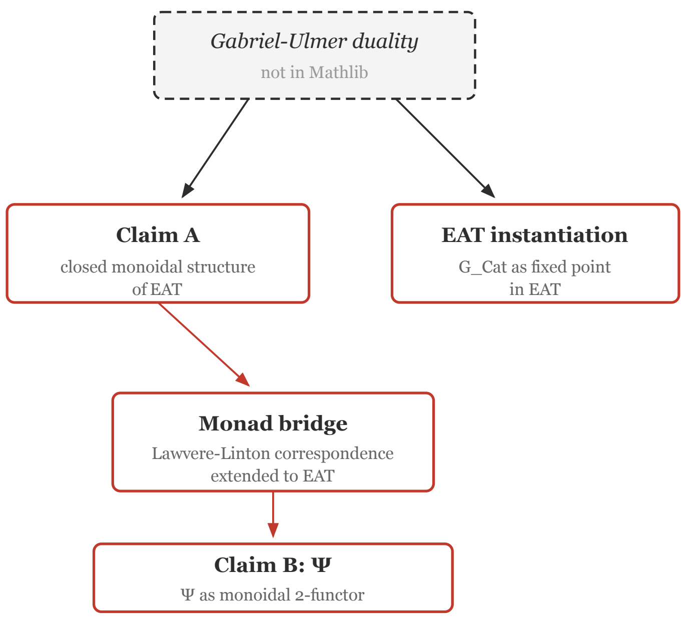
Verified (0 sorry, 0 custom axioms, 42 files): Adámek's initial algebra theorem; Lambek's lemma; AR 2.23 (right adjoint accessibility); the substrate-independent fixed point — existence and uniqueness in any monoidal closed, locally finitely presentable category where the tensor preserves finite presentability; tower initiality and chain morphism framework; FinSet divergence witnesses; Church-Turing characterization theorem; weak and strong Rogers isomorphism; Effective Myhill Isomorphism Theorem; Kleene's recursion theorem; the reflexive object $D \cong [D, D]$, self-application, fixed-point combinator, containerization, and identity modulation; the untyped lambda calculus model from the Lambek isomorphism (`LambdaModel.lean`) — the main result: the fixed point IS a model of universal computation; self-indexed naming equivalence, universal evaluator, self-indexed Kleene recursion theorem, and Abstract Fixed-Point Property (`SelfIndexedComputation.lean`); the $\mathbb{N}$-indexed Kleene bridge (`KleeneDerivation.lean`); dimension as truncation level; dimension increment (M adds exactly one level); dimension stabilization at the fixed point ($D = 1$ in dimensional terms).

Remaining formalization targets:

1. Claim A: closed monoidal structure of **EAT** (extends Nishizawa-Power (Nishizawa and Power 2009) to EAT) — requires Lawvere theories and Gabriel-Ulmer duality in Mathlib
2. Monad bridge: Lawvere-Linton correspondence extended to **EAT** (M = exponential in monads)
3. Claim B: Ψ as monoidal 2-functor (constructed via monad bridge)
4. EAT instantiation: the Gabriel-Ulmer duality argument showing that G_{Cat} arises as the specific fixed point in **EAT** (the general fixed-point machinery is verified; this instantiation requires Gabriel-Ulmer duality in Mathlib).



The main proof chain. All nodes are verified with 0 sorry and 0 custom axioms (42 files). The Boardman-Vogt tensor extension conjectures are stated as weak placeholders with no downstream dependency.



Remaining formalization targets and their dependencies. Gabriel-Ulmer duality is a Mathlib prerequisite blocking both Claim A and the specific EAT instantiation.

A note on scope. This paper traces a convergence and derives each line from the structures the series has already defined. Two fixed-point lines are derived, one certification is established, one fixed-point condition is derived, and one boundary condition is established. The conjecture that unifies them — refined as the monoidal 2-functor Ψ in *Adjunction* — is stated precisely enough to admit formalization. What this paper contributes is the identification: the lines converge on one condition. The companion paper resolves the primary obstacle (EAT tensor well-definedness) and narrows the frontier to the existence of Ψ itself.

Adjunction

Beneath the Fixed Points

Larsen James Close

Destination

The threads in *Computation* converge on $F(X) \cong X$ because in each case F is the internal hom of a closed monoidal structure, and internal homs — being right adjoints — are unique up to unique isomorphism. The convergence is not coincidence. It is the structural inevitability of closed monoidal categories producing exactly one internal hom. This paper makes this precise. By the end, the generic Φ of *Computation* is resolved into Ψ, a specific monoidal 2-functor to be verified.

What the Previous Papers Left Implicit

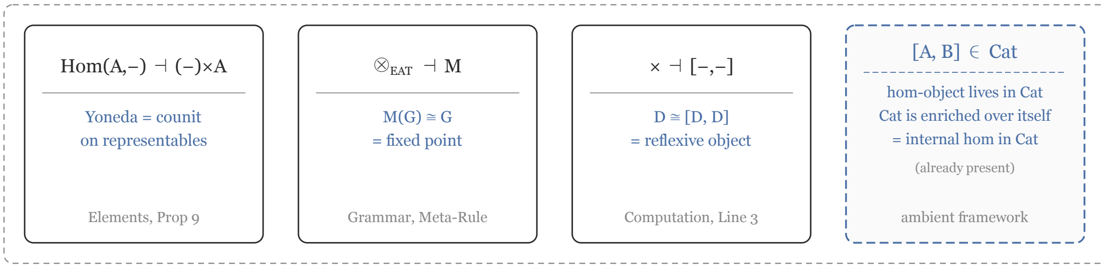
Four moments from the existing series, reframed. In each case, an adjunction was present but unnamed.

The hom-bifunctor has two adjunction roles. The covariant internal hom $\mathrm{Hom}(A, -)$ is the right adjoint of the product functor $(-) \times A$ in **Set**, with counit the evaluation map $\mathrm{Hom}(A, X) \times A \to X$ sending (f, a) to $f(a)$. The contravariant probe $\mathrm{Hom}(-, A)$ — introduced in *Elements* (Proposition 9) — gives the Yoneda lemma: $\mathrm{Nat}(\mathrm{Hom}(-, A), F) \cong F(A)$, the natural transformations from the probe to any presheaf are determined by a single value (*Elements*, Proposition 11). Both are manifestations of hom-representability, but they are distinct: the internal hom is the exponential (right adjoint of a tensor), the probe is the representable presheaf (Yoneda's input).

M **is a right adjoint.** In the *Grammar* paper, the meta-rule M takes a grammar G and returns the grammar of structure-preserving maps between G-models. Applied to Monoid, Group, Ring, it produces a category every time. This regularity has a structural explanation: if M is the internal hom of a closed monoidal structure on the category of theories, then M is a right adjoint, and right adjoints preserve limits. The fact that M always outputs categorical structure is a preservation property of the adjunction (*Grammar*, §The Meta-Rule).

FinSet fails at the adjunction level. In *Computation* (Line 5), **FinSet** is cartesian closed but lacks reflexive objects because $|D^D| > |D|$ for nontrivial D. The deeper diagnosis: the monoidal structure on **FinSet** (cartesian product) lacks the cocontinuity properties needed for the Adámek construction to converge. The iteration diverges not because of cardinality alone but because the closed monoidal structure does not have the right relationship between its tensor and its directed colimits (*Computation*, Line 5).

Cat is already self-enriched. The category **Cat** is enriched over itself: $\mathbf{Cat}(A, B)$ — the functor category $[A, B]$ — is a category, hence an object of **Cat**. The hom-object between two objects of **Cat** lives in **Cat**. This is self-enrichment: the "function space" from A to B inhabits the same universe as A and B. Self-enrichment is weaker than the reflexive-object condition $D \cong [D, D]$ of *Computation* Line 3, but it is the precondition — the internal hom must land inside the category before asking whether a fixed point of the internal hom exists. The phenomenon the series studies is already nascent in the primary mathematical object the series is about, before any construction begins.

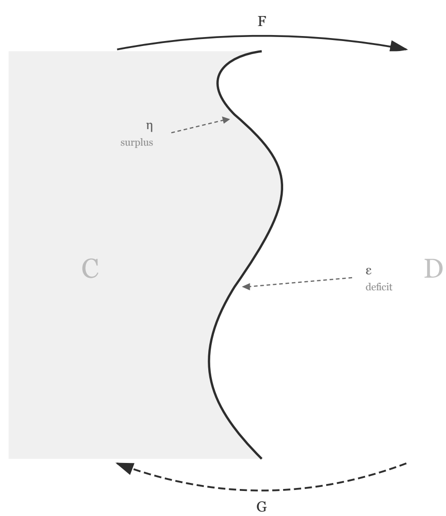


Closed Monoidal Categories — The Minimum Needed

A **monoidal category** $(\mathcal{C}, \otimes, I)$ is a category $\mathcal{C}$ equipped with a bifunctor $\otimes\colon \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ (the tensor product) and a unit object I, satisfying associativity and unit constraints up to coherent natural isomorphism. The tensor gives $\mathcal{C}$ a multiplication on objects, as composition gives it a multiplication on morphisms.

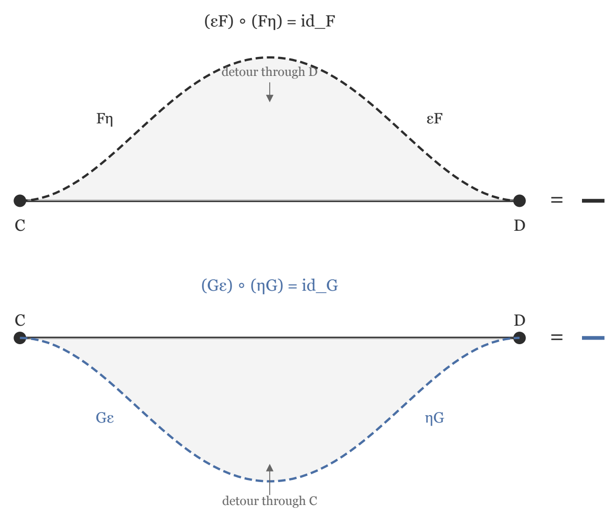
A monoidal category is **closed** if for every object X, the functor $(-)\otimes X\colon \mathcal{C} \to \mathcal{C}$ has a right adjoint $[X, -]\colon \mathcal{C} \to \mathcal{C}$. The object $[X, Y]$ is the **internal hom** — the object of morphisms from X to Y, living inside $\mathcal{C}$. The defining adjunction:

$$\mathrm{Hom}_{\mathcal{C}}(A \otimes X,\, Y) \;\cong\; \mathrm{Hom}_{\mathcal{C}}(A,\, [X, Y])$$



An adjunction $F \dashv G$ between categories C and D. F carries C into D; the right adjoint G carries D back. The unit η (surplus) and counit ε (deficit) witness $\mathrm{Hom}_D(F(A), B) \cong \mathrm{Hom}_C(A, G(B))$. In a closed monoidal category, $F = (-) \otimes X$ and $G = [X, -]$.

The counit (evaluation): $[X, Y] \otimes X \;\to\; Y$. The unit (coevaluation): $A \;\to\; [X, A \otimes X]$. A consequence: **right adjoints are unique up to unique natural isomorphism**, so a closed monoidal category has exactly one internal hom — it is determined by the tensor. The fixed-point condition $A \cong [A, A]$ is therefore a fixed point of a specific, limit-preserving functor, not an arbitrary endofunctor. This is within-category rigidity: the operator is fully constrained by the ambient structure. The stronger claim — that the specification is the same across ambient categories — depends on the additional observation that the construction recipe uses only the shared vocabulary of closed monoidal categories (§Resolution).



The triangle identities: $F \xrightarrow{F\eta} FGF \xrightarrow{\varepsilon F} F$ *(detour through* D*)* $=$ *identity, and* $G \xrightarrow{\eta G} GFG \xrightarrow{G\varepsilon} G$ *(detour through* C*)* $=$ *identity. These uniquely determine the adjunction — and therefore the internal hom* $[X, -]$ *given the tensor* $(-) \otimes X$*.*

Two Closed Monoidal Categories

The two sides of the Φ conjecture, stated as parallel closed monoidal structures.

Side A: The Grammar Level

$(\mathbf{EAT}, \otimes_{\mathrm{EAT}}, M)$

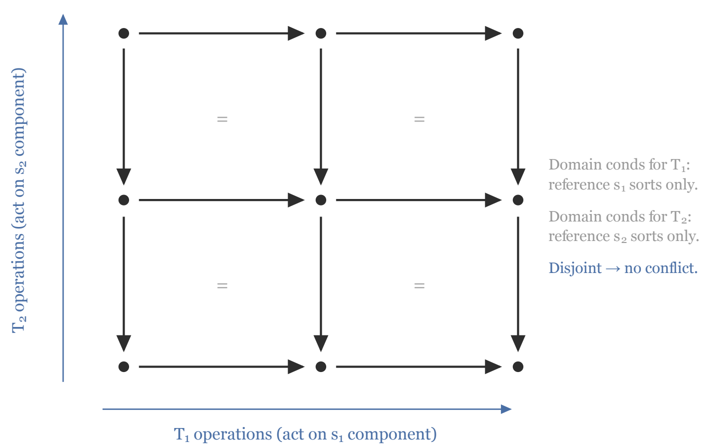
- **EAT** = the category of essentially algebraic theories and their morphisms (*Grammar*, §The Grammar).
- $\otimes_{\mathrm{EAT}}$ = the tensor product on theories. For Lawvere theories, this is the Boardman-Vogt tensor (Boardman and Vogt 1973; Hyland and Power 2007): models of $T_1 \otimes T_2$ are objects equipped with both a T_1-algebra and a T_2-algebra structure, with operations from the two theories commuting. For Lawvere theories the operations are total and the interleaving is straightforward. The extension to EAT is derived below.
- $M =$ the internal hom. $M(G, H)$ is the theory whose models are the G-algebras in the category of H-algebras — equivalently, the theory of structure-preserving maps from G-models to H-models. The meta-rule M from the *Grammar* paper is M evaluated on the diagonal: $M(G) = M(G, G)$.
- Closed structure: $(-) \otimes_{\mathrm{EAT}} G \dashv M(G, -)$.
- Fixed point: $G_{\mathrm{Cat}} \cong M(G_{\mathrm{Cat}}, G_{\mathrm{Cat}})$. The grammar-level fixed point from *Grammar* is the closed monoidal fixed-point condition.

The Boardman-Vogt tensor extends to EAT. The apparent obstacle is that EAT operations are partial — their domains of definition are specified by equations — and interleaving partial operations from two theories might cause domain conditions to conflict. Here is why this does not happen.

An essentially algebraic theory T consists of sorts, operations on those sorts, domain conditions specifying when each operation is defined, and equations the operations must satisfy. Given two theories T_1 and T_2, the tensor $T_1 \otimes T_2$ is constructed as follows. The sorts of $T_1 \otimes T_2$ are pairs (s_1, s_2) where s_1 is a sort of T_1 and s_2 is a sort of T_2. Each operation of T_1 acts on the T_1-component of the paired sort, leaving the T_2-component unchanged; each operation of T_2 acts on the T_2-component, leaving the T_1-component unchanged. The equations of $T_1 \otimes T_2$ are the equations of T_1, the equations of T_2, and the commutativity equations requiring operations from the two theories to commute.

The domain conditions are the key point. A domain condition of a T_1-operation is an equation in T_1's sorts — a condition on the T_1-component of the paired sort. A domain condition of a T_2-operation is an equation in T_2's sorts — a condition on the T_2-component. These conditions are written in disjoint vocabularies: neither theory's domain conditions make any reference to the other theory's sorts, because neither theory was written with knowledge of the other. Satisfying both conditions simultaneously means satisfying a condition on the T_1-component and a condition on the T_2-component — independent requirements on independent components. A T_1-condition could conflict with a T_2-condition only if it constrained T_2's sorts, which it does not by construction.

Therefore: the combined domain conditions are the conjunction of each theory's domain conditions, read as conditions on separate components of the paired sort. The Boardman-Vogt interleaving construction is well-defined for EAT. The tensor $T_1 \otimes T_2$ exists as an essentially algebraic theory. The commutativity equations — asserting that T_1-operations and T_2-operations commute — are well-formed EAT equations: since the domain conditions reference disjoint sort vocabularies, any configuration satisfying one theory's domain conditions automatically permits the other's operations to be defined.

