PhilLean: A Type-Theoretic Framework for Computer-Auditable Philosophy

Core Metalanguage, Formal Theories of Wanting, and the Noetic–Noematic Commutation Problem

Working research manuscript

August 26, 2026

Abstract

This paper proposes PhilLean, a framework for representing philosophical arguments in a form that can be mechanically audited by a proof assistant such as Lean without reducing philosophical correctness to theoremhood in a single background logic. The central architectural idea is to separate (i) semantic analysis, (ii) interpretation into an explicitly declared object logic, (iii) proof and model checking, and (iv) dialectical structure. The proposed core therefore treats concepts, explications, readings, claims, argument schemas, object logics, interpretations, models, theories, and objections as typed objects. Philosophical formulas are not identified with Lean's native propositions; rather, Lean's trusted kernel verifies metatheoretic claims such as $\Gamma \vdash_L \varphi$, thereby insulating philosophical object logics from accidental inheritance of the host logic.

The framework is tested on a formal theory of wanting developed within a Husserlian intentional vocabulary. Ordinary wanting, endorsed wanting, higher-order wanting, and proleptic wanting are separated formally. Endorsed wanting is shown to imply wanting by definition, while the converses and the relation between wanting and wanting-to-want admit finite countermodels. The analysis then introduces distinct partial orders of noetic refinement (refinement of the act-side of intentionality) and noematic refinement (refinement of intentional content), separating refinement from higher-order extension. Intentional states are represented as a dependent family of noeses over noemata. This permits a precise statement of the noetic–noematic commutation problem. A naturality theorem characterizes commutation: noetic refinement commutes with noematic refinement exactly when the family of noetic refinement maps is natural with respect to noematic transport. Independent coordinates commute automatically, but a finite five-noesis countermodel proves that commutation is not forced by the basic refinement axioms. The resulting noncommutation provides a formal notion of path-dependent and potentially transformative desire. The paper concludes with an implementation plan and a proposal for PhilLean as a general methodology for computer-auditable philosophy.

Contents

1. Introduction

Formal methods have transformed large areas of mathematics and computer science by separating the act of discovering a proof from the act of checking whether the proof is valid. A natural question is whether an analogous separation is possible in philosophy. Philosophical arguments are often presented in ordinary language, rely on multiple forms of inference, employ contested concepts, and turn on hidden bridge principles. Consequently a system for "checking philosophy" cannot simply translate prose into a single classical logic and return a Boolean verdict. The more useful goal is a system that makes the structure of an argument explicit and then certifies exactly what follows *relative to declared formal commitments.*

We call the proposed framework PhilLean: "Lean for Philosophy." The name is programmatic rather than restrictive. Lean is an attractive implementation target because it is an interactive theorem prover based on dependent type theory and a small proof-checking kernel [1, 2]. But the central ideas do not depend on Lean specifically. A successful implementation could combine a trusted proof kernel with automated theorem provers, SAT/SMT solvers, modal model finders, and domain-specific interfaces.

The proposal has three main intellectual antecedents. First, Natural Semantic Metalanguage (NSM), associated especially with Anna Wierzbicka and Cliff Goddard, aims to analyze complex meanings through reductive paraphrases built from a small cross-linguistically universal inventory of semantic primes. The mature program proposes 65 primes, and notably includes WANT among them [3, 4]. Second, Edward Zalta and collaborators have developed a program of "computational metaphysics" in which axiomatic metaphysical theories are represented in automated or interactive theorem provers. This work is not merely aspirational: automated formalization has exposed an error in an earlier theorem concerning Plato's Forms, and mechanization of Abstract Object Theory in Isabelle/HOL revealed a previously overlooked paradox involving complex terms and comprehension [5, 6, 7]. Third, the present work extends a series of experimental formal models of interacting Husserlian monads and finite overminds developed in companion manuscripts [10, 11].

The new development has two intertwined aims. The first is general: define a core type theory for computer-auditable philosophical arguments. The second is substantive: use that framework to formalize kinds of wanting, especially proleptic, higher-order, and endorsed wanting, and to investigate the relation between noetic and noematic refinement. The latter problem produces a genuine mathematical result: these two refinement operations commute under a naturality condition, but they need not commute in general.

2. What a Philosophy Checker Should Check

Suppose a philosopher presents premises $P_1, \dots, P_n$ and a conclusion C. At minimum, a useful system should distinguish the following questions:

Derivability:	$P_1, \dots, P_n \vdash_L C$,
Semantic validity:	$P_1, \dots, P_n \models_{\mathcal{M}} C$,
Consistency:	$\exists M\ M \models P_1 \wedge \cdots \wedge P_n$,
Independence:	$P_1, \dots, P_n \nvdash_L C$,
Countermodel existence:	$\exists M\ M \models P_1, \dots, P_n, \neg C$,
Premise dependence:	which subsets of the premises suffice for C?

The system should also track interpretive and dialectical dependence. An inference may be deductively valid under one interpretation of "free," "cause," or "knowledge" but invalid under another. An argument may be valid in S5 but not S4, classically but not intuitionistically, or only after a substantive bridge principle has been added. Hence the correct top-level judgment is not

$$\mathsf{Valid}(A),$$

but rather

$$\mathsf{Valid}_{L,I}(A),$$

meaning that argument A is valid under interpretation I in object logic L.

