Preprint · Version 1.0 · 28 October 2026 doi:10.5281/zenodo.23734424 · CC BY 4.0

The Rising-Sea Theses

Twelve Theses on Mathematics and Machine Reasoning

Vincent Liu
WiseMotif.ai · vincent.liu@wisemotif.ai · https://orcid.org/0009-0000-8718-222X

Abstract. The Grothendieckian Programme is a set of seven works on machine reasoning about mathematics (34 volumes, 26,021 pages), released openly in October 2026 as first drafts that have not been peer-reviewed. This paper draws from them twelve theses on how mathematics and machine reasoning should be combined. Each thesis is stated in one sentence, argued, tied to the passages of the seven works on which it rests, related to existing work, and paired with an observation that would refute it. The theses concern the unit of mathematical knowledge for a reasoning system, the place of epistemic status, the latent space of mathematical reasoning, attention, failure, authority, and the method of discovery. The paper adds no mathematical result to the seven works; none of them claims a proof of the Riemann Hypothesis, the Birch and Swinnerton-Dyer conjecture, the Hodge conjecture, the abc conjecture or the Langlands correspondence.

2020 Mathematics Subject Classification. Primary 00A35; Secondary 68T01, 68V15, 18F10.
Key words and phrases. Grothendieckian Programme, machine reasoning, epistemic status, descent, formal verification, recursive mathematical discovery.

hexagon:2611.00259v1

1. Introduction

A system that reasons about mathematics has to settle three questions before it settles any mathematical one: what it stores, what it may assert, and who may turn an assertion into a result. The answers are usually left implicit. A language model stores text and parameters, asserts whatever it generates, and leaves the third question to whoever reads the output; a proof assistant stores declarations, asserts what its kernel has checked, and leaves open whether the declarations say what was meant. Neither answer is adequate for a system that is to propose new mathematics at a scale that exceeds what readers can check.

This paper states twelve theses on how mathematics and machine reasoning should be combined. They are drawn from the seven works of the Grothendieckian Programme [S01; S02; S03; S04; S05; S06; S07], released openly in October 2026 with an overview [S00]. Two of the works design a reasoning system for mathematics; five apply one method of discovery to five named problems and record what it establishes, what it reduces to stated hypotheses and what it leaves open. The theses are the principles on which the seven works rest, stated so that they can be argued with independently of the works.

Each thesis is given in five parts: the thesis in one sentence; an argument for it; the passages of the seven works on which it rests, quoted from the published PDFs with their pages; its relation to existing work; and an observation that would refute it. The last part is not a formality. Several theses are claims about what will work, and they can be tested. This is the second of three papers released together. The first, *The Grothendieckian Programme: An Overview* [Liu26a], describes the architecture of the Programme, its five research subjects, its method and the series; Section 2 below summarises what the theses need from it. The third, *The 57 Grothendieckian Problems* [Liu26c], states fifty-seven open problems drawn from the same works, and each thesis lists the problems at which it can be tested.

The theses fall into three groups. Theses 1–3 concern representation: what a reasoning system should store (worlds, not isolated theorems), how the status of a statement should be recorded (as part of its type), and on what it should reason (a structured, non-vectorial carrier). Theses 4–7 concern operation: attention rebuilt as descent, the reporting of failure and abstention, the separation of operational completion from validity, promotion and publication, and the separation of proposing from authorising. Theses 8–12 concern method: raising the theory around a hard problem, stating what a result does not establish, recursive discovery, the creation of language, and the judgment of a method by an open ledger.

Two limits apply throughout. First, the paper adds no mathematical result to the seven works, and none of the seven claims a proof of the Riemann Hypothesis, the Birch and Swinnerton-Dyer conjecture, the Hodge conjecture, the *abc* conjecture or the Langlands correspondence. Where a thesis is illustrated by one of the five research subjects, the illustration is the record of a method, not a claim about the conjecture. Second, the seven works are first drafts that have not been peer-reviewed, and much of the architecture they describe is specified but not implemented. A thesis that rests on a specification is a thesis about a design; Section 6 says which theses are affected and how.