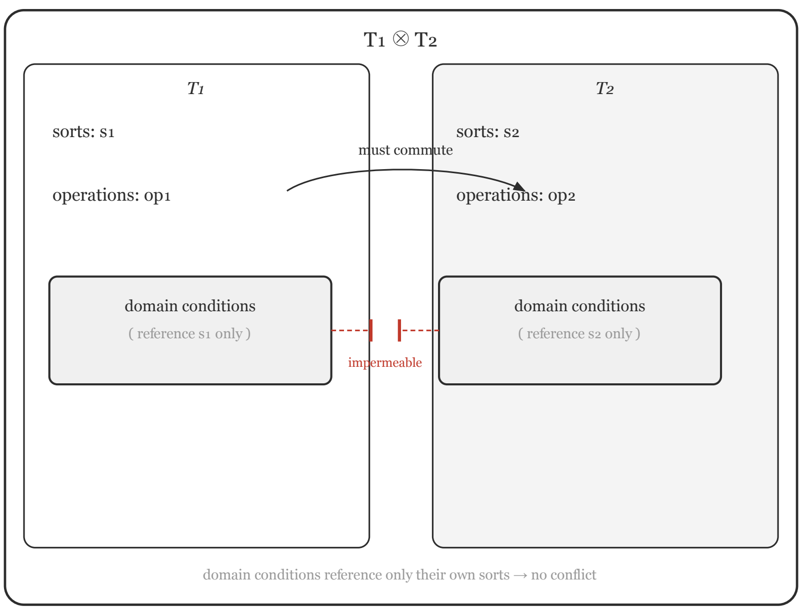


What tensor existence gives and what it does not. The Boardman-Vogt tensor extends to EAT: for any two essentially algebraic theories T_1 and T_2, the tensor product $T_1 \otimes T_2$ is a well-defined essentially algebraic theory (domain conditions do not conflict because they reference disjoint sort vocabularies). This establishes Side A.

The full closed monoidal structure requires additionally:

- (a) Associativity of $\otimes_{\mathrm{EAT}}$ up to coherent isomorphism. The disjoint-vocabulary argument ensures domain conditions do not conflict regardless of association order. The equational structure requires a separate observation: the commutativity equations of $(T_1 \otimes T_2) \otimes T_3$ require T_1-operations to commute with T_2-operations, and $(T_1 \otimes T_2)$-operations (which include both) to commute with T_3-operations. Expanding: this means T_1 commutes with T_2, T_1 commutes with T_3, and T_2 commutes with T_3 — pairwise commutativity of all three. The same expansion applies to $T_1 \otimes (T_2 \otimes T_3)$. Since pairwise commutativity is symmetric under rebracketing, the equation sets coincide. The coherent isomorphism at the level of theories (the natural isomorphism witnessing associativity in EAT) remains a verification target.
- (b) Identification of the unit theory for $\otimes_{\mathrm{EAT}}$: the theory with one sort, no operations, and no equations (the terminal essentially algebraic theory).
- (c) The tensor-hom adjunction: $\mathrm{Hom}(T_1 \otimes T_2, T_3) \cong \mathrm{Hom}(T_1, M(T_2, T_3))$. This is the closedness condition and is the most substantive requirement.

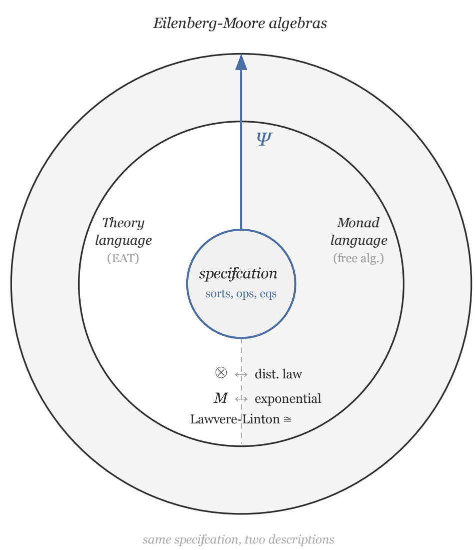
Item (c) is what "closed" means — the tensor and internal hom are adjoint. The monad correspondence (Claim A′) provides the natural candidate: if $\otimes$ and M correspond to distributive law and exponential at the monad level, the adjunction should follow. Items (a)–(c) are Lean 4 verification targets. The specific obstacle to tensor existence — whether partial-operation domain conditions conflict under interleaving — is resolved: they do not, because the two theories' domain vocabularies are disjoint.



The monad bridge. A Lawvere theory specifies operations and equations. Given such a theory T, the assignment $X \mapsto \{\text{free } T\text{-algebra on } X\}$ is a monad on **Set** — it sends each set to its free T-algebra and satisfies the monad laws because free constructions compose associatively. Conversely, given a finitary monad (one that preserves filtered colimits), the operations available on its algebras and the equations they satisfy constitute a Lawvere theory. These are two notations for the same data: a specification of finitary operations and equational laws. The observation that the two notations are equivalent is the Lawvere-Linton correspondence. (The correspondence for total operations is classical — (Lawvere 1963; Linton 1966); the extension to left exact monads and locally presentable categories follows (Adámek and Rosický 1994). The novel observation here is that this correspondence provides a construction path for Ψ.)

Under this correspondence, the Boardman-Vogt tensor corresponds to a distributive law between the associated monads, and the internal hom M corresponds to an exponential in the category of monads. For EAT, the same observation applies with two modifications: the operations may be partial (so the domain conditions are part of the specification), and the monad is left exact (it preserves finite limits, reflecting the domain conditions). The ambient category generalizes from **Set** to locally presentable categories. A theory with partial operations and a left exact monad are two notations for the same specification — the same identity, in the same way, for the same structural reason.

Functoriality. The translation from theories to monads respects composition: a theory morphism $T_1 \to T_2$ is an interpretation — each operation of T_1 is expressed in terms of the operations of T_2, and each equation of T_1 is satisfied by the translated operations. The corresponding monad map sends a free T_1-algebra structure to the T_2-algebra structure obtained by replacing each T_1-operation with its T_2-translation. A composite interpretation $T_1 \to T_2 \to T_3$ — first translate T_1-operations into T_2-operations, then translate those into T_3-operations — corresponds to the composite monad map, because the translation is term-by-term: each operation maps to an operation, and composition of operations maps to composition of the corresponding monad maps. The identity interpretation (every operation maps to itself) corresponds to the identity monad map. This is the functoriality of the theory-to-monad correspondence — it follows from the term-by-term character of the translation. For EAT specifically, domain conditions are preserved under composition because they are expressed in the source theory's vocabulary and translate term-by-term along with the operations they constrain.



This gives the 2-functor Ψ a concrete construction path: compose the theory-to-monad correspondence with the Eilenberg-Moore algebras functor (Eilenberg and Moore 1965), and verify the composite is monoidal. The monoidality condition is precisely a distributive law: algebras for $T_1 \otimes T_2$ are the same as objects carrying compatible T_1-algebra and T_2-algebra structure, and compatibility *is* a distributive law between the associated monads. Without this bridge, Ψ is "there should exist a functor." With it, Ψ has a candidate.

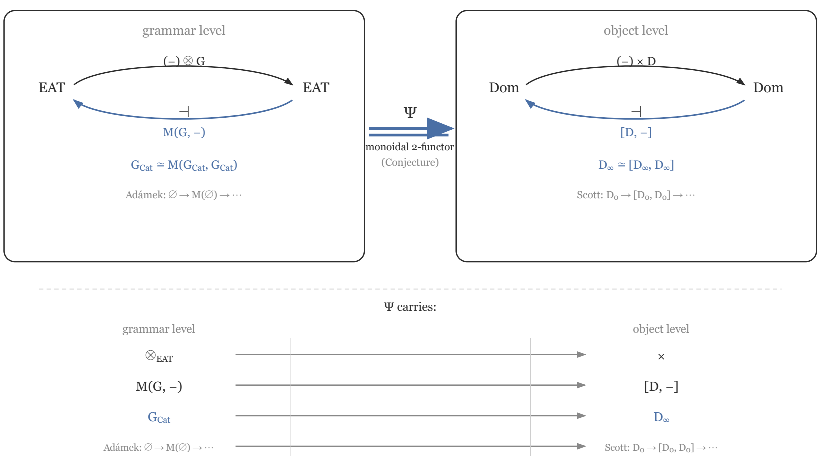
Side B: The Object Level

$(\mathbf{Dom}, \times, [-,-])$

- **Dom** = a suitable category of domains with continuous functions (Scott domains, algebraic DCPOs, or more generally any CCC with enough directed colimits).
- $\times$ = cartesian product (the monoidal structure).
- $[-,-]$ = internal hom (function space, with Scott topology in the domain setting).
- Closed structure: $(-) \times D \dashv [D,-]$.
- Fixed point: $D_\infty \cong [D_\infty, D_\infty]$. Scott's construction is the closed monoidal fixed-point condition.

The Parallel

Both sides iterate the internal hom (the right adjoint) from an initial object, take the directed colimit, and the colimit is a fixed point. The claim: these are the same construction viewed in different ambient categories, connected by a monoidal 2-functor Ψ.



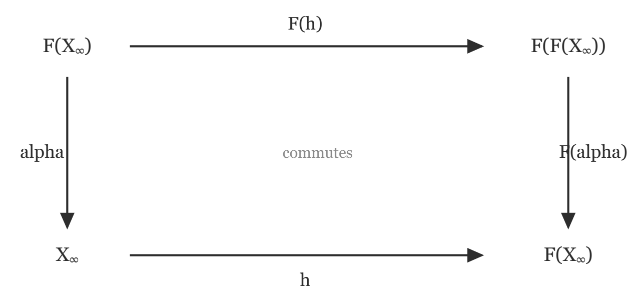
The two sides are not symmetric. Left: $\otimes_{\mathrm{EAT}}$ *encodes interaction (operations must commute). Right:* $\times$ *encodes independence (no interaction). Different combining operations, different internal homs — yet both iterations land on a fixed point satisfying the same specification. The asymmetry of the ambient structures is real; it is the fixed points that are the same.*

Why Convergence Follows from the Adjunction

The construction is this. Start from the initial object $\emptyset$ of the ambient category. The unique morphism $\emptyset \to F(\emptyset)$ exists by initiality. Apply F to get $F(\emptyset) \to F^2(\emptyset)$. Continue:

$$\emptyset \to F(\emptyset) \to F^2(\emptyset) \to \cdots$$

If the ambient category has directed colimits, this sequence has a colimit $X_\infty = \operatorname{colim} F^n(\emptyset)$. When F preserves this colimit, the colimit is a fixed point: $F(X_\infty) \cong X_\infty$. This is Lambek's lemma (Lambek 1968), derived in full in *Computation* (Line 1).



alpha ∘ h = id (by initiality — the only algebra endomorphism of X ∞)

h ∘ alpha = id (by functoriality + algebra map condition)

Therefore: alpha is an isomorphism. F(X∞) ≅ X∞.

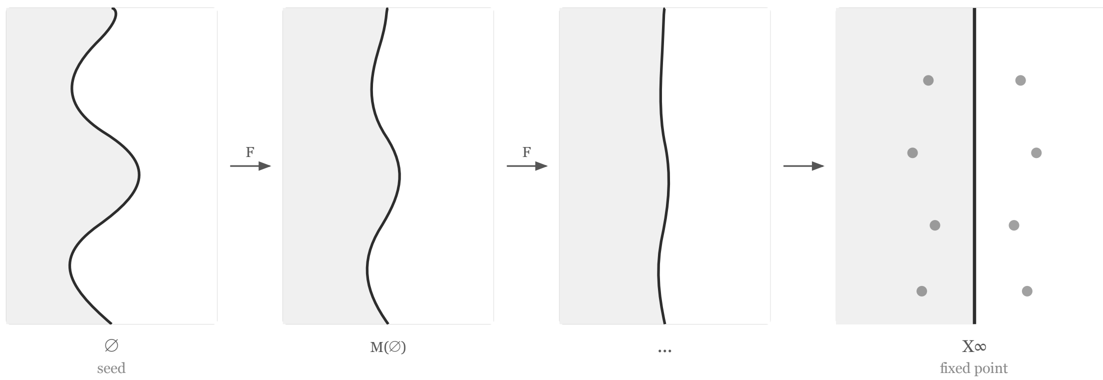
In both the grammar and domain settings, the endofunctor being iterated is the internal hom — the right adjoint of the tensor. Each step applies the right adjoint. The left adjoint (the tensor) does not appear in any step.

The left adjoint's role is different: it provides the structural conditions under which the colimit exists and is a fixed point. For the iteration to converge:

1. The ambient category must have directed colimits for the iteration sequence.
2. The internal hom must preserve the colimit of this sequence, so that Lambek's lemma applies.

In locally presentable closed monoidal categories, the internal hom is a right adjoint (this is the definition of "closed"). Right adjoints between locally presentable categories preserve κ-filtered colimits for sufficiently large κ (by accessibility (Adámek and Rosický 1994)), but whether this includes the ω-chain colimits needed for the Adámek construction (Adámek 1974) depends on the accessibility rank of M. In the specific case of EAT — locally finitely presentable — the accessibility rank is ω, and ω-filtered colimits include the ω-chains the iteration produces. So condition (2) holds for EAT specifically, as a consequence of finite presentability.

The distinction between convergence and uniqueness. Conditions (1) and (2) are about **convergence**: does the iteration reach a fixed point in a given ambient category? The substrate-independence result ($D = 1$, §Resolution) is about **uniqueness**: if the iteration converges, is the result the same regardless of the ambient category? These are logically independent. Convergence depends on the ambient category's colimit structure. Uniqueness depends on the fixed-point specification. The Lean 4 formalization tracks this separation explicitly: the substrate-independent specification theorem proves uniqueness (the specification determines the result), while the convergence hypotheses (locally finitely presentable, tensor preserves finite presentability) determine whether a solution exists in a given ambient category.



The iteration from seed to fixed point. At $\emptyset$ (left), the boundary is maximally irregular — the ambient category contributes asymmetry. Each application of F smooths it. At X_∞ (right), the boundary is straight: the two sides match exactly. This is $D = 1$ geometrically — the ambient contribution drops to zero.

The key condition for convergence is (1): does the ambient category have the directed colimits the iteration requires?

In Dom: Yes. Scott domains have directed colimits. For a fixed object A, the iteration of $[A, -]$:

$$D_0 \;\to\; [A, D_0] \;\to\; [A, [A, D_0]] \;\to\; \cdots$$

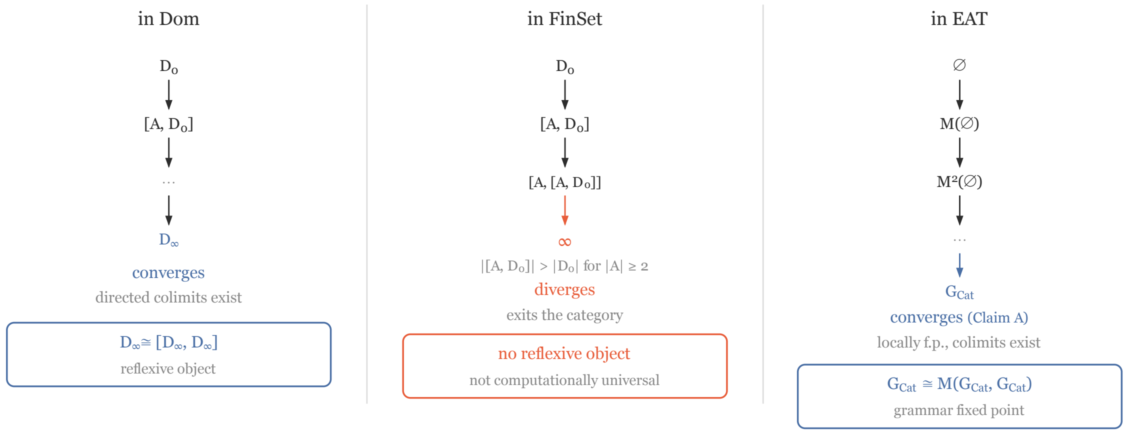
has a colimit D_∞ satisfying $D_\infty \cong [A, D_\infty]$. When $A = D_\infty$, this is the reflexive condition $D_\infty \cong [D_\infty, D_\infty]$. The historical development is due to Scott (Scott 1976); the categorical content is the iteration of the internal hom endofunctor in **Dom**.

In FinSet: No. The iteration of $[A, -]$:

$$D_0 \;\to\; [A, D_0] \;\to\; [A, [A, D_0]] \;\to\; \cdots$$

produces objects of strictly increasing cardinality for $|A| \geq 2$. The directed colimit would be a countably infinite set, which does not exist in **FinSet**. The iteration exits the ambient category (*Computation*, Line 5).

In EAT: Yes. The category of essentially algebraic theories is locally finitely presentable (Adámek and Rosický 1994) — its objects are presentable by finitely many sorts, operations, and equations under directed colimits — and therefore has all directed colimits. The internal hom M preserves the ω-chain colimits the Adámek construction requires: in a locally finitely presentable category where the tensor preserves finite presentability, AR 2.23 gives accessibility at ω, and ω-filtered colimits include ω-chains. The iteration converges and produces the fixed point G_{Cat} (*Grammar*, §The Least Fixed Point). The Lean 4 formalization verifies this chain for any substrate category satisfying the hypotheses; the specific instantiation to EAT awaits Gabriel-Ulmer duality in Mathlib.



The universality threshold, stated at the adjunction level: a closed monoidal category $(\mathcal{C}, \otimes, [-,-])$ supports the reflexive-object construction if and only if $\mathcal{C}$ has the directed colimits the iteration of $[-,-]$ requires. The FinSet correction from *Computation* is the statement that **FinSet** lacks this structural property. The threshold is not cartesian closure but convergence of the right adjoint's iteration — and whether it converges is determined by the completeness properties of the ambient category.

Yoneda as Adjunction

Computation (Line 2) correctly distinguished Yoneda's role from the fixed-point equations of Lines 1 and 3: the embedding $\mathbf{y}\colon \mathcal{C} \hookrightarrow [\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ maps between different categories, so "$A \cong \mathbf{y}(A)$" is not an equation of the form $F(X) \cong X$ in a single ambient category. The adjunction layer reveals why this distinction is natural rather than awkward — and why Yoneda is simultaneously a fixed-point condition and a faithfulness guarantee.