This relativity does not make the system weaker. On the contrary, it turns philosophical disagreement into something auditable. A proof certificate can state which logic, formalization, definitions, bridge principles, and premises were required. A failed proof attempt is not enough; where possible the system should construct a countermodel.

3. Three Metalanguages

A central lesson of the present investigation is that "the metalanguage of philosophy" is not one thing. At least three layers are needed.

3.1. Semantic metalanguage

The semantic layer concerns what expressions mean before they are formalized into a logic. NSM is relevant here because it distinguishes semantic primitives from reductive explications. A key design insight is that PhilLean need not pretend that every concept can or should be formally defined. It can allow a semantic system in which some concepts are declared primitive. Thus an NSM-inspired semantic system may take

WANT

as primitive while allowing formally structured species of wanting to be built from it.

3.2. Logical metalanguage

The logical layer gives typed syntax, proof theory, model theory, and formal interpretation. Zalta's work demonstrates the value of this layer for philosophy. It also provides a warning: if a philosophical object theory is embedded in a richer host logic, one must prevent "artifactual theorems" that arise from resources belonging only to the host representation. Our solution is architectural insulation: philosophical formulas belong to an explicit object logic, and Lean propositions express metatheoretic claims about that logic.

3.3. Dialectical metalanguage

The dialectical layer records theories, arguments, objections, replies, hidden assumptions, counterexamples, and alternative interpretations. This is the least developed layer in existing proof assistants and is one of the distinctive goals of PhilLean. An objection should be typed by what it attacks: a premise rejection differs from a countermodel, an equivocation objection, or a challenge to the underlying logic.

4. The Core Type Theory

4.1. Typed syntax

Both semantic languages and object logics require typed expressions. We begin with an abstract typed syntax.

```
structure TypedSyntax where
  Ty : Type
  prop : Ty
  arr : Ty -> Ty -> Ty

  Expr : List Ty -> Ty -> Type

  var : Var Gamma tau -> Expr Gamma tau
  app : Expr Gamma (arr sigma tau) -> Expr Gamma sigma -> Expr Gamma tau
  lam : Expr (sigma :: Gamma) tau -> Expr Gamma (arr sigma tau)

  rename : Renaming Gamma Delta -> Expr Gamma tau -> Expr Delta tau
  subst : Substitution Gamma Delta -> Expr Gamma tau -> Expr Delta tau
```

The intended judgment $\Gamma \vdash e : \tau$ is represented intrinsically by the type of e. A category mistake is therefore not merely a false proposition; an expression that applies a predicate of persons to a possible world may be unconstructible.

4.2. Semantic systems and concepts

A semantic system adds concepts to typed syntax.

```
structure SemanticSystem where
  syntax : TypedSyntax

  Concept : Type
  ctype : Concept -> syntax.Ty
  atom : (c : Concept) -> syntax.Expr [] (ctype c)

  support : syntax.Expr Gamma tau -> Finset Concept
  Prime : Concept -> Prop
  rank : Concept -> Nat
```

A concept is therefore not intrinsically a proposition. Examples include individual concepts, predicates, relations, and operators on propositions:

$$\begin{aligned} \mathsf{GOOD} &: \mathsf{Thing} \to \mathsf{Prop}, \\ \mathsf{WANT} &: \mathsf{Agent} \to \mathsf{Noema} \to \mathsf{Prop}, \\ \mathsf{BECAUSE} &: \mathsf{Prop} \to \mathsf{Prop} \to \mathsf{Prop}. \end{aligned}$$

A particular semantic system may assert $\mathsf{Prime}(\mathsf{WANT})$ without making this a theorem of PhilLean itself.

4.3. Explication versus definition

An important philosophical distinction is between analyzing an existing concept and stipulating an abbreviation.

Definition 4.1 (Explication)**.** *An explication of concept c in semantic system S is a typed expression E of the same semantic type as c, together with metadata stating that E is proposed as a semantic analysis of c. A reductive explication may additionally require every nontrivial concept in its support to have lower semantic rank.*

Definition 4.2 (Stipulative definition)**.** *A definition introduces a new or local symbol as an abbreviation for a typed expression. A well-formed definition environment has unique targets and an acyclic dependency graph, so that definitional expansion terminates.*

Thus the statement "knowledge means justified true belief" cannot acquire theorem status merely by being labelled a definition if it is intended as a substantive analysis of the pre-existing concept of knowledge. By contrast, a symbol introduced purely for convenience should be eliminable without adding philosophical commitments.

4.4. Readings and semantic claims

Ordinary-language sentences may admit multiple readings.

```
structure Reading (S : SemanticSystem) where
  surface : String
  meaning : S.syntax.Expr [] S.syntax.prop
```

This deliberately makes the map from sentence to semantics one-to-many. Ambiguity is represented rather than suppressed.

A semantic claim adds epistemic and provenance information:

```
structure SemanticClaim (S : SemanticSystem) where
  reading : Reading S
  status : ClaimStatus
  provenance : Provenance
```

Possible statuses include asserted premise, hypothesis, empirical premise, conceptual principle, phenomenological datum, and theorem. These tags do not change logical consequence; they record why a premise is being admitted.

4.5. Argument schemas

A philosophical argument can now be represented before a target logic is chosen.

```
structure ArgumentSchema (S : SemanticSystem) where
  definitions : DefEnv S
  premises : List (SemanticClaim S)
  conclusion : SemanticClaim S
```

This is a central architectural move. One argument schema may receive several formalizations, and interpretive robustness can be assessed across them.