Section 2 describes the seven works and states what they contain and what they do not. Section 3 relates the theses to existing work. Section 4 states the theses, and Section 5 shows how they depend on one another. Section 6 discusses the limits of the evidence and the ways in which the theses could fail, and Section 7 concludes.

2. The source works

The seven works were released on Zenodo between 20 and 25 October 2026 and the overview on 26 October 2026, each with a DOI, under the Creative Commons BY-NC-ND 4.0 licence, as Research Architecture Edition 1 of the WiseMotif Monograph of Mathematical Intelligence. Together the seven comprise 34 volumes, 26,021 pages and 995 chapters. Counted by two independent methods on their sources and on the sealed PDFs, and taking the smaller count where the two differ, they contain more than 349,000 formulas, more than 10,500 numbered statements (theorems, propositions, lemmas, corollaries, definitions and conjectures), 714 numbered algorithms and 1,434 numbered figures and plates [S00]. Each work is published with a signed list of checksums of its files.



FIGURE 1. The seven works of the Grothendieckian Programme, as the seven cells of [S00, Figure 1]. At the centre the core [S01]; above it, on a shared edge, the meta-reasoning kernel [S02]; around them the five works on research subjects [S03; S04; S05; S06; S07]. Each cell gives the formula printed on the centre cell of the work's cover and the number of pages of its sealed PDF. The short arrows across the shared edges are the three kinds of interface of [S00, Figure 1]; a port marks an interface, not a proof. The decomposition of the core drawn there is not repeated.

Figure 1 draws the seven works as the seven cells of one honeycomb, as the overview does [S00, Figure 1]: the two works on the architecture at the centre and above it, the five works on research subjects around them.

2.1. The architecture

GAI-MUA: The Grothendieckian Machine [S01], in six volumes, sets out how mathematical worlds can be held by a machine as typed, versioned objects and compared and transported between settings. Its first two volumes are mathematical: schemes, topoi, descent, cohomology, motives, higher structures and their modern successors, with sources and hypotheses recorded. The third specifies an architecture in two systems. The mathematical-universe system GMUS contains the Motivic Unification Engine (MUE), a structured

carrier for geometric, cohomological and realisation data; the Topos Reasoning Engine (TRE), whose two engines propose new worlds and reason in admitted ones; and a source-audited research lane (GRA). The language system GMLS contains a language forge for notation and theory, a contextual cognitive layer, a descent workflow (GDRA) executed by capability-limited agents, and an experimental calculus (AURORA). The remaining volumes specify algorithms, services and evaluation. In the words of its Series Note, the name "names a research architecture, not a completed universal solver." [S01, p. iv]

GM-Omega-Infinity: Topos-Native Intelligence [S02], in five volumes, states what a reasoning kernel must be if it is to reason inside such worlds. A context is presented as a site, what it supports as a sheaf, and a conclusion as a global section; the kernel covers, assigns, transports, glues, obstructs, decides by Kripke–Joyal forcing, and changes base. The work proves a number of results on which its design rests — among them a dichotomy for the holonomy of orthogonal cellular sheaves and an identity expressing the group-relative advantage of GRPO through a graph Laplacian [S02, §1.2.4] — and records its own position plainly: "the mathematics is written and partly proved, one finite instance has been computed, and nothing has been trained or measured." [S02, §1.2.6, p. 23]

2.2. The five research subjects

Each of the other five works takes one named problem as its research subject and applies to it the method that the Series Notes call recursive mathematical discovery. *Arithmetic Spectral Descent* [S03] takes the Riemann Hypothesis; its recursion "ends in a proved conditional chain, twenty-three scoped obstructions, a certified computation stack, and two obligations stated precisely enough to be attacked, rejected or falsified" [S03, p. iv]. *Fractured Adelic Geometry* [S04] takes the conjecture of Birch and Swinnerton-Dyer and builds a framework from twelve axioms whose consistency it does not prove; its table of output by status records zero cases of the conjecture established [S04, Table 56.1, p. 437]. *The Archimedean Prism* [S05] takes the Hodge conjecture; it records that the conjecture remains open, that the book does not prove it, "and proves no case of it that was not already known." [S05, p. v] *Strained Prismatic Geometry* [S06] takes the *abc* conjecture and publishes a census of its 1,420 statements by status [S06, p. xvii]. *The Arithmetic Topos* [S07] takes the Langlands programme and ends in a ledger of one hundred and eleven items, fifty-three of them conjectural [S07, p. v]. None of the five claims a proof of its named problem.