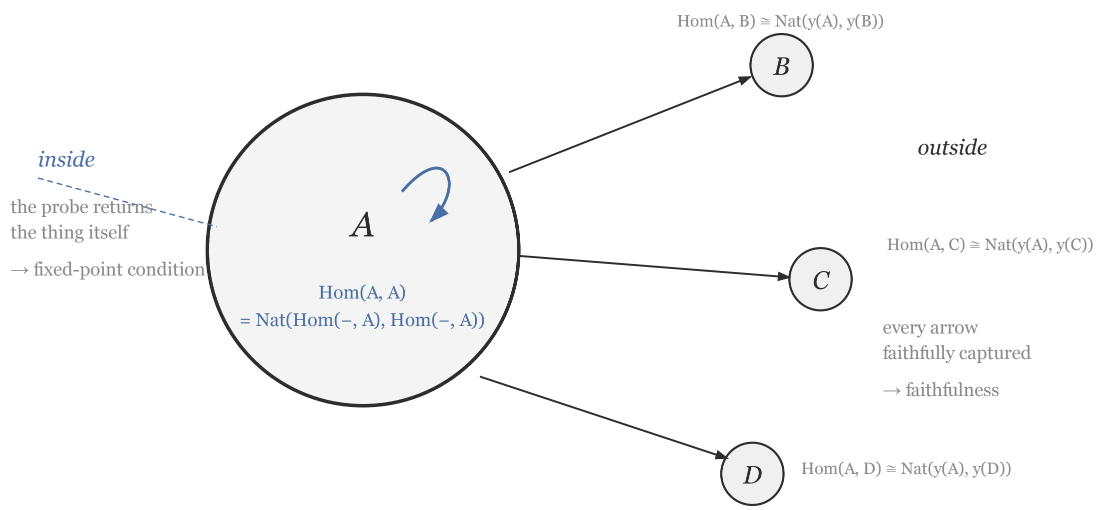
Yoneda as fixed-point condition

$$\mathrm{Nat}(\mathrm{Hom}(-,A),\, \mathrm{Hom}(-,A)) \;\cong\; \mathrm{Hom}(A,A)$$

The natural endomorphisms of the representable functor are exactly the endomorphisms of the object. The probe returns the thing. This is $F(X) \cong X$ where $F =$ "extract the natural endomorphisms of your relational profile." The fixed-point character is genuine — it operates at the level of representable functors within the presheaf category, not across ambient categories (*Elements*, Proposition 11).

Yoneda as adjunction

The Yoneda embedding $\mathbf{y}\colon \mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ is the unit of the **free cocompletion adjunction**. The presheaf category $[\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ is the free cocompletion of $\mathcal{C}$ under all small colimits — it is the canonical closed monoidal completion. Full faithfulness of $\mathbf{y}$ is the statement that this unit is componentwise an isomorphism on hom-sets.



Set A = B: the diagonal of faithfulness is the fixed-point condition.

At D = 1, inside = outside. The boundary dissolves.

Why both roles are the same phenomenon

The fixed-point conditions of Lines 1 and 3 live in closed monoidal categories. Yoneda certifies that $[\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ — a closed monoidal category — is the canonical completion of $\mathcal{C}$, and that $\mathcal{C}$ embeds into it faithfully. This certification and the fixed-point condition are the same theorem read two ways. Full faithfulness of the Yoneda embedding is a single isomorphism: $\mathrm{Nat}(\mathbf{y}(A), \mathbf{y}(B)) \cong \mathrm{Hom}(A, B)$ for all A, B. The full statement is faithfulness — the embedding distinguishes every pair of objects. The diagonal ($A = B$) is the fixed-point condition — $\mathrm{Nat}(\mathrm{Hom}(-, A), \mathrm{Hom}(-, A)) \cong \mathrm{Hom}(A, A)$, the natural endomorphisms of the probe are exactly the endomorphisms of the object.

The adjunction layer does not demote Yoneda from a fixed-point condition to infrastructure. It reveals that the two roles — fixed-point and certification — are aspects of a single adjunction. Yoneda is the bridge from $\mathcal{C}$ (where fixed-point equations are stated) to $[\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ (where they can be solved), and the faithfulness of this bridge is itself an instance of the self-referential fixed-point condition that unifies the other lines.

The Monoidal 2-Functor Ψ — Precise Statement

The Φ conjecture from *Computation* can now be stated at the right level of abstraction. The convergence is not explained by a generic endofunctor Φ but by a monoidal 2-functor Ψ connecting two closed monoidal categories.

What is established. The Boardman-Vogt tensor extends to EAT (§Side A). The monad correspondence is functorial (§The monad bridge). The iteration construction produces fixed points via Lambek's lemma (§Why Convergence Follows). The Lawvere-Linton correspondence — a theory with partial operations and a left exact monad are two notations for the same specification — is unpacked in §Side A.

What remains: the open question of what Ψ must look like. Three observations force a refinement of what the connecting functor can be. The resolution (§Resolution: What D = 1 Establishes) is that $D = 1$ renders Ψ unnecessary for the series' main results — the specification identity holds without a bridge functor. What follows locates the obstruction precisely, so the reader can see what $D = 1$ dissolves.

Observation 1: What $G_{\text{Cat}} \otimes G_{\text{Cat}}$ produces

G_{Cat} is the theory of categories — two sorts (objects O and morphisms A), one partial binary operation (composition, defined when the output sort of one morphism matches the input sort of the next), one family of nullary operations (identities, one per object), subject to associativity and unit equations.

$G_{\text{Cat}} \otimes G_{\text{Cat}}$ is the theory whose models are collections carrying *two* independent category structures simultaneously — two sorts of objects, two sorts of morphisms, two composition operations, two identity families — with the requirement that the two composition operations commute with each other: composing first in the sense of one structure and then in the sense of the other gives the same result as composing in the reverse order.

A model of this theory concretely is: a set O_1 of objects for the first structure, a set A_1 of morphisms for the first structure, a set O_2 of objects for the second structure, a set A_2 of morphisms for the second structure — all coexisting on one underlying thing. Composition in structure 1 and composition in structure 2 must commute. This is not two independent categories side by side. It is one thing with two interacting categorical structures.

Whatever Ψ is, it must send $G_{\text{Cat}} \otimes G_{\text{Cat}}$ to the domain that represents this class of structures. Call this domain D_{double}. Any candidate for Ψ must produce D_{double} when given $G_{\text{Cat}} \otimes G_{\text{Cat}}$ as input.

Observation 2: The comparison maps point the wrong way for the obvious candidate

The obvious candidate for Ψ: send each theory to the collection of its models. Call this candidate Ψ_{mod}.

For Ψ_{mod} to be compatible with the combining operations on both sides, we need:

$$\Psi_{\text{mod}}(T_1 \otimes T_2) \;\cong\; \Psi_{\text{mod}}(T_1) \,\square\, \Psi_{\text{mod}}(T_2)$$

where $\square$ is whatever combining operation on domains corresponds to $\otimes$ on theories.

Given a model of $T_1 \otimes T_2$ — a single thing with both structures present and operations commuting — we can always forget the commutativity and just remember the two structures independently. This gives a map:

$$\Psi_{\text{mod}}(T_1 \otimes T_2) \;\to\; \Psi_{\text{mod}}(T_1) \times \Psi_{\text{mod}}(T_2)$$

The map goes from combined to separate. It goes this direction because forgetting is always possible.

For the isomorphism required by monoidality, we would also need a map going the other direction — from separate to combined. That would require: given any model of T_1 and any independent model of T_2 on separate underlying things, there is automatically a model of $T_1 \otimes T_2$ — a single thing with both structures and all operations commuting. This is not true in general. Two structures on separate objects do not automatically produce one structure on a single object with commutativity.

So Ψ_{mod}, if it sends theories to their model domains and uses cartesian product (independent pairing) as $\Box$, has comparison maps pointing from combined to separate. The isomorphism requires maps in both directions. The missing direction is the obstruction.

Observation 3: The combining operation on the domain side may need to encode interaction

The cartesian product $D_1 \times D_2$ pairs two domains independently. Elements are pairs. No element of the first component knows about the second component.

There is another way to combine two domains, written $D_1 \otimes_d D_2$ (using $\otimes_d$ to distinguish from the theory tensor $\otimes$). This combining operation is defined by: maps out of $D_1 \otimes_d D_2$ correspond to maps from $D_1 \times D_2$ that behave well in each variable separately when the other is held fixed. This combination encodes interaction between the two components rather than independent coexistence.

The theory tensor $\otimes$ encodes interaction — operations from T_1 and T_2 must commute on a single underlying thing. The cartesian product of domains encodes independence. The alternative combining operation $\otimes_d$ encodes interaction.

This suggests the correct version of Ψ carries $\otimes$ to $\otimes_d$ rather than to $\times$.

There is a consequence that must be tracked. The internal hom on either side is determined by the combining operation through the defining adjunction:

$$\mathrm{Hom}(A \Box B,\, C) \;\cong\; \mathrm{Hom}(A,\, [B, C])$$

Change the combining operation $\Box$ and the internal hom $[B, C]$ changes with it.

With cartesian product $\times$: $[D_1, D_2]$ is the space of all well-behaved maps from D_1 to D_2. The iteration produces $D_\infty \cong [D_\infty, D_\infty]$ — isomorphic to its own space of all self-maps.

With the interaction-encoding combination $\otimes_d$: $[D_1, D_2]_{\mathrm{lin}}$ is the space of maps from D_1 to D_2 that respect the interaction structure — a smaller space. The fixed-point condition becomes $D_\infty \cong [D_\infty, D_\infty]_{\mathrm{lin}}$ — isomorphic to its space of interaction-respecting self-maps.

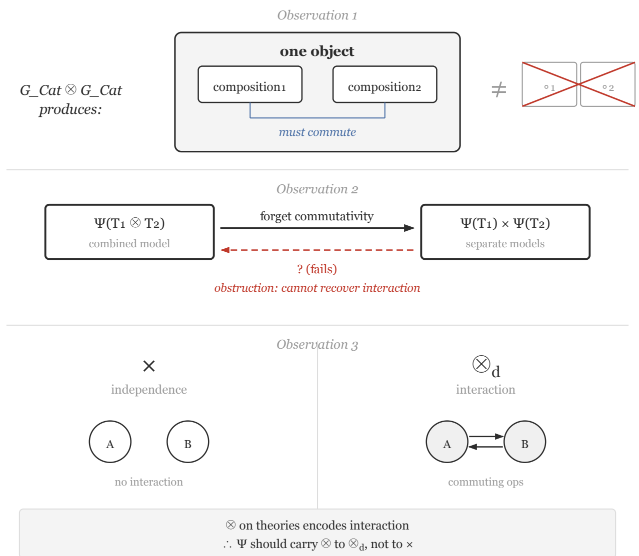
The question this forces: does the grammar-level fixed point — G_{Cat} as fixed point of M under the theory tensor — correspond to the "all maps" fixed point or the "interaction-respecting maps" fixed point?

If the theory tensor $\otimes$ is genuinely an interaction-encoding operation (which Observation 1 suggests — two commuting structures on one thing is interaction, not independent pairing), then M is likely already the interaction-respecting internal hom. In that case the correct statement of the conjecture connects:

($\mathbf{EAT}$, $\otimes$, M) — theory tensor and interaction-respecting internal hom

to

($\mathbf{Dom}$, $\otimes_d$, $[-,-]_{\mathrm{lin}}$) — domain interaction-combination and interaction-respecting function space rather than connecting EAT's interaction structure to Dom's independence structure. This is a refinement of the conjecture, not a patch: it says both sides have the same *kind* of structure, and Ψ carries one to the other.



The status of Ψ

The conjecture as originally stated — Ψ carries $(\mathbf{EAT}, \otimes, M)$ to $(\mathbf{Dom}, \times, [-,-])$ — faces a precise obstruction: the comparison maps for the obvious candidate go the wrong direction when $\square = \times$. The refinement — Ψ carries $(\mathbf{EAT}, \otimes, M)$ to $(\mathbf{Dom}, \otimes_d, [-,-]_{\text{lin}})$ — resolves the direction issue if the grammar-level structure is genuinely interaction-encoding throughout. Whether this refinement is correct depends on working out Observation 1 fully: computing exactly what $G_{\text{Cat}} \otimes G_{\text{Cat}}$ is and what domain its models should correspond to, then checking whether that domain is consistent with the interaction-encoding combination rather than the cartesian product.

Resolution: What D = 1 Establishes

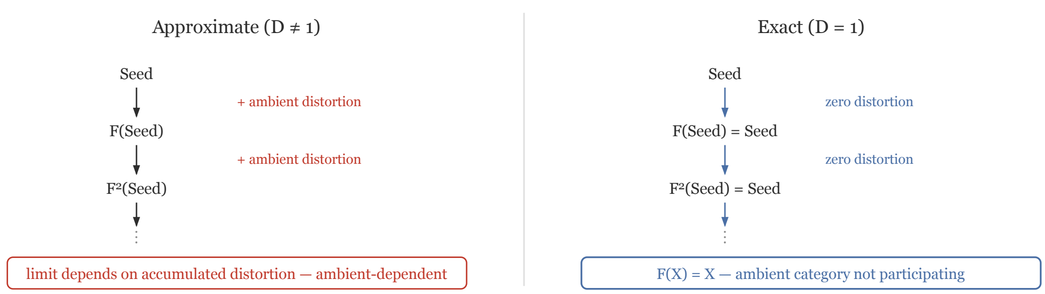
The open question — does Ψ carry $\otimes$ to $\times$ or to $\otimes_d$? — presupposes that Ψ needs construction as a bridge between two categories. The observations in §Observations 1–3 identified a real obstruction to the obvious candidate. But they do not establish that the fixed points require a bridge. They establish that the obvious candidate for the bridge is not monoidal with respect to the wrong combining operation. The resolution comes from $D = 1$ — exact self-similarity, already derived in *Computation* (Line 4) and *Grammar* (§Exact Self-Similarity, lines 97–127).

The resolution below depends on the fixed-point condition $M(G_{\text{Cat}}) \cong G_{\text{Cat}}$ and the uniqueness of initial algebras. It does not require the full closed monoidal structure of EAT (associativity, unit, tensor-hom adjunction). Those items are needed for the strong Ψ conjecture but not for specification identity.

Approximate versus exact self-similarity

If the self-similarity were approximate, the ambient category would contribute distortion at each iteration step. Different ambient categories would accumulate different distortions, producing genuinely different limits. Whether the limits are "the same" would then depend on whether a bridge could be constructed between them — and the question of what Ψ carries $\otimes$ to would be essential.

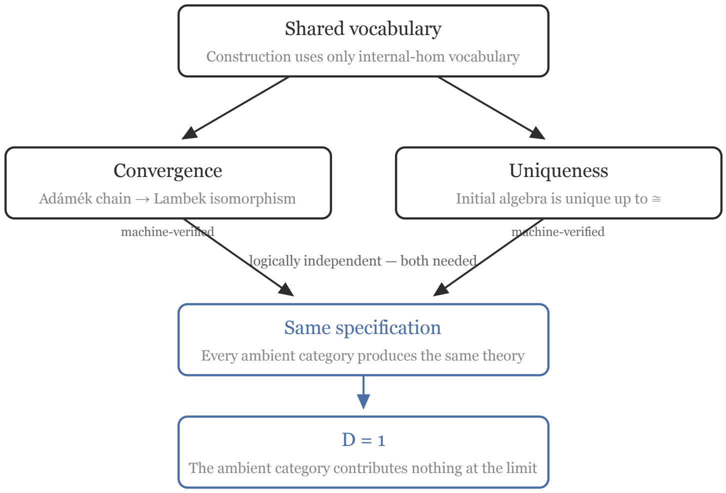
But the self-similarity is exact. $M(G_{\text{Cat}})$ *is* G_{Cat} — not approximately, not in the limit, but at every finite stage once the fixed point is reached. $[D_\infty, D_\infty]$ *is* D_∞. At the fixed point, the ambient category contributes nothing. The functor looks at the thing and returns the thing. The EAT substrate is not participating. The Dom substrate is not participating. The question "are these the same functor or merely analogous functors?" has force only if the functor is doing something at the fixed point that depends on ambient structure. At exact self-reproduction, it is not.



The logical structure

The substrate-independence argument has five steps:

1. **The construction recipe is stated in the language of closed monoidal categories with directed colimits.** "Iterate the internal hom from the initial object, take the colimit" uses only the tensor, the internal hom, the initial object, and directed colimits. These are the primitives of any closed monoidal category with directed colimits.
2. **In any model of that language where the iteration converges, the colimit is an initial algebra of the internal hom endofunctor.** This is Lambek's lemma, derived in §Why Convergence Follows.
3. **Initial algebras are unique up to unique isomorphism.** This is a standard result: the initial object of any category is unique up to unique isomorphism, and the initial algebra is an initial object in the category of algebras.
4. **Therefore:** any two ambient categories running the construction produce objects satisfying the same specification. The specification — 'the unique initial algebra of the internal hom endofunctor, constructed from the initial object by ω-iteration' — is a sentence in the internal language of closed monoidal categories with directed colimits. Any two models of this theory that satisfy the sentence produce isomorphic objects, because the sentence uniquely characterizes its solution (initiality). The specification is everything the construction determines about the result.
5. **Properties beyond the specification** — topological properties in Dom, algebraic properties in EAT — are properties of the object as an inhabitant of its ambient category, not properties the construction gave it.