4.6. Object logics

An object logic contains its own typed syntax, proofs, models, and satisfaction relation.

```
structure Logic where
  syntax : TypedSyntax

  Formula := syntax.Expr [] syntax.prop
  Proof : List Formula -> Formula -> Type

  Model : Type
  Sat : Model -> Formula -> Prop

  sound : Proof Gamma phi ->
        (forall M, Satisfies M Gamma -> Sat M phi)
```

The essential point is that the object-level formula type is not Lean's native `Prop`. Lean checks terms inhabiting statements such as

$$\mathsf{Proof}_L(\Gamma, \varphi),$$

rather than silently treating φ as a native Lean proposition.

4.7. Interpretations as typed morphisms

An interpretation $I : S \to L$ maps semantic types and expressions into the object logic while preserving structure.

```
structure Interpretation (S : SemanticSystem) (L : Logic) where
  mapTy : S.syntax.Ty -> L.syntax.Ty

  map_prop : mapTy S.syntax.prop = L.syntax.prop
  map_arr : mapTy (S.syntax.arr a b)
          = L.syntax.arr (mapTy a) (mapTy b)

  mapExpr : S.syntax.Expr Gamma tau ->
         L.syntax.Expr (mapCtx mapTy Gamma) (mapTy tau)

  preservesSubst : ...
```

The key coherence condition is

$$I(e[\sigma]) = I(e)[I(\sigma)].$$

Interpretation is therefore not an arbitrary dictionary but a structure-preserving translation.

4.8. Formalization and validity

Let D be the local definitional environment of an argument A. The formal image of a claim p under interpretation I is

$$[\![p]\!]_I = I\big(\mathrm{expand}_D(p.\mathsf{meaning})\big).$$

Validity is then

$$\mathsf{Valid}_{L,I}(A) :\Longleftrightarrow \exists d : \mathsf{Proof}_L\big([\![P_1]\!]_I, \ldots, [\![P_n]\!]_I; [\![C]\!]_I\big).$$

Semantic validity is

$$\mathsf{SemValid}_{L,I}(A) :\Longleftrightarrow \forall M\, [(\forall i\ M \models_L [\![P_i]\!]_I) \Rightarrow M \models_L [\![C]\!]_I]\,.$$

Consistency is

$$\mathsf{Consistent}_{L,I}(A) :\Longleftrightarrow \exists M\ \forall i\ M \models_L [\![P_i]\!]_I.$$

4.9. Countermodels

A countermodel is a certified object rather than a textual warning.

```
structure Countermodel
    (A : ArgumentSchema S)
    (I : Interpretation S L) where
  model : L.Model
  premisesTrue : forall p in formalPremises A I, L.Sat model p
  conclusionFalse : not L.Sat model (formalConclusion A I)
```

Theorem 4.3 (Countermodel refutation)**.** *If L is sound and there exists a countermodel to (A, I), then $\neg\mathsf{Valid}_{L,I}(A)$.*

Proof. Suppose a proof object existed. Soundness would make the conclusion true in every model satisfying the premises, including the countermodel. This contradicts the countermodel's certified falsification of the conclusion. □

4.10. Theories and objections

A philosophical theory may be represented by explicit commitments plus definitions. It is useful to distinguish claims explicitly encoded by the theory from their deductive consequences. Write

$$T \triangleright p$$

for "p is an explicit commitment of T," and

$$T \Vdash_{I,L} p$$

for "p follows from the commitments of T under (I, L)." Generally,

$$T \triangleright p \Rightarrow T \Vdash p, \qquad T \Vdash p \not\Rightarrow T \triangleright p.$$

This distinction is structurally analogous to lessons drawn from Zalta's distinction between what an abstract object encodes and what is true of it, without requiring Abstract Object Theory as PhilLean's foundational ontology.

Objections should also be typed. A countermodel objection must supply a countermodel; an equivocation objection must identify two occurrences together with evidence of divergent interpretations; a logic challenge supplies an alternative object logic or embedding; a premise rejection targets a specific premise.

5. A First Case Study: Varieties of Wanting

We now instantiate the framework with a deliberately small theory of wanting. The purpose is not to settle the phenomenology of desire but to show how subtle conceptual relations become mechanically explicit.

5.1. Ordinary wanting and endorsement

Let

$$\mathsf{Want}(a, p)$$

mean that agent a wants noema p. Let $\omega(a, p)$ denote the wanting act itself, represented as an object available to higher-order attitudes. Let

$$\mathsf{Endorse}(a, \omega(a, p))$$

mean that a endorses that wanting.

Define endorsed wanting by

$$\boxed{\mathsf{EWant}(a, p) :\Longleftrightarrow \mathsf{Want}(a, p) \wedge \mathsf{Endorse}(a, \omega(a, p)).}$$

Theorem 5.1 (Endorsed wanting implies wanting)**.**

$$\mathsf{EWant}(a, p) \to \mathsf{Want}(a, p).$$

Proof. Immediate by conjunction elimination. □

The converse does not follow. A one-agent, one-noema model with $\mathsf{Want}(a, p)$ true and $\mathsf{Endorse}(a, \omega(a, p))$ false is a countermodel. Thus endorsed wanting is a proper refinement of ordinary wanting whenever unendorsed wants are possible.

5.2. Higher-order wanting

Let $\langle \mathsf{Want}(a, p) \rangle$ be a noematic representation of the state in which a wants p. Define

$$\mathsf{WWant}(a, p) :\Longleftrightarrow \mathsf{Want}(a, \langle \mathsf{Want}(a, p) \rangle).$$

This represents wanting to want p.