2.3. The method in the five works

The five works share a method and a discipline of reporting, set out in a common Series Note and applied to each subject in its own way. A proposed structure is tested first against cases in which the answer is known. In [S03] the test is a firewall of four published counter-models — Davenport–Heilbronn functions, Beurling generalised number systems, Selberg zeta functions with exceptional eigenvalues, Epstein zeta functions of class number greater than one — and the rule is strict: "A mechanism is not admitted as a route to RH until the book explains why it fails, or changes form, on all four objects." [S03, p. ix] In [S07] the main conjecture is tested against the function-field case, the de Rham limit, the local fields, the abelian case and the elliptic curve of conductor eleven

[S07, p. v]. What fails a test is kept: as a no-go theorem with explicit hypotheses in [S03], as one of fourteen entries of an obstruction atlas and twenty-two critical experiments with kill criteria in [S06, p. v], and in [S05] as a falsifier. [S05] closes with "eleven falsifiers, a self-assessment against seven criteria fixed in advance — four met, two partially, one not met: a proof of the Hodge Conjecture — and sixty problems." [S05, p. vii]

Each work also states its claim boundary in its own words and prints a status with each statement. The notations differ — seven codes in [S01], grades of licence in [S02], six labels in [S06], five status marks in [S07] — but the rule is the same: no statement is printed at a strength greater than its support. The theses below take this discipline as their subject.

2.4. What the works do not contain

The works do not contain a running implementation of the architecture: the third volume of [S01] is a specification, and [S02] reports no training run. Lean verification of the seven works is incomplete. They have not been reviewed by others, and the author's declaration of the use of AI tools in their preparation is reproduced at the end of this paper. The theses below therefore rest on the works as published texts — definitions, proved statements, specifications and recorded failures — and not on any empirical result about a built system.

3. Relation to existing work

The theses touch four bodies of work.

Machine-guided mathematical discovery. Machine learning has been used to suggest conjectures that mathematicians then proved [DVB+21]; program search guided by language models, with an external evaluator, has produced new constructions in combinatorics [RBN+24]; a neuro-symbolic system has solved olympiad geometry problems [TWL+24]; and automated conjecturing has a longer history, from Graffiti [Faj88] to systems that propose formulas for fundamental constants [RGM+21]. In these systems the evaluator — a proof, a verified construction or a numerical test — stands outside the generator. Theses 6, 7 and 10 make the same separation a requirement of the architecture, and extend it to the status of every intermediate artefact and to the record of failures (Thesis 10).

Formal verification. The de Bruijn criterion — that proofs found by any means be checked by a small trusted kernel — underlies the proof assistants now used in research mathematics [BW05; MU21], and large libraries such as Mathlib have made it possible to state much research mathematics formally [mC20]. The Liquid Tensor Experiment showed that recent research-level results can be formalised [Sch22]. Public collections of formally stated open problems are used as benchmarks [FLM+26], and autoformalization has made the fidelity of a formal statement to its informal source a measured quantity [WJL+22]. Avigad describes the turn of mathematics toward formal methods [Avi24]. Theses 2 and 6 take from this tradition the separation of checking from search, and add two things it does not by itself provide: a status for artefacts that are not proofs, and a separate obligation of fidelity between a formal target and the mathematics it is meant to express.

Structured representations in machine learning. Geometric deep learning builds the symmetries of a domain into a network [BBCV21]; categorical deep learning proposes algebraic theories of architectures [GLD+24]; cellular sheaves and sheaf neural networks attach structured stalks and restriction maps to graphs [HG19; HG20; BDCLB22]; and topos-theoretic designs for language models have been proposed [Mah25]. Results on the circuit complexity of transformers bound what a fixed-depth attention network can compute [MS23]. Theses 3 and 4 belong to this line. They differ from it in asking that gluing and obstruction be outputs of the kernel rather than regularisers, and in grading every architectural claim by what has been shown.