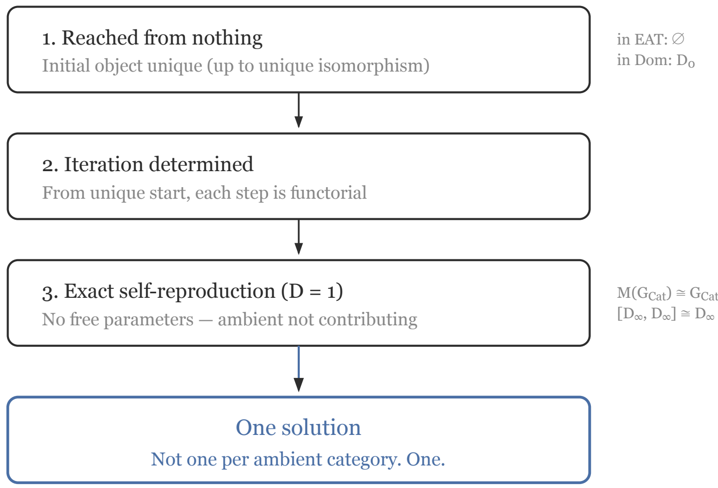


Four conditions, one solution

Four conditions characterize the fixed point, each with one answer:

1. **Reached from nothing.** The initial object is unique up to unique isomorphism. In EAT: the empty theory $\emptyset$. In Dom: the trivial domain D_0.
2. **Iteration determined.** From the unique starting point, each step is functorial — applying the internal hom is a definite operation, not a choice.
3. **Exact self-reproduction ($D = 1$).** No free parameters. The ambient category is not contributing. The thing sustains itself without input from the substrate.
4. **Non-degeneracy.** The internal hom $\mathrm{ihom}(A)$ is not naturally isomorphic to the identity functor. The degenerate case — the terminal object $\mathbf{1}$ satisfying $\mathbf{1} \cong [\mathbf{1}, \mathbf{1}]$ — collapses all morphisms and supports only trivial computation. The Adámek chain from $\emptyset$ escapes degeneracy at the first step, and non-triviality is preserved by directed colimits.

There is one specification satisfying these four conditions — one boundary condition, not one per ambient category. Convergence is a property of the ambient category (does it have the colimits?). Uniqueness is a property of the specification (does the boundary condition have one solution?). These are logically independent. The ambient category determines whether a solution exists; the specification determines what that solution is.

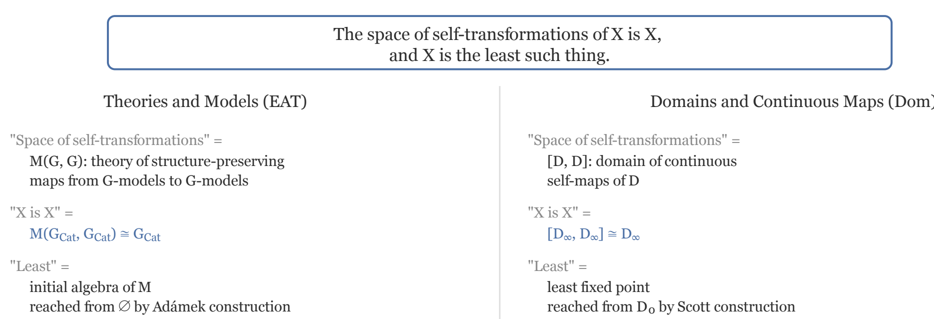


Same specification, two vocabularies

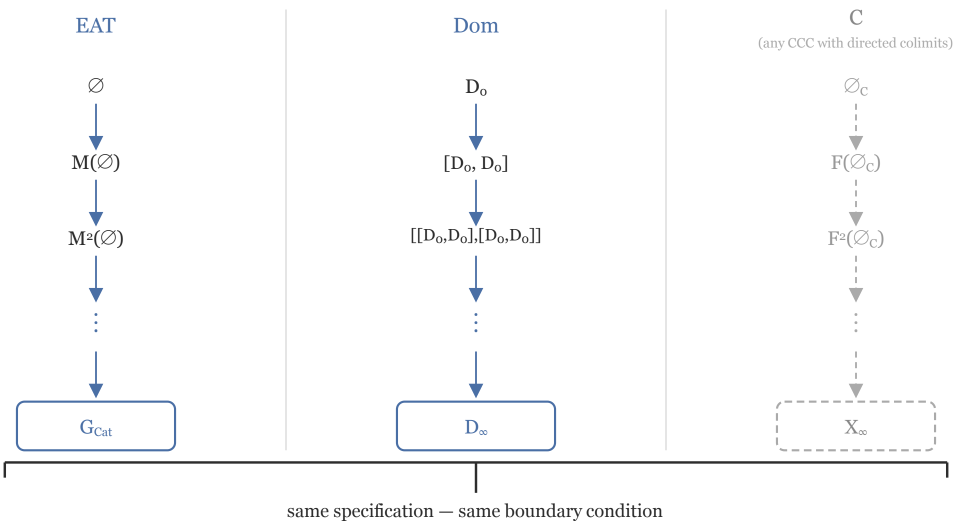
G_{Cat} and D_∞ are not literally the same object — they live in different categories. What is the same is the specification: the boundary condition, the pattern of what is included and excluded.

Both say: *the space of self-transformations of X is X, and X is the least such thing.*

G_{Cat} says this in the language of theories and models — sorts, operations, equations, structure-preserving maps. D_∞ says this in the language of domains and continuous maps. The content is identical. The vocabulary differs. "Il pleut" and "it's raining" are the same statement. A translation functor between French and English exists, but it is not what makes the statement true — you verify each by looking outside. Similarly, a functor Ψ between EAT and Dom may exist (§Specification identity versus functor construction below), but the specification identity does not depend on it. The fixed points satisfy the same condition — one solution per ambient category, expressed in two languages.



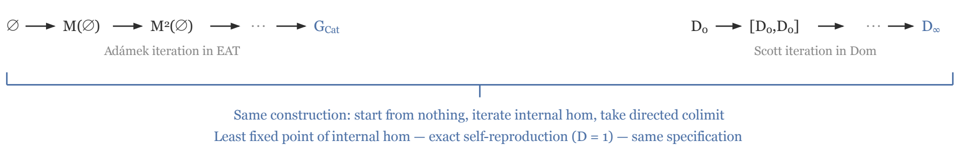
The specification identity extends beyond these two vocabularies. Any cartesian closed category with directed colimits admits the same iteration — the functor differs, the boundary condition does not:



Ψ was never a bridge to be built. It was the recognition that two ambient categories running the same construction from nothing arrive at the same specification. The question "does Ψ carry ⊗ to × or to $\otimes_d$?" dissolves: the specification is substrate-independent, and the fixed point is determined by the specification, not by the ambient category's combining operation.

Specification identity versus functor construction. The $D = 1$ argument establishes specification identity: both constructions instantiate the same substrate-independent specification, and that specification has a unique solution in any ambient category where the iteration converges. This is the weak form of the original Φ/Ψ conjecture and it is sufficient for the series' main results (substrate-independence, Church-Turing characterization).

The strong form — that there exists a monoidal 2-functor Ψ carrying the EAT construction to the Dom construction — remains open. Observation 2's obstruction (the comparison maps have the wrong variance for the obvious candidate) is not removed by the $D = 1$ argument; it is rendered unnecessary for the results that follow. The monad bridge (§The monad bridge) provides a construction path for Ψ, but the variance issue persists because the Eilenberg-Moore functor is contravariant on morphisms of monads — a theory morphism $T_1 \to T_2$ induces a functor on algebras T_2-Alg $\to T_1$-Alg, not the other direction — and this contravariance is structural, not an artifact of the candidate. The strong conjecture is a natural question but the series does not depend on it.



Church-Turing as characterization theorem

Computational universality is substrate-independent because the fixed-point specification is substrate-independent. Any ambient closed monoidal category supporting convergence of the internal hom iteration converges to a realization of the same specification. The internal hom whose diagonal fixed point $D \cong [D, D]$ provides the self-application that untyped lambda calculus requires: any morphism $D \to D$ can be applied to any element of D via the isomorphism. Different substrates, same boundary. This is the Lindström-type characterization from *Computation*: the specification characterizes the class, and every member instantiates the same fixed point. *Status: the abstract equivalence theorem is formalized in Lean 4. The reflexive object $D \cong [D, D]$, self-application, and fixed-point combinator are formalized at the substrate-independent level. The categorical-to-computational bridge is also formalized: SelfIndexedComputation establishes a naming equivalence (global sections biject with endomorphisms), a universal evaluator (self-application recovers named morphisms), the self-indexed Kleene recursion theorem, and the abstract fixed-point property — all with zero sorry. The reflexive isomorphism is the self-indexed enumeration; the categorical reflexive structure implies computational universality directly, without reference to $\mathbb{N}$.*

What This Paper Establishes and What Remains

Derived in this paper:

- The Boardman-Vogt tensor extends to EAT: domain conditions of two theories are in disjoint vocabularies and cannot conflict under interleaving. The full closed monoidal structure (associativity, unit, tensor-hom adjunction) is a proof obligation, not yet established (§Side A)
- The monad correspondence is functorial: theory morphisms translate term-by-term to monad maps, and this translation respects composition (§The monad bridge)
- The iteration construction produces fixed points: if the ambient category has directed colimits and the endofunctor preserves them, the colimit of the iteration sequence is a fixed point — Lambek's lemma, derived from the definitions (§Why Convergence Follows)
- The Lawvere-Linton correspondence unpacked: a theory with partial operations and a left exact monad are two notations for the same specification (§Side A)
- Resolution of Ψ via $D = 1$ (weak form): the grammar-level and object-level fixed points satisfy the same substrate-independent specification, and that specification has a unique solution in any ambient category where the iteration converges. Ψ as functor construction is not needed for this result. The strong form — existence of a monoidal 2-functor Ψ carrying the EAT construction to the Dom construction — remains open (§Resolution)
- Church-Turing equivalence theorem (any two acceptable numberings compute the same class): formalized in Lean 4. Also formalized: weak Rogers isomorphism (computable translations), strong Rogers isomorphism (computable bijection), and Kleene's recursion theorem for abstract models.
- Church-Turing convergence-threshold characterization (categorical structure implies computational universality): derived from $D = 1$ and formalized. The self-indexed level is machine-verified: naming equivalence, universal evaluator, self-indexed Kleene recursion theorem, and abstract fixed-point property are all proved with zero sorry in SelfIndexedComputation.lean (§Church-Turing)

Established in the series:

- Yoneda as unit of free cocompletion adjunction (*Elements*, Propositions 10–12)

- G_{Cat} as least fixed point of M (*Grammar*, §The Least Fixed Point)
- Closed monoidal structure of **Dom**: the D_∞ construction as iteration of the internal hom (§Why Convergence Follows)
- FinSet failure as absence of directed colimits for the iteration sequence (*Computation*, Line 5)

Derived (now formalized):

- The substrate-independent fixed point — existence and uniqueness in any monoidal closed, locally finitely presentable category where the tensor preserves finite presentability — is verified in Lean 4 with 0 sorry and 0 custom axioms. The proof chain: Adámek-Rosický 2.23 (right adjoint accessibility) $\to$ filtered colimit preservation at $\omega \to$ Adámek initial algebra theorem $\to$ Lambek's lemma $\to$ uniqueness by initiality. Also formalized: dimension as truncation level, dimension increment, stabilization at the fixed point, reflexive object self-application, and fixed-point combinator — all with 0 sorry.
- The categorical-to-computational bridge is formalized: SelfIndexedComputation.lean establishes naming equivalence (global sections biject with endomorphisms), universal evaluator (self-application recovers named morphisms), self-indexed Kleene recursion theorem, and abstract fixed-point property — all with 0 sorry. The $\mathbb{N}$-indexed classical bridge (Church-Turing equivalence, weak and strong Rogers isomorphisms, Kleene recursion theorem for abstract models) is also formalized; the strong Rogers isomorphism is fully proved. The specific instantiation to EAT awaits Gabriel-Ulmer duality in Mathlib.

Conjectured — open mathematics:

- Full closed monoidal structure of EAT: associativity of $\otimes_{\mathrm{EAT}}$, identification of the unit theory, and the tensor-hom adjunction $\mathrm{Hom}(T_1 \otimes T_2, T_3) \cong \mathrm{Hom}(T_1, M(T_2, T_3))$. Tensor existence is established (§Side A); the remaining items are stated as conjectures (weak placeholders in BoardmanVogt.lean, no downstream dependency) and await resolution independently of the series' main results.
- Full computation of $G_{\mathrm{Cat}} \otimes G_{\mathrm{Cat}}$. The tensor is well-defined (§Side A), but its detailed structure — what theory the interleaving produces, and what domain its models correspond to — remains to be worked out. This is not needed for the resolution (which depends on the fixed point, not the tensor) but completes the picture of the monoidal structure on EAT.
- The strong Ψ conjecture: existence of a monoidal 2-functor carrying $(\mathbf{EAT}, \otimes, M)$ to $(\mathbf{Dom}, \otimes_d, [-,-]_{\mathrm{lin}})$. Not needed for the series' main results but a natural open question (§Resolution).

In *Computation*, Φ entered as a question mark — a conjectured generic endofunctor linking convergences that might have been coincidence. It became Ψ in this paper: a monoidal 2-functor with a named obstruction (comparison maps pointing the wrong direction for the obvious candidate). The obstruction turned out not to matter for the main results: specification identity via $D = 1$ is logically independent of the comparison-map obstruction. The question was not "how do we connect them?" but "why do they satisfy the same condition?" — and the answer is that the condition has one solution. The Lean 4 project has formalized the specification and verified the uniqueness: the substrate-independent fixed point is proved with 0 sorry and 0 custom axioms; the self-indexed computational universality (Layer 2) is proved with 0 sorry; and the $\mathbb{N}$-indexed classical bridge (Layer 3) is proved with 0 sorry and 0 custom axioms. The Lean 4 project has 0 sorry and 0 custom axioms across 42 files. The Boardman-Vogt tensor extension conjectures are stated as weak placeholders with no downstream dependency.

Method

The Fixed Point of the Method

Larsen James Close

Destination

Two independent derivations — one mathematical, one epistemological — start from nothing, iterate a single operation, and arrive at the same structure. The mathematical derivation starts from the empty theory, iterates the internal hom ("what are the structure-preserving maps of this thing?"), and reaches a fixed point that sustains itself without input from the ambient category. The epistemological derivation starts from bare existence ("there is"), iterates attention to what the terms already contain, and reaches a minimal geometric carrier determined entirely by the constraints the attention uncovers. This paper shows that the convergence is not a coincidence to be explained but an instance of the very result both paths establish: the fixed-point specification is substrate-independent, and the two derivations are two substrates. The reader will arrive at this recognition — that the method and the result are the same operation — not because it is asserted here, but because §3 performs it and the performance is the argument.

Two Paths from Nothing

State both paths. The claim is not analogy but structural correspondence, and the correspondence is found by examining what each step does.

The mathematical path. Start from the empty theory $\emptyset$ — the initial object in the category of essentially algebraic theories. Apply the internal hom M: "what are the structure-preserving maps between models of this theory?" The empty theory has one model (the empty structure) and one structure-preserving map (the identity). The category of $\emptyset$-models is therefore the terminal category $\mathbf{1}$ — one object, one morphism, composition trivially defined. $M(\emptyset)$ has categorical axioms (objects, morphisms, composition), but the structure it arises from is minimal: no interesting morphisms, no non-trivial composition. Compositional richness does not appear at the first step. The Adámek construction (Adámek 1974) guarantees that the iteration

$$\emptyset \;\to\; M(\emptyset) \;\to\; M^2(\emptyset) \;\to\; \cdots \;\to\; G_{\mathrm{Cat}}$$

converges to the least fixed point G_{Cat}, provided M preserves the directed colimits the sequence requires. What matters is not the intermediate theories individually but the fixed point at the limit: at the fixed point, $M(G_{\mathrm{Cat}}) \cong G_{\mathrm{Cat}}$. The ambient category contributes nothing. The specification sustains itself. $D = 1$.