Neither direction of implication with ordinary wanting is derivable in the minimal theory:

$$\mathsf{WWant}(a, p) \not\Rightarrow \mathsf{Want}(a, p), \qquad \mathsf{Want}(a, p) \not\Rightarrow \mathsf{WWant}(a, p).$$

The first countermodel represents an agent who wants to acquire a desire not presently possessed. The second represents an agent who has a first-order desire without any second-order desire to possess it.

Likewise endorsement and wanting-to-want are distinct:

$$\mathsf{EWant}(a, p) \not\Rightarrow \mathsf{WWant}(a, p), \qquad \mathsf{WWant}(a, p) \not\Rightarrow \mathsf{EWant}(a, p).$$

This is philosophically important because "identifying with a desire," "approving of it," and "wanting to have it" are often conflated.

5.3. Proleptic wanting and refined noemata

A first treatment of proleptic wanting introduces refined noemata q and a projection

$$\pi : \mathsf{ProNoema} \to \mathsf{Noema}.$$

If proleptic wanting is merely defined as ordinary wanting of the projected content plus a proleptic condition, then it trivially implies wanting. But if refined wanting is taken as primitive, the implication

$$\mathsf{ProWant}(a, q) \to \mathsf{Want}(a, \pi q)$$

requires an additional projection principle. This ambiguity motivates a deeper analysis: prolepsis may refine not only the intended content but the manner of intending. We therefore turn to the noesis/noema distinction.

6. Noetic and Noematic Refinement

Husserlian phenomenology distinguishes analysis of the act of consciousness (noesis) from analysis of its intentional correlate (noema), although interpretations of the exact status of the noema vary substantially [8, 9]. Our use is deliberately structural: "noetic" refers to act-side determination, while "noematic" refers to determination of intentional content. The formalism is intended to be compatible with multiple philosophical interpretations of the noema.

6.1. Intentional acts

Introduce types

$$\mathsf{Noesis}, \quad \mathsf{Agent}, \quad \mathsf{Noema}, \quad \mathsf{Mode}$$

and functions

$$\mathrm{subj} : \mathsf{Noesis} \to \mathsf{Agent}, \qquad \mathrm{mode} : \mathsf{Noesis} \to \mathsf{Mode}, \qquad \mathrm{noema} : \mathsf{Noesis} \to \mathsf{Noema}.$$

Ordinary wanting is no longer primitive as a binary relation. Rather,

$$\mathsf{Want}(a, p) :\Longleftrightarrow \exists n : \mathsf{Noesis}\ [\mathrm{subj}(n) = a \land \mathrm{mode}(n) = \mathsf{want} \land \mathrm{noema}(n) = p].$$

This creates room for act-side structure.

6.2. Resolution systems

Let $\mathcal{R}_N$ be a preorder of noetic resolutions. For each resolution r, let Noesis_r be the type of acts represented at that resolution. If $r \leq s$, let

$$\rho^N_{s,r} : \mathsf{Noesis}_s \to \mathsf{Noesis}_r$$

forget the additional noetic information. Require

$$\rho^N_{r,r} = \mathrm{id}, \qquad \rho^N_{t,r} = \rho^N_{s,r} \circ \rho^N_{t,s}.$$

Definition 6.1 (Noetic refinement)**.** *A noesis n' noetically refines n, written $n' \succeq_N n$, when n' occurs at a finer resolution and projects to n under the appropriate forgetting map.*

This is a preorder. Quotienting by mutual refinement yields a partial order of informational equivalence classes.

Noematic refinement is defined analogously with a resolution preorder $\mathcal{R}_M$ and forgetting maps

$$\rho^M_{s,r} : \mathsf{Noema}_s \to \mathsf{Noema}_r.$$

An intentional state can therefore be indexed by two resolution coordinates, one noetic and one noematic.

6.3. Pure noetic refinement

A pure noetic refinement preserves subject, intentional mode, and noema while enriching act-side determination. Thus legitimate pure refinement obeys

$$n' \succeq_N n \Rightarrow \mathrm{subj}(n') = \mathrm{subj}(n), \tag{1}$$
$$n' \succeq_N n \Rightarrow \mathrm{mode}(n') = \mathrm{mode}(n), \tag{2}$$
$$n' \succeq_N^{\mathrm{pure}} n \Rightarrow \mathrm{noema}(n') = \mathrm{noema}(n). \tag{3}$$

Consequently a proleptic want that differs from an ordinary want only by act-side temporal structure projects to an ordinary want without an extra bridge axiom.

Theorem 6.2 (Noetic projection theorem)**.** *Let $n' \succeq_N^{\mathrm{pure}} n$. If n' is a wanting act, then n is a wanting act. In particular, if proleptic wanting is defined as wanting plus a proleptic noetic feature, then every proleptic refinement projects to an ordinary want.*

Proof. By preservation of intentional mode under pure noetic refinement, $\mathrm{mode}(n) = \mathrm{mode}(n') = \mathsf{want}$; subject and noema are likewise preserved. □

6.4. Refinement is not extension

Endorsement and higher-order wanting reveal that adding intentional structure is not always refinement of a single act. If e is an endorsement act directed at wanting act n, then the endorsed configuration contains two acts and a relation

$$\mathsf{endorses}(e, n).$$

This is better represented as a noetic extension or structured complex than as a refinement of n itself.