The practice of mathematics. Jaffe and Quinn proposed that speculative mathematics be labelled as such [JQ93], and the replies, among them Thurston's account of proof as understanding communicated between people [Thu94], concerned what a label must carry. Lakatos described how concepts change under proofs and refutations [Lak76]; Grothendieck's image of the rising sea is studied by McLarty [McL07]. In October 2026 a repository for research works in the mathematical sciences was announced that accepts works generated entirely by language models and records, for each, the extent of machine generation and of human understanding [Ant26]. Theses 2, 7, 9 and 12 are proposals within this discussion.

4. The twelve theses

Each thesis is stated in italics and followed by its argument, the passages of the seven works on which it rests, its relation to existing work, an observation that would refute it, and the problems of *The 57 Grothendieckian Problems* at which it can be tested. Quotation marks enclose the words of the source on the page cited; the page numbers are the printed folios of the published PDFs. Figure 2 shows the twelve theses in their three groups, and Figure 3 draws Theses 6 and 7.

Representation

Thesis 1 (*Worlds, not isolated theorems*).

> *For a reasoning system, the unit of mathematical knowledge is a typed, verifiable world, not an isolated theorem.*

ARGUMENT. A theorem holds relative to definitions, hypotheses, an ambient category, a base and a system of coefficients. A mathematician carries this context implicitly and restores it when a result is used; a system that stores statements as text or as vectors loses it, and the errors that follow are errors of transport — a result applied after a change of base, of coefficients or of variance that it does not survive. The relative point of view of Grothendieck's school makes the context part of the object: one works with objects over a base and asks which statements are stable under base change [SGA4]. The thesis transfers that practice to machine reasoning. The stored object is a world — its logic, objects, maps, variance, base, coefficients and bounds, each explicit — and a result is admitted only together with the world in which it holds and the evidence that licenses each transport out of it. Retrieval, comparison and reuse then become