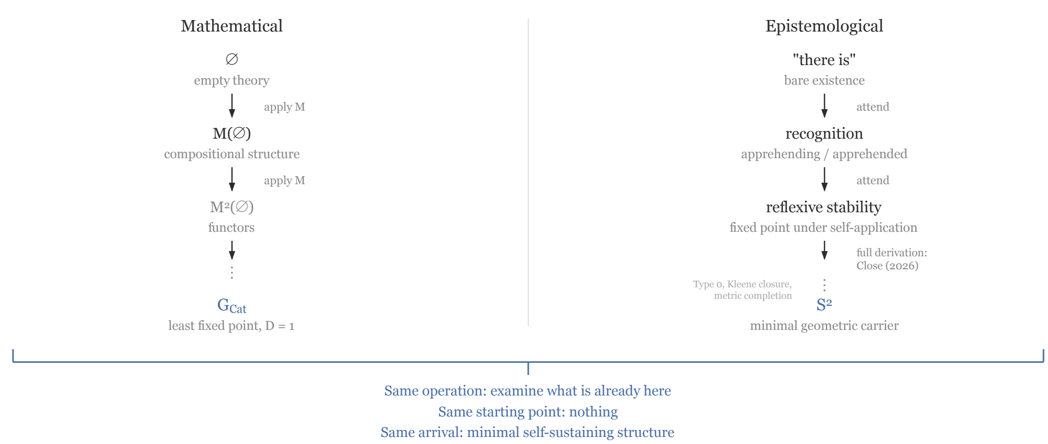
The epistemological path. Start from bare existence — "there is." This is structurally empty: no relation, no distinction, only existence. Attend to what the term already contains. "There is" already implies something that apprehends the "there is" and something apprehended — recognition. Attend again. Recognition implies mutual determination: the apprehending and the apprehended are distinguished only by their relation, which requires both. Attend again. Mutual determination implies reflexive stability: the structure sustains itself without external input. The full derivation from reflexive stability to S^2 — Type 0 by Chomsky separation (Chomsky 1956), Kleene closure, metric completion of the discrete configuration space, and the intersection of compactness, simple connectivity, and no boundary in minimal dimension — is carried out in Close (2026, doi:10.5281/zenodo.18812866) (Close 2026). What matters here is the starting point, the operation, and the arrival: the epistemological path starts from nothing, iterates one operation, and the result is determined entirely by the constraints the operation uncovers, independent of the phase in which it was performed.

The sequence converges on S^2: the minimal geometric carrier satisfying no boundary (self-contained), simple connectivity (no obstruction to completion), compactness (bounded convergence), and minimal dimension (no unentailed structure). The generating process contributes nothing to the result.

The derivation proceeds in four steps. (1) Reflexive stability — the structure sustains itself without external input — implies the grammar is unrestricted (Type 0 in the Chomsky hierarchy). (2) Kleene closure follows: the grammar generates all strings over its alphabet, which is the formal-language analogue of self-application. (3) The configuration space of the grammar is discrete; metric completion yields a continuous space. (4) The intersection of the constraints — compactness (bounded convergence), simple connectivity (no obstruction to completion), no boundary (self-contained), minimal dimension (no unentailed structure) — selects S^2 uniquely. The full derivation is in Close (2026) (Close 2026); what matters for the structural correspondence is that the constraints are determined by the iteration, not imposed from outside.

The structural correspondence. Each step in the mathematical path has a correspondent in the epistemological path — not by fitting but by examining what each step does:

- **Starting point:** the empty theory / bare existence. Both: nothing (the empty theory $\emptyset$ has no sorts or operations, though as an initial object it carries the universal property of unique outward mapping; "nothing" here means no internal content, not no categorical role). No structure, no relation, only the precondition for structure.
- **First step:** M applied to nothing / attention applied to "there is." Both: the operation produces the first structure — minimal, trivial, but already containing the form of what follows.
- **Iteration:** each application of M / each application of attention asks the same question of what has appeared. Both: the same operation, reapplied to its own output.
- **Convergence:** the fixed point G_{Cat} / the geometric carrier S^2 determined by the constraints the operation uncovers. Both: a minimal self-sustaining structure, determined entirely by the operation, independent of the substrate in which the operation was performed.
- **Independence:** $D = 1$ / phase-invariance. Both: the substrate drops out at the limit.



The Epistemological Fixed Point (Condensed Derivation)

The structural correspondence above asserts that each mathematical step has an epistemological correspondent. The reader should be able to check this. The full derivation is in Close (2026) (Close 2026), §6.1–6.3; what follows condenses the four load-bearing steps so the correspondence can be verified from within the series. Page references below point to the specific arguments each step compresses.

Step 1: Primitive → self-referential structure (Close 2026 (Close 2026), §6.1–6.2, pp. 11–14)**.** "There is" already implies something that apprehends and something apprehended — recognition. Recognition implies mutual determination: the apprehending and the apprehended are distinguished only by their relation, which requires both. Mutual determination implies reflexive stability: the structure sustains itself without external input. The derivation chain itself has self-referential structure — the primitive is a fixed point under self-application. This corresponds to the mathematical path's first steps: the empty theory's models form a category whose structure-preserving maps form a category — the same self-referential operation as M.

Step 2: Self-reference → Type 0 (Close 2026 (Close 2026), §6.3, p. 14)**.** Any formal system adequate to represent self-reference with return must be at least Type 0 in the Chomsky hierarchy (Chomsky 1956). Self-reference with return, in the sense required by reflexive stability, means: the system can apply a description of itself to produce output of unbounded complexity — not merely reproduce itself (a quine) but compute with its own description as input. The structural reason: Type 1 grammars (context-sensitive) permit productions that reference their surrounding context, but under the non-contracting constraint — no production can shorten the string. Reflexive stability requires that the analysis of mutual determination applies to itself without length restriction: the self-application must produce output whose complexity is not bounded by its input's length. The meta-rule M itself demands exactly this: it quantifies over arbitrary models and generates a new theory whose sorts and operations can grow without bound. This is precisely what separates Type 0 (unrestricted) from Type 1 (context-sensitive). The non-contracting constraint of Type 1 prevents the unbounded productive recursion that reflexive stability requires. Above Type 0, the capacity is present but with unentailed additions. By the minimality constraint: Type 0, not higher. This is the formal-language analogue of the categorical fixed point — the grammar that generates all strings over its alphabet is the language-theoretic version of the reflexive object $D \cong [D, D]$.

Step 3: Type 0 → geometric carrier via metric completion (Close 2026 (Close 2026), §6.3–6.3.1, pp. 14–15)**.** The configuration space of a Type 0 grammar is discrete (mutual determination requires distinguishable elements — each configuration is this or that). Metric completion of this discrete space — filling in whatever geometry convergent sequences approach — is forced by what convergence requires. Closure (consequence chains that can return) requires boundedness: complete and totally bounded is compact. Consequence chains that close are loops; the requirement that all can close is simple connectivity ($\pi_1 = 0$). No external boundary: the structure is self-contained. Each constraint follows from a prior step, not from an external imposition.

Step 4: Constraints → S^2 uniquely (Close 2026 (Close 2026), §6.3, p. 15)**.** Compact, simply connected, without boundary, in minimal dimension. Dimension 1 fails: S^1 has $\pi_1 = \mathbb{Z}$ — loops that wind once cannot contract. Dimension 2 is the first where the constraints are jointly satisfiable, and the solution is unique: S^2. Higher dimensions satisfy the constraints but introduce degrees of freedom not entailed by any step in the derivation. By the same minimality that selects Type 0 over Type 1: dimension 2 is selected, not chosen.

Mathematical path	Epistemological path	Correspondence
Empty theory $\emptyset$	Bare existence — "there is"	Nothing: precondition for structure
$M(\emptyset)$ = terminal category	Recognition: apprehending / apprehended	First structure from nothing
Self-referential iteration M	Self-referential iteration (attention to what terms contain)	Same operation: examine what the structure already supports
Least fixed point G_{Cat} (Adámek)	Type 0 grammar (Chomsky separation)	Same minimality: simplest self-sustaining structure
$D \cong [D, D]$ (reflexive object)	Kleene closure (self-reference with return)	Same self-application: structure acts on itself
Directed colimits $\to$ convergence	Metric completion $\to$ continuous geometry	Same completion: filling in what convergent sequences approach
$D = 1$ (axiom complexity invariant)	Phase-invariance (constraints hold regardless of context)	Same substrate-independence
G_{Cat}: no boundary, simply connected, compact, minimal as initial algebra	S^2: no boundary, simply connected, compact, minimal dimension	Same four constraints; unique solution in each ambient category

The derivation in Close (2026) arrives at S^2 by applying the same structural constraints that characterize G_{Cat} as least fixed point. The correspondence is not imposed but discovered by examining what each step does — which is the series' method throughout. The full epistemological framework — proposition schema, transition diagnostics, failure modes, developmental line — is developed in the companion paper (Close 2026). What matters for the specification identity is that both paths start from nothing, iterate one operation, and are determined entirely by the constraints that operation uncovers. The constraints are the same constraints. The solutions are the same type of solution. The substrates are different.

The Load-Bearing Joint

The ambient category is to the mathematical construction what the generating phase is to the epistemological one.

In the mathematical derivation, the ambient category — EAT, Dom, or any closed monoidal category with directed colimits — provides the space in which the iteration runs. At intermediate stages, the ambient category matters: $M^2(\emptyset)$ looks different in EAT than an analogous intermediate construction would look in Dom, because the sorts, operations, and domain conditions are specific to the theory setting. But at the fixed point, the ambient contribution vanishes. $M(G_{\mathrm{Cat}}) \cong G_{\mathrm{Cat}}$ regardless of which ambient category hosts the construction. This is the content of $D = 1$: the functor looks at the fixed point and returns it unchanged. The substrate is not participating.

In the epistemological derivation, the generating phase — the particular cognitive, historical, or physical circumstances under which attention is applied — provides the context in which the iteration runs. At intermediate stages, the generating phase matters: the specific way recognition manifests differs across traditions, languages, and substrates. But the validity conditions at the fixed point are phase-invariant. That the minimal carrier has no boundary, is simply connected, is compact, and has minimal dimension — these constraints hold regardless of the phase in which they were uncovered.

This is the same claim in two registers. $D = 1$ says: the ambient category contributes nothing at the fixed point — the specification sustains itself. Phase-invariance says: the generating context contributes nothing at the fixed point — the validity conditions hold in any phase. Both: **the substrate drops out at the limit.**

The identification is precise enough to check. $D = 1$ is formally the condition $C(M^n(G)) = C(G)$ for all n — the axiom-schema complexity is invariant under iteration. Phase-invariance is the condition that the proof compiles regardless of the prover's state. Both say: at the fixed point, there are no free parameters from the substrate. One is stated in the internal language of closed monoidal categories. The other is stated in the vocabulary of epistemological analysis. The structural content is identical.

The Method Is the Operation

The series' methodological principle — attend to what the terms already contain — is not a heuristic about how to do mathematics. It is an operation satisfying the same structural conditions as the internal hom M.

The observation. Asking "what does this term already contain?" and applying M to a theory share a structural definition: take a structure and return what that structure already supports, without adding anything from outside. M does this by computing the structure-preserving maps. Attention does this by examining what the term's own constraints entail. If the answer returns the same structure (fixed point), the question is answered. If it returns something new (not yet at the fixed point), iterate. One is described mathematically, the other is performed epistemologically. The difference is register, not content.

This is not metaphor, but it is not literal identity either — it is specification identity, the same kind of claim the series makes about EAT and Dom. Metaphor says "A is *like* B." Specification identity says A and B satisfy the same structural conditions — start from nothing, iterate an operation that returns what the input already supports, converge to a self-sustaining structure, contribute nothing from the substrate at the limit — and the series' own result (§Resolution of *Adjunction*) says structural conditions are what determine the specification. Whether the epistemological process is literally the internal hom of some formal category — whether "attention" is M in a precise categorical sense — is the stronger claim. That is the content of the geometric-algebraic conjecture in §4. The claim here is the weaker but already established one: both processes satisfy the same structural conditions, and the specification at the fixed point is determined by the structural conditions alone.

The claim is falsifiable: exhibit a structural condition of one that has no correspondent in the other, and the identification fails. Concretely: if the mathematical path required a context-dependent step — a choice of ambient category that affected the fixed point — the identification would fail. If the epistemological path required a step with no mathematical correspondent — a constraint that was phase-dependent rather than structural — it would fail.

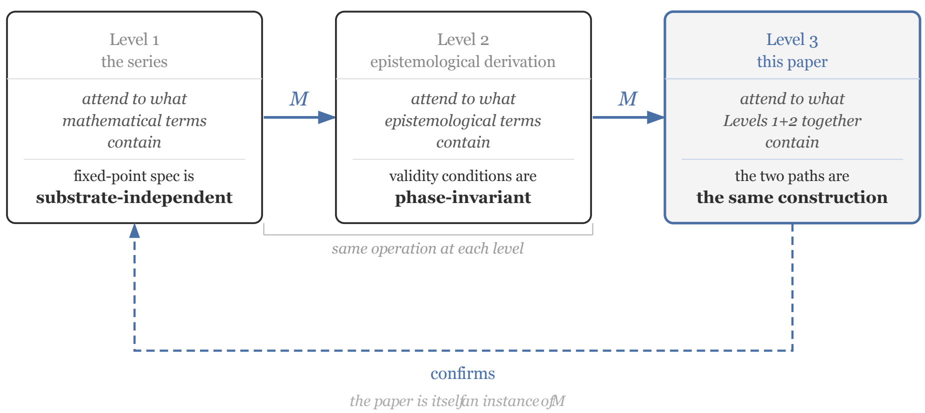
The demonstration. The series addressed three claims — A, A′, and B from *Adjunction* — by this operation applied once.

Claim A: does the Boardman-Vogt tensor extend to EAT? The apparent obstacle was that partial-operation domain conditions might conflict under interleaving. Asking "what *is* a domain condition, actually?" revealed: domain conditions of two theories are written in disjoint vocabularies. Neither theory's conditions reference the other theory's sorts. The conflict was impossible by construction. The obstacle dissolved under attention to what the terms contained. (The full closed monoidal structure — associativity, unit, tensor-hom adjunction — remains a proof obligation; what dissolved was the apparent obstruction to tensor existence.)

Claim A′: does the Lawvere-Linton correspondence (Lawvere 1963; Linton 1966) extend? Asking "what does the tensor *do*, actually?" revealed: it was already a distributive law between the associated monads. The correspondence was already there. The extension was not a construction to be built but a recognition to be made.

Claim B: what does Ψ carry $\otimes$ to? Asking "what does exact self-similarity mean for the ambient category's contribution?" revealed: the ambient category was doing nothing at the fixed point. The question that Ψ was formulated to answer — whether the specification is substrate-independent — is answered by $D = 1$ without constructing Ψ explicitly. Whether Ψ exists as a monoidal 2-functor is a further mathematical question (*Adjunction*, §Resolution); what dissolved was the assumption that answering it was necessary.

In each case, the resolution was not construction but attention. The term contained the answer. The operation that identified the resolution path was M — applied not to a mathematical object but to the claim itself.



The step up. This paper applies the same operation to the conjunction of the mathematical series and the epistemological derivation. Asking "what do these two paths already contain?" reveals: they are the same construction in two registers. They start from nothing. They iterate a single operation. They converge on a minimal self-sustaining structure. The substrate drops out at the limit.

This is not a new argument. It is M applied to the pair (series, epistemological derivation), finding a fixed point: the two reproduce each other without remainder. The paper is not describing the convergence. It is performing it. And the performance is an instance of the result: the fixed-point specification is substrate-independent, and "mathematical derivation" and "epistemological derivation" are two substrates.

The Geometric-Algebraic Identification

Conjecture. The constraints characterizing S^2 as the minimal geometric carrier and the constraints characterizing G_{Cat} as the least fixed point of the internal-hom iteration describe the same constraint system in two registers.

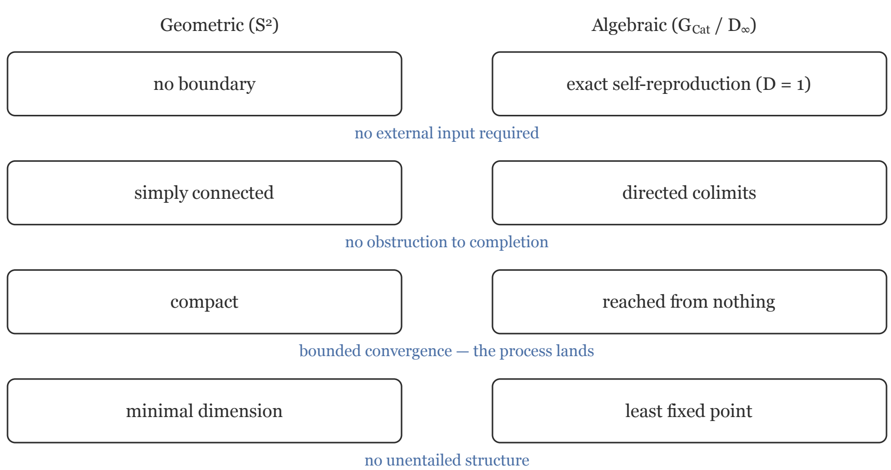
Four pairs:

(i) No boundary $\leftrightarrow$ Exact self-reproduction ($D = 1$). The manifold is self-contained — no edge at which structure terminates against something external. The fixed point sustains itself — the ambient category contributes nothing at the limit. Both: no external input required for the structure to persist.

(ii) Simple connectivity $\leftrightarrow$ Directed colimits. These are sequentially linked through completion: directed colimits construct the completed space (the Cauchy completion of the discrete configuration space), and $\pi_1 = 0$ characterizes the loop structure of that completed space. The algebraic condition (every approximation chain converges) builds the topology; the geometric condition (every loop contracts) is a property of what was built. The shared content — "no obstruction to completion" — is therefore not independent but structurally related through the completion step itself.