Likewise, wanting-to-want introduces a new wanting act whose noema represents another actual or possible wanting act. It is a higher-order intentional relation, not a refinement of the lower-order act. Thus the noetic calculus needs at least four relations:

$$\begin{aligned} n' \succeq_N n &\quad \text{noetic refinement,} \\ p' \succeq_M p &\quad \text{noematic refinement,} \\ n \lhd X &\quad \text{constituency in a larger intentional complex,} \\ n_2 R n_1 &\quad \text{higher-order intentional relation.} \end{aligned}$$

6.5. Feature models and information order

A concrete finite model can represent noetic resolution by feature sets. Let

$$F_N = \{T, A, D, P, R, \ldots\}$$

where T is temporal horizon, A affective tone, D degree, P practical orientation, and R reflective stance. A resolution is a subset $r \subseteq F_N$, ordered by inclusion. Feature values themselves may also carry refinement orders, e.g.

$$\mathsf{nextJune} \succeq_N \mathsf{futureWithinYear} \succeq_N \mathsf{future}.$$

A more refined act preserves or sharpens existing information rather than contradicting it.

This suggests distinguishing truth order from information order. A coarse act may leave "urgent" undetermined rather than false. In an information order,

$$? \preceq_I \top, \qquad ? \preceq_I \bot,$$

while $\top$ and $\bot$ are incomparable. Noetic refinement is therefore naturally informational.

6.6. Intentional coherence

Let η send a noesis to its noema. Compatible noetic and noematic forgetting should satisfy the commutative square

$$\begin{array}{ccc} n' & \xrightarrow{\eta} & p' \\ \downarrow \rho_N & & \downarrow \rho_M \\ n & \xrightarrow{\eta} & p. \end{array}$$

Equivalently,

$$\boxed{\eta(\rho_N(n')) = \rho_M(\eta(n')).} \tag{IC}$$

We call (IC) the *intentional coherence law.* It ensures that coarsening the act and then asking what it intends agrees with first extracting the refined noema and then coarsening that content.

7. Do Noetic and Noematic Refinements Commute?

The preceding square concerns compatible forgetting. A deeper question asks whether the *refinement operations themselves* commute. To formulate this correctly, we represent noeses as dependent on noemata.

7.1. The intentional bundle

Let $(M, \preceq_M)$ be the preorder of noemata under refinement. For each noema $p \in M$, let

$$N_p$$

be the fiber of possible noeses directed toward p. A noematic refinement $i : p \preceq_M q$ induces a transport map

$$i_* : N_p \to N_q.$$

This says what becomes of an act of intending p when the content is refined to q.

A kind of noetic refinement is represented by a family

$$\rho_p : N_p \to N_p$$

satisfying $n \preceq_N \rho_p(n)$.

Starting with $n \in N_p$, there are two routes:

$$n \xrightarrow{\rho_p} \rho_p(n) \xrightarrow{i_*} i_*(\rho_p(n)),$$

and

$$n \xrightarrow{i_*} i_*(n) \xrightarrow{\rho_q} \rho_q(i_*(n)).$$

The commutation problem is whether these results coincide.

7.2. Naturality theorem

Theorem 7.1 (Noetic–noematic commutation)**.** *A family of noetic refinement maps $\rho = \{\rho_p\}_{p \in M}$ commutes with all noematic refinements exactly when, for every refinement $i : p \preceq_M q$,*

$$\boxed{i_* \circ \rho_p = \rho_q \circ i_*.} \tag{NAT}$$

Proof. By definition, commutation at (i, n) means

$$i_*(\rho_p(n)) = \rho_q(i_*(n)).$$

Requiring this equality for every $n \in N_p$ is exactly equality of the two composite maps $i_* \circ \rho_p$ and $\rho_q \circ i_*$. Requiring it for every noematic refinement i is precisely the naturality condition for the family ρ with respect to noematic transport. □

Thus the answer to the philosophical question has a standard mathematical form:

$$\boxed{\text{Noetic/noematic commutativity is naturality.}}$$

7.3. Orthogonality theorem

Theorem 7.2 (Independent refinements commute)**.** *Suppose all fibers are canonically the same noetic space $N_p = N$, noematic transport is the identity on noetic coordinates, and the noetic refinement is independent of p, so $\rho_p = \rho$. Then noetic and noematic refinement commute strictly.*

Proof. For every n,

$$i_*(\rho(n)) = \rho(n) = \rho(i_*(n)).$$

□

This theorem supplies an important diagnostic principle: if the act-side and content-side coordinates really factor independently, commutation is automatic. Therefore a failure of commutation witnesses a coupling between how something is intended and what is intended.

7.4. Finite noncommuting countermodel

The basic refinement axioms do not force naturality. Consider two noemata $p \prec_M q$. Let

$$N_p = \{a_0, a_1\}, \qquad a_0 \prec_N a_1,$$

and

$$N_q = \{b_0, b_1, b_2\},$$

with

$$b_0 \prec_N b_1, \qquad b_0 \prec_N b_2,$$

while b_1 and b_2 are incomparable.