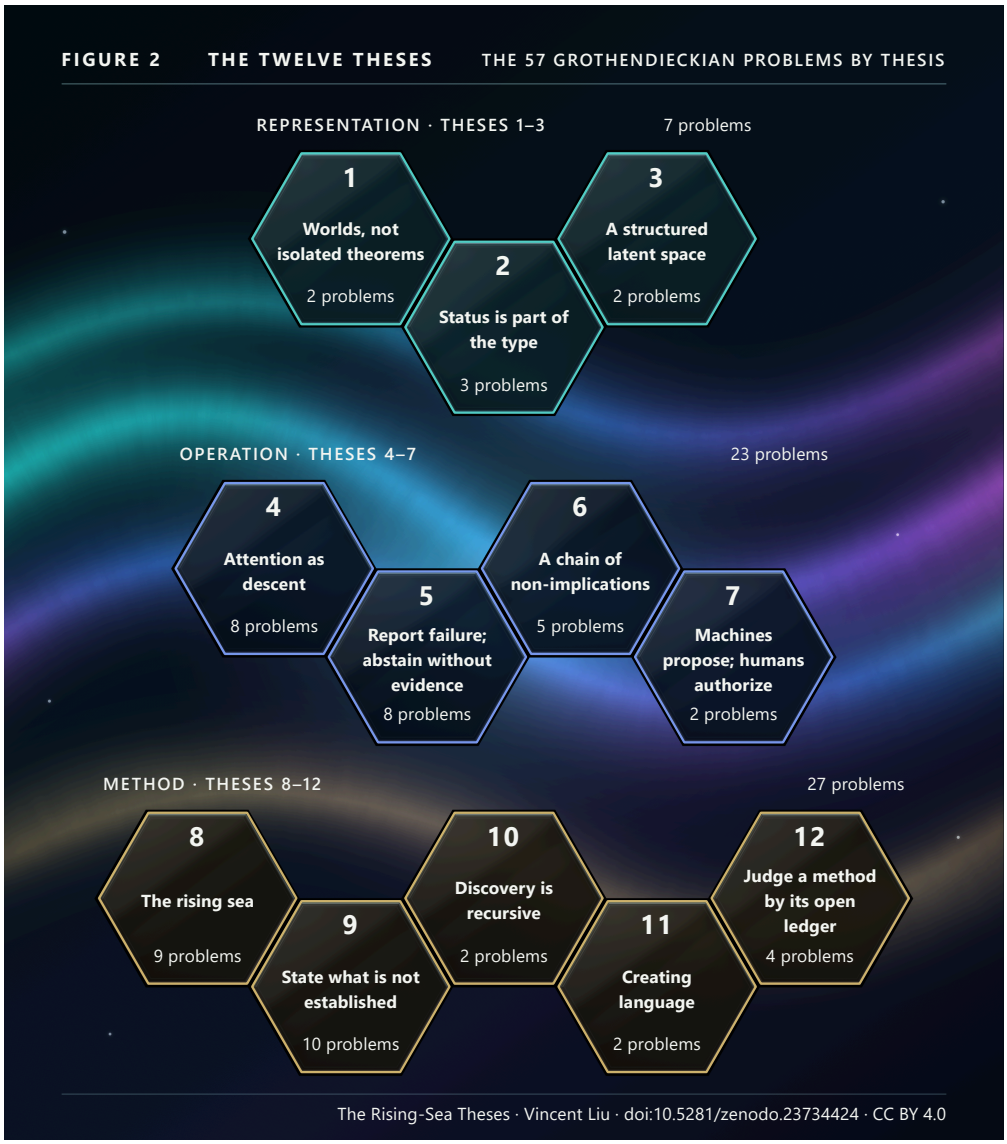


FIGURE 2. The twelve theses in their three groups — representation (Theses 1–3), operation (Theses 4–7) and method (Theses 8–12) — with the number of problems of *The 57 Grothendieckian Problems* at which each can be tested; the problems are listed with each thesis. Each cell links to its thesis.

operations on worlds, with a record of what each preserves, loses or leaves unknown.

EVIDENCE IN THE SEVEN WORKS. The Series Note of [S01] states the architectural obligation as "to express mathematical worlds as typed, versioned, provenance-bearing artifacts that can be searched, compared, formalized, computed with, and evolved without allowing statistical confidence to masquerade as proof." [S01, p. iv]

The contract of the mathematical-universe system: "A result is valid only relative to the world in which it was formed and to the preservation evidence attached to any transport." It "therefore works with a family of worlds and maps rather than with a single undifferentiated mathematical memory." [S01, Vol. III, Ch. 14, p. 8218]

AN INSTANCE. Two local sections report a flux across the boundary of an overlap, one in watts per square metre with outward orientation, the other in milliwatts per square centimetre with inward orientation. “A shape-only checker compares the numerals and reports conflict. A permissive checker takes absolute values and reports agreement. Both are wrong because neither binds the transformation.” [S01, Vol. III, §61.19, p. 9227] The typed comparison converts units by an exact scale map, applies the orientation through a declared sign representation, checks variance, and only then compares; a missing orientation witness makes the comparison ill-typed rather than compatible or incompatible. Each of these steps is a fact about the world in which each section was formed, and none is visible in the numbers.

RELATION TO EXISTING WORK. Mathematical knowledge management has long separated statements from their context of definitions and theories [CF09]; proof assistants enforce context through types and type classes, as in Mathlib [mC20]. What the thesis adds is that transports between contexts are first-class records, with their preservation and loss stated, rather than coercions resolved silently by the elaborator.

WHAT WOULD REFUTE IT. A system that stores statements without world records and makes no more transport errors — after base change, change of coefficients or change of variance — than one that stores worlds, on a corpus of the kind specified for Problem 15.

PROBLEMS. In *The 57 Grothendieckian Problems*: Problems 4, 15.

Thesis 2 (*Status is part of the type*).

The epistemic status of a statement belongs to its type.

ARGUMENT. A confidence score places statements on one scale; a status records what kind of support a statement has. The two are different kinds of object. A checked theorem, a definition proposed by the project, a research construction that may fail, a conjecture, an unmet obligation and a disputed claim do not lie on a line: they license different uses, and the composition of two of them has a status determined by rules, not by averaging. A system that collapses status into a number loses the distinction at exactly the point where a downstream user needs it, and allows, for example, a disputed premise to support an established conclusion. Making status part of the type puts the rules of composition where a checker can enforce them. The seven works apply the discipline to themselves: every statement is printed with its status, and each work publishes the counts.