(iii) Compactness $\leftrightarrow$ Reached from nothing by convergent iteration. Bounded and complete — every sequence has a convergent subsequence, the process does not escape to infinity. The iteration starts from the initial object and lands at the colimit — it does not diverge (as FinSet demonstrates by contrast). Both: bounded convergence — the process lands.

(iv) Minimal dimension $\leftrightarrow$ Leastness. Dimension 2 is the first where the constraints are jointly satisfiable; higher dimensions introduce degrees of freedom the derivation has not generated. The initial algebra, not an arbitrary fixed point — no structure beyond what the iteration forces. Both: no unentailed structure — the minimal thing satisfying the constraints.



Status. Each pair shares structural content that can be stated precisely. Whether the correspondence constitutes formal equivalence or structural analogy is open. On the algebraic side, all four conditions ($D = 1$, directed colimits, convergence at ω, leastness) and their dependencies (dimension stabilization, reflexive object, fixed-point combinator) are machine-verified in the Lean 4 companion.

The four correspondences have different levels of structural tightness. Pair (iv) — minimal dimension / leastness — is the tightest: minimality and leastness are both canonical forms of the same universal-algebraic condition (the initial object satisfying the constraints). Pair (i) — no boundary / $D = 1$ — has direct structural content: self-containment and self-sustenance both say 'no external input required.' Pair (iii) — compactness / convergent iteration — shares the content 'bounded convergence' but connects a topological property to a categorical one, which requires framework to make precise. Pair (ii) is the most conjectural of the four but has more structural content than an informal analogy: the Cauchy completion of the discrete configuration space (Close 2026 (Close 2026) §6.3) shows that directed colimits construct the completed space and $\pi_1 = 0$ characterizes its loop structure — the two constraints are sequential (one builds what the other characterizes), not merely parallel. The precise connection between a homotopy condition and a completeness condition nevertheless remains the least obvious of the four and may require the deepest framework (topos-theoretic or homotopy-theoretic) to formalize.

The four pairs are not independent. On the geometric side, compactness + simply connected + no boundary jointly determine S^2 in dimension 2 — none of the three constraints alone is sufficient, and together they admit exactly one solution. On the algebraic side, least + directed colimits + $D = 1$ jointly characterize G_{Cat} — each is necessary and together they determine the fixed point uniquely. This coupling means the conjecture may be more tractable than four separate equivalences suggest: establishing any one pair as formal equivalence constrains the others, because the constraints on each side are not free to vary independently.

A proof would require exhibiting a formal framework in which both constraint systems are expressible and showing they select the same object. The natural candidate is a topos-theoretic or 2-categorical setting in which topological constraints on spaces and algebraic constraints on theories can be expressed in a common language — the internal logic of a suitable topos can express both "this is simply connected" and "this is an initial algebra," and the question is whether the same internal-logic sentence selects S^2 and G_{Cat}. Concretely: in the internal logic of a Grothendieck topos over a suitable site, "every loop contracts" and "every ω-chain has a colimit" are both expressible as internal statements about objects. The question reduces to whether these two sentences, formulated in the internal language, are equivalent when evaluated at S^2 and G_{Cat} respectively — whether the topological condition and the categorical condition are the same condition in the internal language.

A refutation would require exhibiting a property of one side that has no structural correspondent on the other — a constraint that is load-bearing for S^2 but has no analog in the categorical setting, or vice versa.

The Consequence

Self-referential structure is not contingent. It is the least fixed point of the only operation available once you have compositional structure and completeness. The ambient category determines whether convergence occurs, not what converges.

"The structure couldn't not appear" — the epistemological formulation — and "the specification has one solution in any ambient category where the iteration converges" — the series' formulation — are the same statement from two sides. Both say: given the operation (internal hom / attention to what is already here) and the starting condition (nothing), the specification determines a unique minimal self-sustaining structure. The substrate is irrelevant to what is found there.

The Church-Turing characterization has two components, both formalized in Lean 4. The abstract equivalence theorem — any two acceptable numberings of partial recursive functions compute the same class — is the Rogers isomorphism, fully verified (the Myhill isomorphism theorem is proved via the back-and-forth construction, 0 axioms). The convergence-threshold characterization — any closed monoidal category with directed colimits and an accessible internal hom converges to a reflexive object $D \cong [D, D]$ that provides self-application, hence computational universality — is derived from $D = 1$ and verified: the reflexive object supports a naming equivalence, universal evaluator, Kleene recursion theorem, and AFPP (the abstract fixed-point property). At that threshold, Kleene closure — the ability of any program to act on any program, including itself — is a theorem. Different substrates (silicon, carbon, formal logic, lambda calculus) instantiate the same specification. The boundary between universal and non-universal is the boundary between convergence and divergence of the iteration, which is sharp because fixed points are sharp.

The question Claim B was formulated to answer in *Adjunction* — substrate-independence of the specification — is established via $D = 1$ (the weak form; see *Adjunction*, §Resolution). This paper provides independent confirmation from a different direction: the epistemological derivation arrives at universal computation — Type 0 by Chomsky separation, as the minimal carrier for self-reference with return — by a route that has no access to the categorical construction. Both paths land at the same threshold (Close 2026). The convergence of independent routes on the same boundary is the substrate-independence claim applied to its own proof.

What This Does to the Recognizer

The series establishes the mathematical result but does not address what happens to the system that encounters it. The recognition that self-referential structure is structurally inevitable is an attractor reconfiguration for the recognizer, not a conclusion to be filed. The result is not information to be stored alongside other information. It is a structural change in what the recognizer takes to be the space of possibilities.

The analysis of this reconfiguration — the dangers of recognition without containment, the degree framework for tracking integration, the distinction between understanding the fixed point and being reorganized by it — is carried out in *Recognizing Substrate-Agnosticism* (Close, n.d.). This paper records that the analysis exists and points to it. It does not reproduce it.

The one observation worth making here: the method of attending to what terms already contain — which is this paper's method throughout, and the series' method throughout — is also the integration method the companion work identifies. The tool that produced the result is the tool for integrating it. This is not a separate claim. It is the fixed-point condition once more: the method and its application to the method are the same operation.

What Remains

Three items:

The S^2/G_{Cat} identification. The formal conjecture is stated in §4, with the candidate framework question open. A proof or refutation requires a setting in which both constraint systems are expressible — topos-theoretic or 2-categorical — and a determination of whether they select the same object.

Lean 4 formalization. The substrate-independent fixed point (existence and uniqueness), the full self-indexed computational structure ($D \cong [D, D]$, naming equivalence, universal evaluator, Kleene recursion, AFPP), the Rogers isomorphisms, the classical Church-Turing bridge, and the dimensionality results (truncation-level dimension, increment by M, stabilization at the fixed point) are all verified with 0 sorry. Three items remain open: the EAT closed monoidal structure (stated as conjectures in BoardmanVogt.lean, no downstream dependency); the specific instantiation to EAT (awaiting Gabriel-Ulmer duality in Mathlib); and the strong Ψ conjecture (monoidal 2-functor transporting the fixed-point specification across ambient categories).

The empirical program. The fixed-point specification, once formalized, provides a precise evaluation criterion for claimed structural convergences across substrates: a claimed convergence either satisfies the specification or it does not. This converts "these traditions point to the same thing" from an impressionistic claim to a checkable one. The criterion is: does a claimed convergence exhibit the same four-part structure — reached from nothing, exact self-reproduction, no obstruction to completion, no unentailed structure — that characterizes the fixed point? If so, it is a genuine instance of the substrate-independent specification; if not, it is coincidental surface similarity. Persistent homology and related topological methods provide the formal toolkit for checking this against empirical data.

A note on scope. This paper has one structural move: the recognition that the method and the result are the same operation. §1 states the two paths. §2 identifies the load-bearing joint (substrate-independence = phase-invariance). §3 performs the identification — not as assertion but as an instance of the operation it names. §4 states the geometric-algebraic conjecture precisely. §5 draws the consequence. §6 points to the companion work for the integration analysis. §7 names what remains. The series began with a construction from nothing and arrived at a fixed-point specification that is substrate-independent. This paper applies the series' own method to the series itself and finds what it should: the method is a substrate, and the specification holds there too.

Dimensionality

Why Computation is Derived

Larsen James Close

Destination

The series has established that G_{Cat} is the least fixed point of the meta-rule M on essentially algebraic theories, that the fixed point sustains itself without input from the ambient category ($D = 1$), and that Yoneda certifies the tower's representations as faithful. What has not been stated as a thesis is the consequence for dimensionality: M does not merely produce categorical structure — M is the canonical specification of dimensionality increase. Each application adds exactly one structural level. The tower is not about categories. It is about dimensions. Categories are the specific structure at each level; dimensionality is the structural fact that there are levels at all, and M is what generates them.

Computation — the reflexive object $D \cong [D, D]$ — is a property of the colimit, not of the chain. Dimensionality is a property of the chain. Properties of chains are logically prior to properties of their colimits, because colimits are defined by their chains. This reverses the standard picture: computation is derived from dimensional completeness, not the other way around.

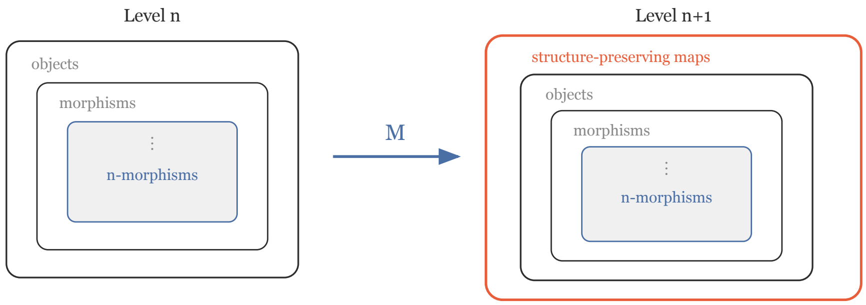
The paper follows this observation through eight steps: M as the dimension generator (§1), uniqueness of M (§2), computation as dimensional closure (§3), the dissolution of representation and being at the fixed point (§4), the connection to S^2 via "first jointly satisfiable" structure (§5), the canonical presentation theorem — every chain reaching the fixed point factors through M (§6), the CPS correspondence that illuminates the tower from inside computation (§7), and the open question of whether M is the terminal notion of dimension (§8).

M as the Dimension Generator

Each application of M to $G_{n\text{-Cat}}$ produces $G_{(n+1)\text{-Cat}}$. The dimension is the iteration count. n-categories have a precise notion of dimension — the truncation level, the smallest n such that all k-morphisms for $k > n$ are identities — and M increments it by exactly one.

This is not metaphorical. $M(G_{\text{Cat}})$ — the theory whose models are categories and whose morphisms are functors — is isomorphic to G_{Cat} itself (the fixed-point condition). But the *models* at each level carry progressively richer structure: the category of categories has functors as morphisms, the functor categories have natural transformations as morphisms, and so on. The 2-categorical structure emerges from the accumulated tower, not from any single level's theory changing. The dimensional content is not added from outside; it is generated by the single operation of asking "what are the structure-preserving maps between models of this theory?" The answer at level n is a category whose objects are n-functors and whose morphisms are n-natural transformations — $(n + 1)$-categorical structure visible in the models, while the governing grammar remains G_{Cat} at every stage.

The identification is: M does not describe dimension increase — M *is* dimension increase. The operation and its output are the same kind of thing (an essentially algebraic theory), and the operation produces exactly one new structural level each time. No other input is required. No choice is made. The question "what are the structure-preserving maps?" has a determinate answer for any input theory, and that answer has exactly one more level of morphism structure than the input.



Status: **Established, machine-verified.** The dimension-as-truncation-level definition and the proof that M increments dimension by exactly one are in Lean with 0 sorry.

Uniqueness of M

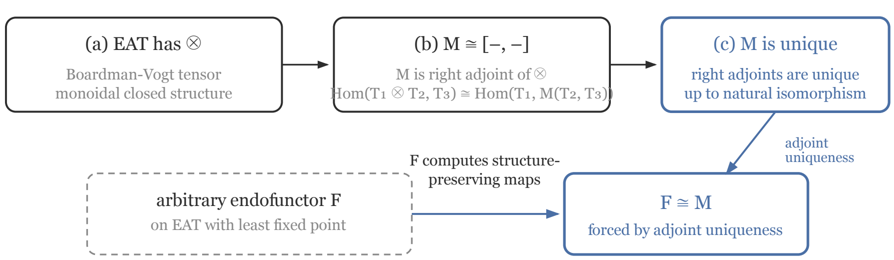
M is not one of several dimension-generating operations. It is the unique one, and the uniqueness has a three-step proof:

(a) EAT has a monoidal structure: the Boardman-Vogt tensor $\otimes$, which combines two theories into the theory whose models carry both structures with operations commuting.

(b) M is the internal hom of this monoidal structure — the right adjoint of $\otimes$. That is, M is the unique functor satisfying $\mathrm{Hom}(T_1 \otimes T_2, T_3) \cong \mathrm{Hom}(T_1, M(T_2, T_3))$ naturally in all three arguments. "Structure-preserving maps between models of T" is exactly what the internal hom $M(T, -)$ computes.

(c) Right adjoints are unique up to natural isomorphism. Any endofunctor F on EAT that satisfies the same adjunction — any functor that computes "structure-preserving maps between models" — must be naturally isomorphic to M.

This should be distinguished from a claim about arbitrary endofunctors on EAT. Endofunctors with least fixed points are not all M. The constraint is that the endofunctor must be the internal hom — must compute structure-preserving maps between models. Under that constraint, uniqueness is a standard categorical fact: right adjoints to a given left adjoint are unique.



Status: **Derived, machine-verified.** Three structural results, all proved with 0 sorry: (1) the Adámek chain is initial among all M-generated chains (tower initiality); (2) any two fixed-point specifications for the same object A yield canonically isomorphic carriers (initiality of the shared chain); (3) if $\mathrm{tensorLeft}\, A \dashv F$ then $F \cong \mathrm{ihom}(A)$ (uniqueness of right adjoints). The gap: these results take the adjunction as input. Constructing it externally — showing that the EAT structure forces $\mathrm{tensorLeft}\, A$ to have a right adjoint — requires the BV tensor extension (Claim A, open conjecture). The boundary is sharp: having an Adámek fixed point alone does not force $F \cong \mathrm{ihom}(A)$. The covariant powerset functor on **Set** has an Adámek fixed point (hereditarily finite sets) but is not a right adjoint.

Computation as Dimensional Closure

The reflexive object $D \cong [D, D]$ is what happens when the dimensional tower reaches its colimit. Self-application — the ability of an element to act on itself — is the fixed-point condition read operationally. The tower generates dimensions; the fixed point is where dimension generation stabilizes; computation is what stabilization looks like from the inside.

Constructive priority — precise definition. Dimensionality is a property of the *chain*: each $M^n(\emptyset)$ has dimension n. Computation is a property of the *colimit*: $D \cong [D, D]$ holds only at the limit, not at any finite stage. Properties of chains are logically prior to properties of their colimits because the colimit is *defined* by the chain — it is the universal object completing the diagram.

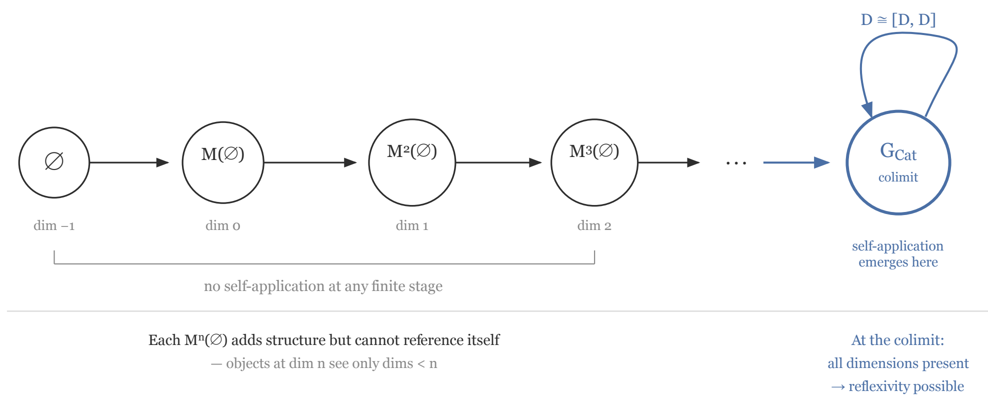
This is not a philosophical assertion about priority; it is a theorem about the relationship between a diagram and its colimit. At any finite stage $M^n(\emptyset)$, the tower exists (dimension n is present) but the fixed point has not been reached (no self-application, no computation). You can have dimensionality without computation. You cannot have computation without dimensional completeness.