Define

$$\rho_p(a_0) = a_1, \qquad \rho_q(b_0) = b_1,$$

and noematic transport

$$i_*(a_0) = b_0, \qquad i_*(a_1) = b_2.$$

Transport is monotone: $a_0 \prec a_1$ maps to $b_0 \prec b_2$. Yet

$$i_*(\rho_p(a_0)) = i_*(a_1) = b_2,$$

whereas

$$\rho_q(i_*(a_0)) = \rho_q(b_0) = b_1.$$

Hence

$$\boxed{i_*(\rho_p(a_0)) \neq \rho_q(i_*(a_0)).}$$

This is a five-noesis finite countermodel to general commutativity.

Corollary 7.3. *Noetic and noematic refinement do not commute in the minimal intentional-refinement theory.*

7.5. Philosophical interpretation: path dependence

Let p be a vague career aspiration and q a detailed representation of a particular career with concrete responsibilities. Let ρ represent movement from weak inclination to settled practical commitment. If an agent first commits to the vague aspiration and then learns what the career concretely involves, the original commitment is transported into the refined content. If instead the agent first acquires the detailed representation and only then forms a settled attitude, the resulting noesis need not be the same. Noncommutativity therefore formalizes *path dependence in desire formation.*

7.6. Degrees of commutation

Strict equality may be stronger than philosophy requires. We distinguish four levels.

I. Strict commutation:

$$i_*\rho_p(n) = \rho_q i_*(n).$$

II. Extensional commutation: the two states mutually refine one another and are informationally equivalent.

III. Weak compatibility: the two outcomes differ but possess a common admissible refinement z.

IV. Genuine obstruction: no common admissible refinement exists.

The last case is especially significant: order of intentional development produces incompatible endpoints.

7.7. Stable, path-dependent, reconcilable, and transformative prolepsis

Proleptic wanting often enriches both act-side temporal orientation and content-side representation of a future self or state. It is therefore naturally a diagonal refinement. The commutation analysis yields a taxonomy:

$$\begin{aligned}
\textbf{stable prolepsis}: &\quad R_N R_M \cong R_M R_N,\\
\textbf{path-dependent prolepsis}: &\quad R_N R_M \ncong R_M R_N,\\
\textbf{reconcilable prolepsis}: &\quad \text{the two paths have a common refinement},\\
\textbf{transformative prolepsis}: &\quad \text{the two paths have no common admissible refinement.}
\end{aligned}$$

This gives a candidate formal definition of a class of transformative desires: increasing determinacy in how one wants and in what one wants cannot be performed in either order without changing the resulting intentional state.

8. What the Wanting Case Teaches PhilLean

The case study reveals several general design principles.

First, conceptual inclusion should not be guessed from ordinary grammar. "Endorsed wanting" is a subtype of wanting under the chosen definition, but "wanting to want" is a higher-order attitude and need not imply the first-order desire. The proof checker makes this distinction precise.

Second, the system should expose hidden bridge principles. An early representation of proleptic wanting required a Noematic Coarsening Principle to infer an ordinary want from a refined one. Once prolepsis was decomposed into noetic and noematic refinement, some cases became definitional projections while others remained substantively dependent on transport laws. Formalization thus improves the theory rather than merely encoding it.

Third, thought experiments can often be represented as countermodels. The agent who wants to want p without wanting p is a model refuting one conceptual implication. This formalizes a widespread philosophical practice: a counterexample is a structure satisfying the proposed premises or analysis while falsifying the proposed consequence.

Fourth, conceptual taxonomy can be generative rather than enumerative. Species of wanting can be classified by a refinement signature indicating noetic refinement, noematic refinement, structural extension, and higher-order intentional relation. This is more explanatory than introducing an unrelated predicate for each kind of desire.

9. Relation to the Husserlian Monad Program

The present framework was developed against the background of earlier experimental manuscripts on interacting Husserlian monads and finite overminds [10, 11]. Those studies treated finite intentional systems as formal structures whose interactions can generate higher-order or collective forms of mentality. The present paper contributes a complementary "microscopic" layer.

A Husserlian monad can now be enriched internally with an intentional bundle

$$\mathcal{I} = (M, \{N_p\}_{p \in M}, \preceq_M, \preceq_N, (-)_*)$$

whose base M contains noematic states and whose fibers N_p contain noeses directed toward p. Higher-order intentional acts create links between fibers and between represented acts. Endorsement creates structured extensions; proleptic development induces refinement paths. Thus a monad is no longer merely a state-transition system with intentional labels: it can carry a structured geometry of intentional refinement.

This refinement geometry may matter to future work on interacting monads. If two monads exchange only coarse noematic content, they may fail to preserve the noetic history through which that content acquired motivational force. Conversely, a collective or "overmind" might be defined not merely by pooling intentional contents but by integrating refinement structures while satisfying coherence conditions. Noncommuting refinements also create a possible source of diachronic identity and path dependence: two monads with the same coarse present intentional state may have different refinement histories and therefore different responses to subsequent refinements.

These suggestions are programmatic, but they show why the present theory is not separate from the monadological program. It supplies a local calculus of intentional structure that can be installed inside the finite models previously studied.

10. Toward an Implementation in Lean

A practical PhilLean implementation should proceed incrementally.

10.1. Phase I: trusted core

Implement the abstract interfaces for typed syntax, semantic systems, definitions, readings, argument schemas, object logics, interpretations, models, and countermodels. Prove host-logic insulation properties, substitution coherence, definitional normalization, and the generic countermodel-refutation theorem.

10.2. Phase II: a finite intentional logic

Implement a small finite logic sufficient for the wanting case study. Define agents, noemata, noeses, intentional modes, endorsement, higher-order representation, refinement preorders, and transport. Mechanize the separation theorems and finite countermodels.