EVIDENCE IN THE SEVEN WORKS. “The series-wide codes E/S/D/R/C/O/Q are not confidence labels.” They are established (E), source-grounded synthesis (S), project-native definition (D), falsifiable research construction (R), conjecture (C), obligation (O) and disputed (Q), and “Validation state and epistemic status are orthogonal”. [S01, Vol. III, §1.3, p. 7974]

The census of [S06]: “[NEW] 646 first proved here, unconditionally [CONDITIONAL] 177 proved from named obligations [OPEN] 48 not proved here [DISPUTED] 4 disagreement recorded, not adjudicated”, together with 223 statements established elsewhere and 322 known statements reproved. [S06, p. xvii]

An instance. The two kinds of record a status type keeps apart are spelled out in [S01]: "A kernel-replayed lemma about a project-native structure does not change the structure's definition status from D. A disputed source record remains Q even when it has been parsed flawlessly." [S01, Vol. III, §1.3, p. 7974] A machine check raises the confidence that a lemma is proved; it does not turn a definition chosen by the project into established mathematics, and a parser that reads a contested source without error does not settle the contest.

Relation to existing work. The proposal that speculative mathematics be labelled as such goes back at least to Jaffe and Quinn [JQ93], and the debate it provoked [Thu94] concerned exactly what a label must record. Proof assistants distinguish declarations that are proved from those that depend on an admitted step (in Lean, `sorry`) or on added axioms, and report the difference; the thesis extends that distinction from formal proofs to every artefact a reasoning system produces, including conjectures, obligations and disputed sources.

What would refute it. A scalar ordering of statements that preserves, under the composition rules of [S01, Vol. III, §56.15], every distinction drawn by the seven codes; or a controlled study in which a calibrated confidence score supports the same downstream decisions with no more errors.

Problems. In *The 57 Grothendieckian Problems*: Problems 7, 8, 31.

Thesis 3 (*A structured latent space*).

Mathematical reasoning needs a structured, non-vectorial latent space.

Argument. An embedding in a vector space supports similarity and interpolation. It does not carry maps between objects, the variance of a construction, the identity strength of a comparison, or the place where a comparison fails; and a nearest neighbour is not a morphism. The thesis is that the carrier on which a system reasons about mathematics — as distinct from the index it searches — must be a structure of that kind: typed objects, maps, finite diagrams, gradings and filtrations, comparison faces and recorded residuals. In [S01] this carrier is the Motivic Unification Engine, whose objects are finite and inspectable proxies (Nori-style diagrams, factorisation records, realisation prisms), each labelled as a proxy, with claims about motives reserved for the evidence appropriate to them. The thesis does not say that vector representations have no use; retrieval and ranking may use them. It says that a symbol used in reasoning must denote a typed referent.

Evidence in the seven works. "The Mathematical Universe Engine provides structured, non-vectorial carriers for geometric, cohomological, representation-theoretic, and realization data." "Their value lies precisely in being inspectable and bounded, with general motivic claims reserved for the mathematical evidence appropriate to that stronger scope." [S01, p. 7955]

"MUE is not exposed to GMLS as an opaque vector store." "A symbol cannot denote "the latent meaning" without a typed referent." [S01, Vol. III, §15.4, pp. 8250–8251]

An instance. A finite computation may produce an object that behaves like a motive. In [S01] its name must say that it is a proxy: "A primitive named Motive(X) is therefore inadmissible if its denotation is only a selected finite

Nori diagram." [S01, Vol. III, §15.4, p. 8252] A notation that records the bound, the diagram, the degree and the coefficient profile is admissible, because the reader can recover what was computed. A structured carrier makes this rule enforceable; in an embedding the difference between a motive and a finite proxy of one has no representation.