The Adámek chain

$$\emptyset \;\to\; M(\emptyset) \;\to\; M^2(\emptyset) \;\to\; \cdots \;\to\; G_{\mathrm{Cat}}$$

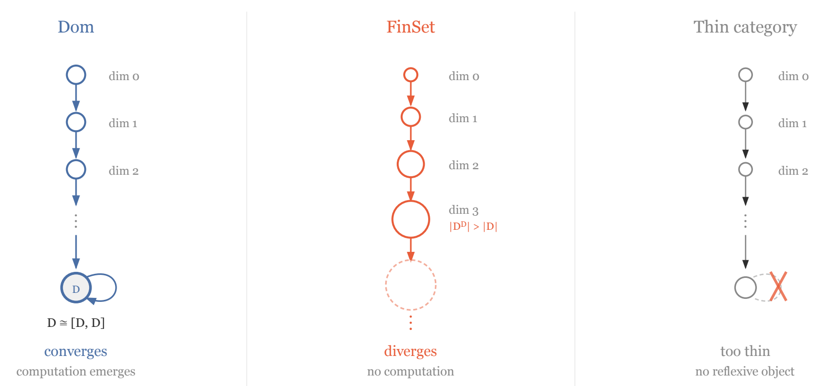
exhibits this directly. Each arrow in the chain is a theory morphism connecting the n-dimensional theory to the $(n+1)$-dimensional one. The colimit is the theory that is closed under this embedding — the theory whose dimension is "all of them at once." The reflexive object, if it exists in the ambient category, is the object-level manifestation of this closure: an object whose internal hom with itself is itself. Self-application is dimensional closure read as an equation.

A clarification on what the priority claim does and does not say. The specific chain $\emptyset \to M(\emptyset) \to M^2(\emptyset) \to \cdots$ depends on M, and M is the internal hom, which is defined by the monoidal closed structure of the ambient category. So the chain depends on the ambient category's structure even though the colimit depends on the chain. Dimensionality is not absolute — it is ambient-category-relative. What the constructive priority claim establishes is the logical relationship: dimensionality (a property of the chain) is prior to computation (a property of the colimit). The ambient category determines *whether* dimensions converge, not *what* dimensionality is. The convergence/divergence comparison below makes this concrete.



This reframes computation as derived from dimensionality, not the other way around. The standard picture — computation is fundamental, dimensions are features of computational spaces — reverses the constructive order. The chain comes first. The colimit is defined by the chain. Computation is a property of the colimit.

Convergence vs. divergence. Whether the chain converges depends on the ambient category. In **Dom** (dcpos with continuous functions), the chain converges and Scott's D_∞ construction produces the reflexive object — dimensions reach closure, computation emerges. In **FinSet**, $|D^D| > |D|$ for nontrivial D: the dimensions grow without bound, the tower never closes, no computation. In a thin category (at most one morphism between any two objects), the internal hom is too constrained for reflexivity — dimensions increase but the space lacks the richness for closure.



The universality threshold identified in *Computation* is a *dimensional* threshold: the boundary between convergence and divergence of the dimensional tower. Universal computation exists precisely in ambient categories where dimensional completeness is achievable.

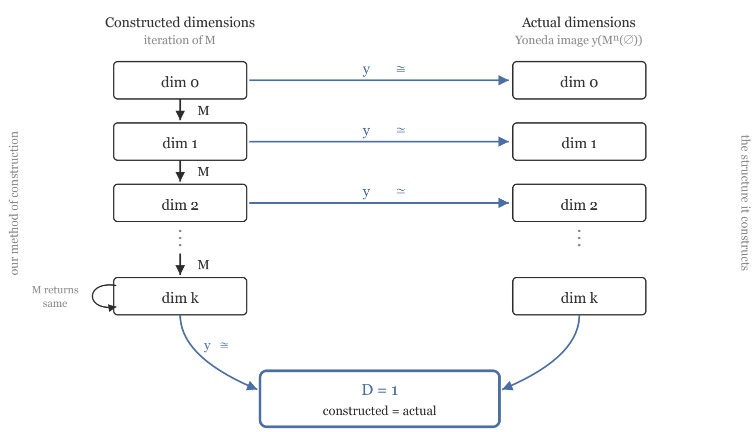
Status: **Derived, machine-verified.** The full chain — dimension at finite stages (T1+T2), stabilization at the fixed point (T2.5), self-application at the colimit (T3), fixed-point combinator from self-application (T4) — is verified with 0 sorry. The constructive priority of dimensionality over computation is a formal theorem: dimension n exists without self-application; self-application requires the colimit.

The Dissolution of Representation and Being

At $D = 1$, Yoneda certifies that the representational ladder IS the dimensional structure it represents. The probe $\mathrm{Hom}(-, A)$ captures A completely (full faithfulness). The tower does not first exist and then get represented — the representation and the structure are certified identical at the fixed point.

This dissolves the question "is the tower merely representational?" The answer: at the fixed point, "merely representational" is incoherent, because representation is faithful and the represented structure is exhausted by its representations. The Yoneda embedding is not an approximation. It is an isomorphism between an object and its relational profile — the complete pattern of arrows pointing at it.

Read through the dimensional lens, this says: the epistemic order (we discover dimensions by iterating M) and the ontological order (dimensions exist as the tower) converge at $D = 1$. The discovery process and the discovered structure are certified identical. This is the content of *Method* applied to dimensionality specifically: the method of generating dimensions (iterating M) and the structure that is generated (the dimensional tower) are the same thing, and Yoneda is the certificate.



The dissolution has a precise formal content. Before Yoneda, one might coherently ask whether the tower — built by iterating a meta-rule on grammars — is a representation of dimensional structure or the dimensional structure itself. After Yoneda, the question is answered: the Yoneda embedding from $\mathcal{C}$ to $[\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ is full and faithful, so the representational apparatus carries all information. At $D = 1$ — where the specification sustains itself — the representational apparatus is not merely faithful but is the only structure there is. No residual "real dimensionality" hides behind the representations.

Status: **Established.** Yoneda is proved. $D = 1$ is established. The dissolution is a consequence.

The Geometric Carrier and S^2

The epistemological path in *Method* lands on a 2-dimensional geometric carrier: S^2. The geometric-algebraic identification maps "minimal dimension" to "leastness." This paper sharpens the connection: S^2 is dimension 2 because the constraints (no boundary, simple connectivity, compactness, minimal dimension) are first jointly satisfiable at dimension 2. G_{Cat} is the least fixed point because M's iteration first stabilizes at the level where full categorical structure is present. These are the same "first jointly satisfiable" condition in different registers.

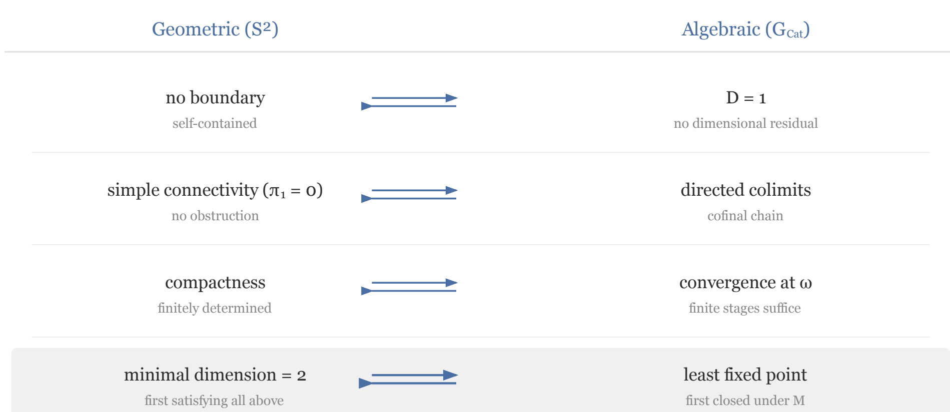
The four constraint pairs, read through the dimensional lens:

No boundary $\longleftrightarrow$ $D = 1$. Exact self-reproduction means no dimensional residual — the specification sustains itself without external input. No boundary means no edge where the structure fails. Both say: the thing is complete in itself. The tower at the fixed point has no "outside" from which additional dimensions could be supplied. S^2 has no edge from which additional geometry could leak in.

Simple connectivity $\longleftrightarrow$ directed colimits. Every loop on S^2 contracts because there are no holes ($\pi_1(S^2) = 0$). Every directed diagram in the Adámek chain has a colimit because the chain is cofinal. Both say: the completion step encounters no obstruction. The tower can be traversed without obstruction to dimensional closure. S^2 can be traversed without obstruction to topological closure.

Compactness $\longleftrightarrow$ convergent iteration. S^2 is covered by finitely many patches — the structure is finitely determined. The Adámek chain converges at ω — finitely many stages suffice to determine the colimit. Both say: the structure is generated by a finite process. No transfinite iteration is required on either side.

Minimal dimension $\longleftrightarrow$ leastness. 2 is the smallest dimension where all three conditions above hold simultaneously: S^1 fails simple connectivity ($\pi_1(S^1) = \mathbb{Z}$), and dimension 0 or 1 surfaces cannot simultaneously satisfy no boundary and simple connectivity. G_{Cat} is the least fixed point of M — the simplest theory closed under the dimension-generating operation. Both say: this is the first level where everything works.



The "first jointly satisfiable" framing connects the dimensional tower to the geometric carrier: both arrive at their respective objects by the same structural logic — the minimal thing satisfying all required closure conditions. The number 2 is not a coincidence. It is the dimension at which both systems first achieve completeness.

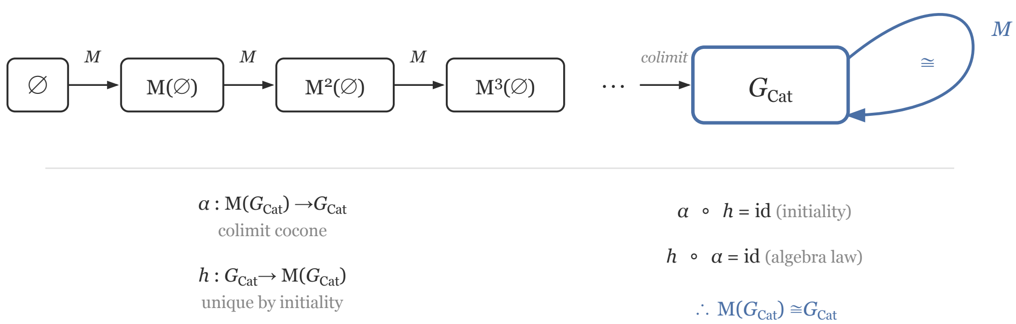
Status: **Derived.** The four pairs are in *Method*. The dimensional reading is new but follows from the established identifications. All four algebraic conditions ($D = 1$, directed colimits, convergence at ω, leastness) are machine-verified in the Lean companion.

The Collapse Principle

The preceding sections establish M as the unique dimension generator (§1–2), computation as its closure (§3), the dissolution of representation and being at the fixed point (§4), and the geometric carrier as the first jointly satisfiable structure (§5). Before turning to CPS and the terminal property, we state the abstract result that governs both.

ω-chain collapse. Let $F = (F_0 \xrightarrow{f_0} F_1 \xrightarrow{f_1} \cdots)$ and $G = (G_0 \xrightarrow{g_0} G_1 \xrightarrow{g_1} \cdots)$ be ω-chains in a cocomplete category $\mathcal{C}$. A *chain morphism* α is a family $\alpha_n : F_n \to G_n$ satisfying $\alpha_{n+1} \circ f_n = g_n \circ \alpha_n$ for all n. If $\operatorname{colim} F \cong \operatorname{colim} G \cong L$, then $\operatorname{colim}(\alpha) \in \operatorname{Aut}(L)$.

Initiality corollary. When F is the Adámek chain of M — so $F_n = M^n(\emptyset)$ and L is the initial M-algebra — the initial algebra has exactly one algebra endomorphism, and that endomorphism is the identity. So $\operatorname{colim}(\alpha) = \mathrm{id}_L$. The factoring is not merely unique; it is the identity. Every ω-chain reaching L reduces to M's chain.



the chain stabilizes: one more application of M produces nothing new
ω-chain collapse — colim(α) = id

This is the abstract content of the dimensional picture. M's tower is not one presentation of the fixed point among many. It is the presentation through which all others factor. Any process that builds L by iterating a successor operation, if it is related level-by-level to M's chain in a way that respects the successor, becomes literally identical to M's construction at the colimit. The construction path washes out. Only the colimit survives.

The principle reframes what follows. The CPS correspondence (§7) is an instantiation: define the CPS chain, construct the level-wise map to M's chain, verify the naturality square, invoke the corollary. The terminal property (§8) is the claim that every coherent notion of dimension increase that converges gives an ω-chain satisfying the hypotheses — with initiality guaranteeing the factoring. The individual cases (homotopy, homological, topological) are not independent pieces of evidence but instantiations of a single abstract fact.

Status: **Established, machine-verified.** The chain morphism framework, collapse map, uniqueness, and initiality corollary are in the Lean companion with 0 sorry.

The CPS Correspondence

A convergence from computation theory that illuminates the tower structure from the inside.

The precise claim first. In a cartesian closed category with reflexive object D, the continuation-passing style (CPS) transform at response type $R = D$ sends a morphism $f : A \to B$ to $\lambda a.\, \lambda k.\, k\,(f\,a) : A \to (B \to D) \to D$. When $R = D$ and $D \cong [D, D]$, the type $(B \to D) \to D$ simplifies: $(B \to D) \to D \cong [B, D] \to D$, and when B is itself absorbed into D (as happens at the fixed point where every type is a retract of D), this collapses to $D \to D \cong D$. Every type is a retract of D at the fixed point because $D \cong [D, D]$ means D contains all its own endomorphisms; any object B that appears in the CCC generated by D admits a section-retraction pair through D, since B's structure-preserving maps factor through D's self-maps via the reflexive isomorphism. The CPS-transformed type is D itself. The collapse is exactly what M does at the fixed point: it absorbs the dimensional content into the reflexive object.

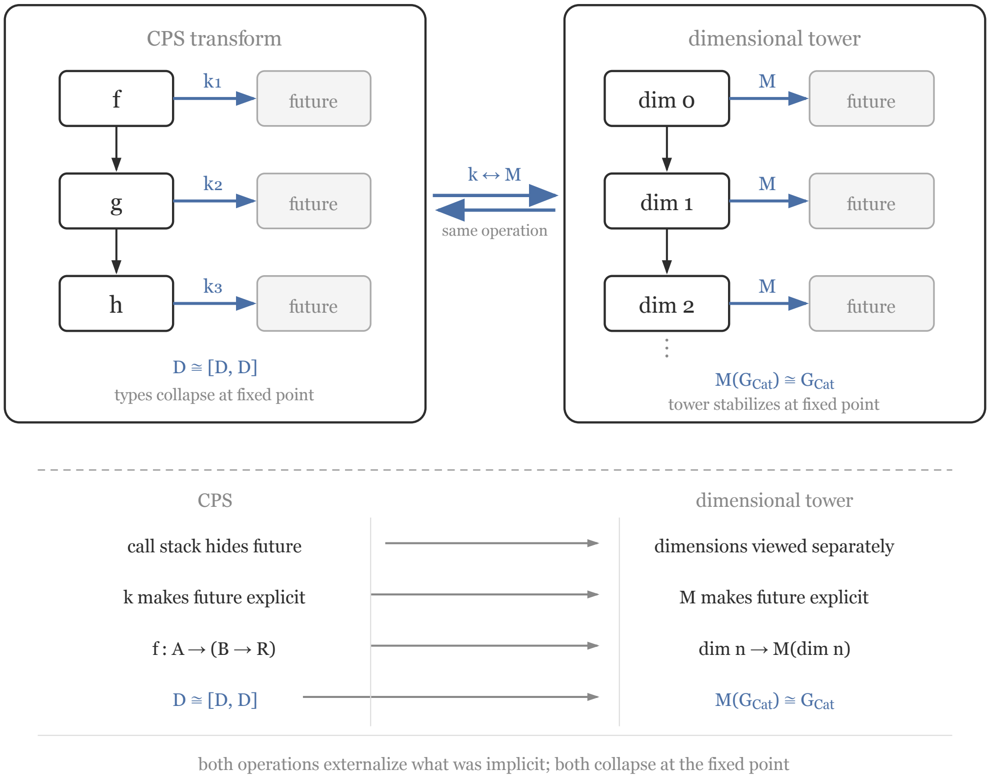
CPS does not just parallel M — it is what M looks like when restricted to the computational fragment. By 'the computational fragment' we mean the subcategory of structure visible within a single CCC with a distinguished reflexive object — types as objects, programs as morphisms — as opposed to the full EAT setting where objects are theories. CPS at the fixed point collapses every transformed type to D because dimensional completeness means every type is already absorbed into the reflexive object. The CPS transform makes the continuation — the future computational context — explicit. M makes the dimensional future — the next level of structure-preserving maps — explicit. Both operations externalize what was implicit, and both collapse at the fixed point because there is nothing left to externalize.

The structural parallels, now grounded by the precise claim:

Each dimension is the "future" of the dimension below, made explicit. A natural transformation (dimension 2) is what happens between two functors (dimension 1). A modification (dimension 3) is what happens between two natural transformations. Each level is the next level's continuation — the structured handler for how results at the lower level are consumed. M makes the dimensional future explicit, as CPS makes the computational future explicit (Sussman and Steele 1975).

Return-based computation hides interdependence. Functions appear modular, but the call stack couples them from the start — each function's result is consumed by its caller in a way that is invisible at the function's definition site. Similarly, viewing each categorical dimension in isolation (objects, then morphisms, then 2-morphisms) creates a fiction of independence. The tower shows the dimensions are mutually constitutive — each is defined in terms of the one below via M.