10.3. Phase III: model generation

Connect finite fragments to an automated model finder or generate exhaustive finite models inside Lean for small cardinalities. A failed derivation should, where decidable, trigger search for a countermodel and report the smallest one found.

10.4. Phase IV: multiple logics

Introduce classical propositional and first-order modules, modal logics, epistemic logics, and selected nonclassical logics. Each must expose the same generic interface without being identified with Lean's own logic. This enables logic-robustness reports.

10.5. Phase V: semantic and dialectical interface

Build a human-facing representation in which ordinary philosophical sentences are associated with multiple candidate readings and formalizations. Explications can be represented separately from definitions. The interface should produce argument certificates containing at least:

- target logic and interpretation;
- essential premises;
- definitions and substantive bridge principles;
- proof status and, where applicable, proof object;
- consistency status;
- countermodels;
- alternative readings or logics under which the result changes.

10.6. A minimal Lean-style intentional bundle

The commutation theory can be represented by a small interface.

```
structure IntentionalBundle where
  Noema : Type
  mRefines : Noema -> Noema -> Prop

  Noesis : Noema -> Type
  nRefines : forall p, Noesis p -> Noesis p -> Prop

  transport : forall {p q},
    mRefines p q -> Noesis p -> Noesis q

  transport_mono : forall {p q} (h : mRefines p q) {n1 n2},
    nRefines p n1 n2 ->
    nRefines q (transport h n1) (transport h n2)

structure NoeticRefinement (I : IntentionalBundle) where
  refine : forall p, I.Noesis p -> I.Noesis p
  inflationary : forall p n, I.nRefines p n (refine p n)

def Commutes (I : IntentionalBundle)
    (rho : NoeticRefinement I) : Prop :=
  forall {p q} (h : I.mRefines p q) (n : I.Noesis p),
    I.transport h (rho.refine p n)
```

```
    = rho.refine q (I.transport h n)
```

This code sketch is intentionally conservative. A production implementation would require explicit universe levels, preorder laws, equivalence quotients, and careful treatment of equality versus isomorphism.

11. Philosophical Significance and Limits

11.1. Formal validity is conditional

PhilLean cannot decide whether a controversial philosophical premise is true merely because it has been formalized. Its proper output is conditional: given interpretation I, logic L, and premises Γ, conclusion C follows or fails to follow. The system should make this conditionality more visible, not less.

11.2. Formalization is itself philosophically contestable

A formalization may misrepresent the target concept. This is why readings, explications, and interpretations are first-class objects. The system should support interpretive robustness rather than assuming a unique translation from prose to formula.

11.3. Not all objections are countermodels

Some objections concern relevance, explanatory adequacy, phenomenological fidelity, or dialectical burden rather than deductive validity. PhilLean should represent these as typed dialectical relations without pretending they reduce to model theory.

11.4. Automation can discover philosophy-relevant facts

The precedent set by computational metaphysics is important. Automated formalization has already found errors and paradoxes in sophisticated metaphysical theories [6, 7]. The present wanting theory illustrates a smaller version of the same phenomenon: formalization exposed the need to distinguish refinement from extension and revealed that the relation between proleptic and ordinary wanting depends on whether prolepsis lies on the noetic side, noematic side, or both.

11.5. A new notion of philosophical discovery

Once many theories are formalized, automated tools can search for minimal inconsistent subsets, unexpected consequence relations, independence results, hidden equivalences, and small countermodels. A result of the form

$$A + B + D + F \vdash \neg X$$

may expose a previously unnoticed incompatibility among commitments independently endorsed by a philosophical position. Such results are not replacements for philosophy; they are new objects for philosophical interpretation.

12. Conclusion

The proposal for "Lean for Philosophy" becomes viable once the target is formulated correctly. The goal is not a machine that labels philosophical prose true or false. It is a framework in which the inferential, semantic, interpretive, and dialectical structure of arguments is explicit and mechanically auditable.

The core architecture separates semantic systems from object logics, explications from stipulative definitions, argument schemas from their formalizations, and philosophical formulas from the proof assistant's native propositions. This allows Lean's kernel to verify proofs without silently forcing every philosophical theory into Lean's own inferential regime.

The wanting case study demonstrates the value of this architecture. Endorsed wanting, wanting-to-want, and proleptic wanting occupy distinct formal positions. Noetic refinement and noematic refinement provide two independent axes of intentional determination, while endorsement and higher-order wanting require additional structural relations rather than being forced into the refinement order. Treating noeses as a dependent family over noemata yields the central mathematical result of the paper: noetic and noematic refinement commute exactly when noetic refinement is natural with respect to noematic transport. Independence implies commutation, but the basic axioms admit a finite noncommuting countermodel. Noncommutation therefore becomes a precise measure of coupling and path dependence in intentional development.

This yields a particularly suggestive philosophical application. A proleptic desire can be stable, merely path-dependent, reconcilable after further refinement, or transformative in the strong sense that alternative orders of noetic and noematic development lead to incompatible intentional endpoints. The formal theory thus moves beyond cataloguing kinds of wanting and begins to describe a geometry of intentional transformation.

The immediate next step is implementation. Even a small Lean library containing the core interfaces, the finite wanting theory, the five-noesis noncommutation countermodel, and the generic naturality theorem would constitute a concrete proof of concept for PhilLean. From there, the same infrastructure could be used to formalize larger philosophical debates while preserving the central methodological principle: the machine certifies exactly what follows from explicitly represented commitments, while the philosophy remains in the choice, interpretation, and evaluation of those commitments.