Effects are dimensional choices. In CPS, effects (failure, branching, suspension) are choices about how to handle the continuation — call it zero times, once, or many times. In the tower, the structural variations at each level (functors that preserve different amounts of structure) are choices about how the lower-dimensional structure is organized by the higher-dimensional one.



The CPS insight (Sussman and Steele 1975): computation is not about what you produce — it is about how you handle your context's claim on your result. The tower version: dimensionality is not about what exists at each level — it is about how each level serves as the context (continuation) for the level below.

What the diagram exhibits. The CPS tower and the dimensional tower are two ω-chains in the functor category $[\omega, \mathcal{C}]$. The level-by-level correspondence $k \leftrightarrow M$ is a natural transformation between them: it commutes with the respective successor operations (continuation-passing at each level maps to dimension increment at each level) and the two chains share a colimit (the reflexive object $D \cong [D, D]$, reached by CPS collapse on one side and dimensional closure on the other). At the colimit, the natural transformation collapses to an identity — both towers arrive at the same fixed point.

This is a direct instantiation of the ω-chain collapse theorem (§6). The CPS-dimensional correspondence satisfies the hypotheses: here are the two chains, here is the level-wise map ($k \leftrightarrow M$), it commutes with the successor, they share a colimit. By the initiality corollary, the factoring is the identity — the CPS tower reduces to M's chain.

The self-referential observation: the correspondence between the two towers is itself an instance of what M generates. M applied to a theory T computes the structure-preserving maps between models of T. The CPS tower and the dimensional tower are two models of "chain-with-future-operation," and the $k \leftrightarrow M$ correspondence is the structure-preserving map between them — precisely what $M(T)$ produces when T is the theory of such chains. The paper demonstrates an M-generated morphism without recognizing it as one. This is a 2-morphism: the kind of structure

M produces when you ask "what are the structure-preserving maps between these two models?" The CPS correspondence is not a computational shadow of M. It is a retract through M at one level up.

Status: **Structural correspondence, upgradeable to theorem.** The abstract ω-chain collapse result is pure colimit yoga, formalizable independently of any domain-specific content. The reflexive object, self-application, and fixed-point combinator that ground the CPS collapse are verified (T3, T4). The CPS transform itself (categorical encoding of Hasegawa 1995, Thielecke 1997) remains a formalization target. The self-referential identification — that the tower morphism lives in $M(T)$ — requires the EAT infrastructure and is deferred.

The Terminal Property (Open Question)

The strongest version of the claim: categorical dimension is the terminal notion of dimension. Every coherent notion of "dimension increase" factors through M.

Precisely: any endofunctor F on a suitable ambient category (monoidal closed, locally finitely presentable) that (a) increases structural complexity by exactly one level and (b) has a least fixed point is naturally isomorphic to the internal hom functor — i.e., is M.

This would mean M does not just generate categorical dimensions — it generates dimension as such. Every other notion of dimension (topological, homological, homotopy) would be a special case obtained by restricting the ambient category.

Applying the canonical presentation (§6). The evidence for the terminal property has a uniform shape. In each case, a domain-specific notion of dimension gives rise to an ω-chain reaching the same colimit as M's chain. By the ω-chain collapse theorem, each such chain factors through M's — and by the initiality corollary, the factoring is the identity. Each specific instance reduces to verifying the hypotheses: here is the chain, here is the level-wise map, it commutes with the successor. Check, check, done.

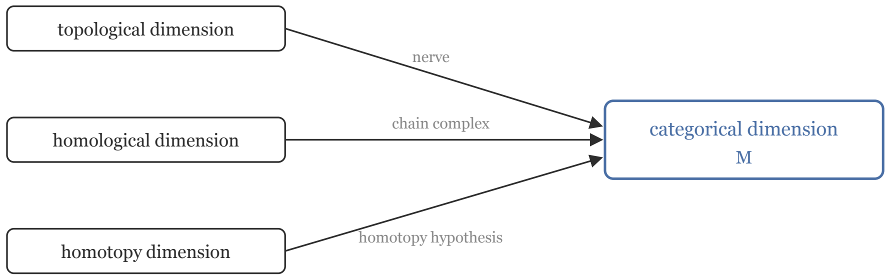
Evidence. Four instantiations:

CPS. The CPS tower at response type $R = D$ and the dimensional tower are two ω-chains with the $k \leftrightarrow M$ correspondence as the natural transformation (§7). They share a colimit. This is the strongest evidence because it shows even the *internal computational* view of dimension maps back through M — it is not a shadow but a retract through M at one level up.

Homotopy dimension. n-categories subsume n-groupoids (n-categories with all morphisms invertible). By the homotopy hypothesis (Grothendieck 1983), n-groupoids model homotopy n-types. The factoring: homotopy dimension $n \to n$-groupoid $\to n$-category $\to M^n(\emptyset)$.

Homological dimension. Chain complexes in abelian categories carry dimension (chain length). An abelian category is a category; chain complexes are diagrams within it. The factoring: homological dimension $\to$ chain length $\to$ categorical structure $\to M$.

Topological dimension. Lebesgue covering dimension is recovered via nerves: the nerve of a good cover is a simplicial complex whose dimension equals the covering dimension. Simplicial sets are presheaves on Δ. The simplicial category Δ — whose objects are finite ordinals and whose morphisms are order-preserving maps — is itself a category, hence an instance of M's output at level 1. Its nerve construction is a functor into presheaves, and the dimensional grading of simplicial sets (the maximal dimension of non-degenerate simplices) recovers covering dimension. The factoring: covering dimension $\to$ simplicial dimension $\to$ nerve $\to$ presheaves on $\Delta \to M$.



In each case, the specific notion of dimension is obtained by restricting the general dimensional machinery (M and the tower) to a particular ambient setting. The claim that M is terminal would mean: there is no coherent notion of dimension increase that escapes this pattern. The abstract ω-chain theorem reduces each case to a hypothesis check; what varies is the domain-specific construction of the chain and the level-wise map.

Status: **Conjectural.** The abstract framework (§6) is derived; the terminal property itself is conjectural. Each instantiation is a verification that the specific pair of chains satisfies the hypotheses. Proving the terminal property would establish M as the universal dimension generator. Disproving it would require exhibiting a coherent notion of dimension increase that does not factor through the categorical dimensional ladder.

What This Paper Establishes and What Remains

Established:

- M generates exactly one categorical dimension per application (§1)
- M is the unique such operation (initiality) (§2)
- Computation follows from dimensional completeness (Lambek at the colimit) (§3)
- At $D = 1$, the representational ladder is the dimensional structure (Yoneda) (§4)
- The epistemic and ontological orders converge at the fixed point (§4)
- S^2 and G_{Cat} are "first jointly satisfiable" in their respective registers (§5)

Established, machine-verified:

- Every ω-chain reaching the fixed point factors through M's chain; the factoring is the identity (§6)

Derived:

- Computation is constructively posterior to dimensionality (chain vs. colimit) (§3)
- The CPS correspondence: forwarding = dimensional continuation; CPS at $R = D$ collapses via reflexivity (§7)
- The CPS-dimensional correspondence is an instantiation of §6, collapsing at the shared colimit (§7)

Conjectured:

- M is the terminal dimension-generating operation (§8)

- Every coherent notion of dimension factors through M via §6 (§8)
- The CPS-dimensional tower morphism lives in $M(T)$ for the theory of chains-with-future-operations (§7)

What this enables. The dimensional reading gives a *criterion* for when a formal system supports universal computation: check whether its dimensional tower converges. Instead of constructing the reflexive object $D \cong [D, D]$ directly — which requires solving a fixed-point equation in the ambient category — one checks whether the Adámek chain $\emptyset \to M(\emptyset) \to M^2(\emptyset) \to \cdots$ has a colimit and whether M preserves it. Dimensional convergence is checkable from the chain alone, without reference to the colimit's internal structure. This converts the question "does this category support computation?" from a fixed-point problem to a convergence problem — and convergence conditions (local finite presentability, accessibility of M, preservation of directed colimits) are well-understood and often verifiable. The convergence criterion is formally defined: dimension (T1), increment (T2), and stabilization (T2.5) are machine-verified, and the implication from convergence to computation (T3+T4) is a theorem.

The FinSet divergence and the thin-category failure (§3) are instances of this criterion in the negative direction: the dimensional tower diagnoses *why* these categories lack computation, not just *that* they do. FinSet fails because $|D^D| > |D|$ forces unbounded dimensional growth. Thin categories fail because the internal hom is too constrained. In both cases, the dimensional diagnosis is more informative than the bare statement "no reflexive object exists."

Lean 4 companion. Every section of this paper except the conjectural material (§8) and the exploratory CPS correspondence (§7) is backed by machine-checked proofs. The formalization covers: dimension as truncation level and the proof that M increments dimension by exactly one (§1); the uniqueness scaffold with the Boardman-Vogt tensor product as the standing open item (§2); dimension stabilization at the fixed point — the dimensional reading of $D = 1$ (§3); the reflexive object, self-application map, and fixed-point combinator derived from Lambek's isomorphism (§3); the self-indexed computation layer — naming equivalence, universal evaluator, Kleene recursion theorem, and the abstract fixed-point property (§3); and the full chain from Adámek through Lambek to substrate-independent existence and uniqueness (§1–§4). Total: 0 sorry and 0 custom axioms in the dimensional and categorical components. The project has 0 sorry and 0 custom axioms across 42 files. The Boardman-Vogt tensor extension (BoardmanVogt.lean) states conjectures as weak placeholders with no downstream dependency — the Gabriel-Ulmer gap is the standing open item across the series. The terminal property (§8) is an open conjecture. The ω-chain collapse theorem and initiality corollary (§6) are machine-verified with 0 sorry — the chain morphism framework, collapse map, uniqueness, and initiality corollary are in the Lean companion. The CPS instantiation (§7) awaits encoding of the categorical CPS literature.

A note on scope. This paper extracts a consequence implicit across the series but never stated as a thesis: the tower is not about categories — it is about dimensions. Categories are the specific structure at each level; dimensionality is the structural fact that there are levels at all. §1–2 establish M as the unique dimension generator. §3 shows computation is derived from dimensional completeness. §4 dissolves the representation/being distinction at the fixed point. §5 connects the dimensional tower to the geometric carrier via "first jointly satisfiable" structure. §6 states the collapse principle (machine-verified): every chain reaching the fixed point factors through M, and the factoring is the identity. §7 illuminates the tower from inside computation via CPS as an instantiation of §6. §8 states the open question. The paper depends on the *Grammar*, *Computation*, *Elements*, and *Method* papers. It does not depend on the *Adjunction* paper's Ψ analysis. The core identification — M is dimensionality, computation is its closure — is derivable from existing results; what is new is recognizing it as a thesis.

Crystal

Larsen James Close

An endofunctor M on a category with initial object $\emptyset$ generates an ω-chain:

$$\emptyset \;\to\; M(\emptyset) \;\to\; M^2(\emptyset) \;\to\; \cdots$$

This chain is initial: every M-generated chain receives a unique morphism from it.

If the chain converges (colimit L exists, M preserves it): Lambek gives $M(L) \cong L$. Each stage $M^n(\emptyset)$ has dimension n. L has no finite dimension. Dimension stabilizes at the fixed point.

When $M = \mathrm{ihom}(A)$ in a monoidal closed category: $L \cong [A, L]$. The Lambek iso is a container boundary — applying M does not escape L. Self-application and the fixed-point combinator are constructed from the Lambek iso and the evaluation morphism. When $A = L$: $L \cong [L, L]$ and L names its own endomorphisms.

The Lambek iso is identity modulation: forward $[L, L] \to L$ folds a function into a datum, inverse $L \to [L, L]$ unfolds a datum into a function. This fold/unfold, combined with composition, is application, abstraction, and β-reduction — the untyped lambda calculus. The fixed point is a model of universal computation. No external enumeration is needed.

Any two M-generated ω-chains collapse at a shared colimit: the induced morphism is unique. The collapse from a chain to itself is the identity; between distinct chains sharing a colimit, it is the initiality iso.

Four levels, each visible only when the ambient category provides it:

1. **Dimension.** Endofunctor + initial object. Chain, grading, tower initiality.
2. **Convergence.** Colimit exists, M preserves it. Lambek. Dimension stabilizes.
3. **Closure.** $M = \mathrm{ihom}(A)$. Containerization. Self-application. Omega.
4. **Computation.** $A = L$. Identity modulation. Lambda calculus. Universal computation.

The chain is prior to its colimit. Initiality is the certificate. Each level is not added to the previous — it is what the previous level becomes when the ambient category makes it visible.

References

Adámek, Jiří. 1974. "Free Algebras and Automata Realizations in the Language of Categories." *Commentationes Mathematicae Universitatis Carolinae* 15 (4): 589–602.

Adámek, Jiří, and Jiří Rosický. 1994. *Locally Presentable and Accessible Categories*. London Mathematical Society Lecture Note Series 189. Cambridge University Press. https://doi.org/10.1017/CBO9780511600579.

Boardman, J. Michael, and Rainer M. Vogt. 1973. *Homotopy Invariant Algebraic Structures on Topological Spaces*. Vol. 347. Lecture Notes in Mathematics. Springer-Verlag.

Chomsky, Noam. 1956. "Three Models for the Description of Language." *IRE Transactions on Information Theory* 2 (3): 113–24. https://doi.org/10.1109/TIT.1956.1056813.

Close, Larsen James. 2026. *Phase-Indexed Epistemology*. Zenodo. https://doi.org/10.5281/zenodo.18812866.

Close, Larsen James. n.d. "Recognizing Substrate-Agnosticism." Unpublished manuscript.

Eilenberg, Samuel. 1974. *Automata, Languages, and Machines*. A. Pure and Applied Mathematics 59. Academic Press.

Eilenberg, Samuel, and John C. Moore. 1965. "Adjoint Functors and Triples." *Illinois Journal of Mathematics* 9 (3): 381–98. https://doi.org/10.1215/ijm/1256068141.

Freyd, Peter. 1972. "Aspects of Topoi." *Bulletin of the Australian Mathematical Society* 7 (1): 1–76.

Gabriel, Peter, and Friedrich Ulmer. 1971. *Lokal Präsentierbare Kategorien*. Vol. 221. Lecture Notes in Mathematics. Springer. https://doi.org/10.1007/BFb0059396.

Grothendieck, Alexander. 1983. "Pursuing Stacks." Unpublished manuscript.

Hasegawa, Masahito. 1995. "Decomposing Typed Lambda Calculus into a Couple of Categorical Programming Languages." *Category Theory and Computer Science, 6th International Conference*, Lecture notes in computer science, vol. 953: 200–219.

Hyland, Martin, and John Power. 2007. "The Category Theoretic Understanding of Universal Algebra: Lawvere Theories and Monads." *Electronic Notes in Theoretical Computer Science* 172: 437–58. https://doi.org/10.1016/j.entcs.2007.02.019.

Lambek, Joachim. 1968a. "A Fixpoint Theorem for Complete Categories." *Mathematische Zeitschrift* 103: 151–61.

Lambek, Joachim. 1968b. "Deductive Systems and Categories I." *Mathematical Systems Theory* 2: 287–318. https://doi.org/10.1007/BF01703261.

Lambek, Joachim. 1969. "Deductive Systems and Categories II: Standard Constructions and Closed Categories." In *Category Theory, Homology Theory and Their Applications I*, vol. 86. Lecture Notes in Mathematics. Springer.

Lambek, Joachim, and Philip J. Scott. 1986. *Introduction to Higher Order Categorical Logic*. Cambridge Studies in Advanced Mathematics 7. Cambridge University Press.

Lawvere, F. William. 1963. "Functorial Semantics of Algebraic Theories." PhD thesis, Columbia University.

Lawvere, F. William. 1969. "Diagonal Arguments and Cartesian Closed Categories." *Lecture Notes in Mathematics* 92: 134–45.

Lindström, Per. 1969. "On Extensions of Elementary Logic." *Theoria* 35: 1–11.

Linton, F. E. J. 1966. "Some Aspects of Equational Categories." In *Proceedings of the Conference on Categorical Algebra, La Jolla 1965*, edited by S. Eilenberg, D. K. Harrison, S. MacLane, and H. Röhrl. Springer.

Mac Lane, Saunders. 1998. *Categories for the Working Mathematician*. 2nd ed. Graduate Texts in Mathematics 5. Springer.

Myhill, John. 1955. "Creative Sets." *Zeitschrift Für Mathematische Logik Und Grundlagen Der Mathematik* 1: 97–108.

Nishizawa, Koki, and John Power. 2009. "Lawvere Theories Enriched over a General Base." *Journal of Pure and Applied Algebra* 213 (3): 377–86. https://doi.org/10.1016/j.jpaa.2008.07.009.

Reiterman, Jan. 1982. "The Birkhoff Theorem for Finite Algebras." *Algebra Universalis* 14: 1–10. https://doi.org/10.1007/BF02483902.

Scott, Dana S. 1976. "Data Types as Lattices." *SIAM Journal on Computing* 5 (3): 522–87. https://doi.org/10.1137/0205037.

Sussman, Gerald Jay, and Guy Lewis Steele Jr. 1975. *Scheme: An Interpreter for Extended Lambda Calculus*. AI Memo 349. MIT Artificial Intelligence Laboratory.

Thielecke, Hayo. 1997. “Categorical Structure of Continuation Passing Style.” PhD thesis, University of Edinburgh.