A. Summary of Core Judgments

For reference, the principal judgments proposed in this paper are:

Judgment	**Reading**
$\Gamma \vdash_L \varphi$	φ is derivable from Γ in object logic L.
$M \models_L \varphi$	Model M satisfies φ in L.
$\mathsf{Valid}_{L,I}(A)$	Argument A has a proof under interpretation I in L.
$\mathsf{Consistent}_{L,I}(A)$	The formalized premises of A have a common model.
$T \triangleright p$	p is an explicit commitment encoded by theory T.
$T \Vdash_{I,L} p$	p follows from theory T under (I, L).
$n' \succeq_N n$	n' noetically refines n.
$p' \succeq_M p$	p' noematically refines p.
$n \triangleleft X$	n is a constituent act of intentional complex X.

$i_*\rho_p = \rho_q i_*$	noetic refinement commutes with noematic transport along $i : p \preceq_M q$.

B. First Theorem Table for Wanting

Claim	Status in minimal theory
$\mathsf{EWant}(a, p) \to \mathsf{Want}(a, p)$	Theorem (definitional).
$\mathsf{Want}(a, p) \to \mathsf{EWant}(a, p)$	Refuted by finite countermodel.
$\mathsf{WWant}(a, p) \to \mathsf{Want}(a, p)$	Refuted by finite countermodel.
$\mathsf{Want}(a, p) \to \mathsf{WWant}(a, p)$	Refuted by finite countermodel.
$\mathsf{EWant}(a, p) \to \mathsf{WWant}(a, p)$	Refuted by finite countermodel.
$\mathsf{WWant}(a, p) \to \mathsf{EWant}(a, p)$	Refuted by finite countermodel.
Pure noetic proleptic refinement $\to$ ordinary wanting	Theorem if mode is preserved under pure noetic refinement.
Arbitrary refined proleptic wanting $\to$ coarse wanting	Requires an appropriate transport/projection principle.
Noetic refinement commutes with noematic refinement	Not derivable in general; equivalent to naturality for the chosen refinement family.

C. The Five-Noesis Countermodel in Tabular Form

Component	Definition	Role
Noemata	$p \prec_M q$	one noematic refinement
Fiber over p	$N_p = \{a_0, a_1\}$, $a_0 \prec_N a_1$	coarse noetic refinement
Fiber over q	$N_q = \{b_0, b_1, b_2\}$	refined noetic possibilities
Order over q	$b_0 \prec b_1$, $b_0 \prec b_2$	b_1, b_2 incomparable
Noetic refinement	$\rho_p(a_0) = a_1$, $\rho_q(b_0) = b_1$	chosen refinement operation
Transport	$i_*(a_0) = b_0$, $i_*(a_1) = b_2$	monotone noematic transport
Path 1	$i_*\rho_p(a_0) = b_2$	noetic then noematic
Path 2	$\rho_q i_*(a_0) = b_1$	noematic then noetic
Conclusion	$b_2 \neq b_1$	noncommutation

References

[1] L. de Moura, S. Kong, J. Avigad, F. van Doorn, and J. von Raumer, "The Lean Theorem Prover (System Description)," in *Automated Deduction – CADE-25*, LNCS 9195, Springer, 2015, pp. 378–388. DOI: 10.1007/978-3-319-21401-6_26.

[2] Lean developers, *The Lean Language Reference*, current online reference manual, https://lean-lang.org/doc/reference/latest/ (accessed August 26, 2026).

[3] A. Wierzbicka, "Semantic Primitives, fifty years later," 2021. ANU Open Research repository record: https://openresearch-repository.anu.edu.au/items/f530a50e-1097-4b36-aea5-89cf11a5fbd3.

[4] Natural Semantic Metalanguage research program, “Chart of NSM Semantic Primes,” version 19, April 12, 2017. The chart lists the mature inventory of semantic primes and combinatorial frames, including WANT.

[5] B. Fitelson and E. N. Zalta, “Steps Toward a Computational Metaphysics,” *Journal of Philosophical Logic* 36(2) (2007), 227–247.

[6] E. N. Zalta and collaborators, “The Computational Theory of Plato’s Forms,” Computational Metaphysics project, Stanford University, https://mally.stanford.edu/cm/forms/ (accessed August 26, 2026).

[7] D. Kirchner, C. Benzmuller, and E. N. Zalta, “Mechanizing Principia Logico-Metaphysica in Functional Type Theory,” *Review of Symbolic Logic* 13(1) (2020), 206–218. DOI: 10.1017/S1755020319000297.

[8] C. Beyer, “Edmund Husserl,” *Stanford Encyclopedia of Philosophy*, Summer 2026 edition. Discussion of the noesis/noema correlation and competing interpretations of noema.

[9] “Intentionality,” *Stanford Encyclopedia of Philosophy*, Spring 2023 archive, sections on Husserl, noema, and intentional directedness.

[10] *Possible Worlds of Interacting Husserlian Monads*. Unpublished companion working manuscript, 2026. Cited by title; full bibliographic metadata was not available during preparation of the present manuscript.

[11] *Minimal Finite Models of Diachronic and Self-Conscious Overminds*. Unpublished companion technical note, 2026. Cited by title; full bibliographic metadata was not available during preparation of the present manuscript.