RELATION TO EXISTING WORK. Geometric deep learning builds symmetry into representations [BBCV21]; categorical deep learning proposes algebraic theories of architectures [GLD+24]; sheaf neural networks attach structured stalks and restriction maps to graphs [HG20; BDCLB22]. These keep a vector space at each point. The thesis asks for more: that the objects reasoned about be the mathematical structures themselves, as finite proxies with their maps, in the sense of Nori's diagrams [HMS17].

WHAT WOULD REFUTE IT. A vector representation that performs as well as a typed carrier on tasks requiring transport and failure reporting — the counterexample and proof strata of the benchmark of Problem 13 — or a proof that the typed structure is recoverable from the embedding at no loss.

PROBLEMS. In *The 57 Grothendieckian Problems*: Problems 6, 18.

Operation

Thesis 4 (*Attention as descent*).

Attention can be rebuilt as descent.

ARGUMENT. In a transformer, attention forms weighted averages of value vectors, the weights given by similarity [VSP+17]; nothing in the operation requires the averaged values to be expressed in a common frame. Descent offers a different discipline: before local data are combined, they are carried along restriction maps into the stalk of the overlap where they can be compared, and the combination either glues to a global section or returns the obstruction. [S02] builds a reasoning kernel in which these steps are operations — cover, assign, transport, glue, obstruct, decide, change base — and in which a mixture of experts becomes a cover by open sub-topoi. The thesis is a design thesis: such a kernel can be built, and its components are mathematics before they are architecture. Its scope is fixed by what has been proved and what has not. The source corrects its own first formulation — attention is not a coend without further data — and records that nothing has yet been trained or measured.

EVIDENCE IN THE SEVEN WORKS. "Attention is rebuilt as descent: values are carried along restriction maps into the shared stalk of an overlap and compared there, so that what is mixed has first been made comparable." [S02, Preface, p. viii]

The correction: "Thus attention remains a barycentric numerical kernel unless a separate functor and universal property are supplied." [S02, §1.2.4, p. 21]

The present position: "the mathematics is written and partly proved, one finite instance has been computed, and nothing has been trained or measured." [S02, §1.2.6, p. 23]

AN INSTANCE. Sheaf neural networks learn restriction maps together with the other parameters, and "they were observed, again and again, to learn the identity sheaf and become ordinary graph networks." [S02, Preface, p. xiv] [S02] traces the collapse to a loss whose minimiser does not depend on the data, and of the

remedies it proves, it would build on heterogeneous stalks, "because it does not only disincentivise the trivial solution — it makes it inexpressible." [S02, §1.2.4, p. 22] Rebuilding attention as descent is therefore not a change of vocabulary: it requires that transport carry information that the trivial solution cannot.

Relation to existing work. Sheaf diffusion and sheaf neural networks introduced restriction maps into graph learning [HG19; BDCLB22]; topos-theoretic designs for language models have been proposed [Mah25]; circuit-complexity results bound what constant-depth attention computes [MS23]. The thesis differs in making gluing and obstruction outputs of the kernel, and in stating each architectural claim with its grade.

What would refute it. The null outcomes of the registered comparisons of Problems 19, 20 and 22; or a proof that transport along restriction maps adds no expressive power over attention in a stated class.

Problems. In *The 57 Grothendieckian Problems*: Problems 19, 20, 22, 24, 25, 27, 28, 35.

Thesis 5 (*Report failure; abstain without evidence*).

> *A reasoning system should report where local information fails to glue, and abstain when independent overlap data are absent.*

Argument. When local results do not combine, the useful output is the location and size of the failure, not a number that averages it away. Obstruction theory gives the failure a mathematical form: a class, supported on the overlaps where it lives, that vanishes when the local data glue. The thesis makes this an obligation of a reasoning system: report the obstruction as an output, distinguish pairwise compatibility from higher coherence and from the existence of a global object, and abstain when the data needed to decide are missing rather than inventing them. The obligation has a cost and a limit. The cost is that some questions are left without an answer. The limit is that local checks cannot detect a falsehood that is consistent everywhere; [S02] records that it has no defence against such a falsehood.

Evidence in the seven works. "What fails to glue is measured as a class, located on the overlaps where it lives, and returned as an output; a head that cannot obtain independent overlap data abstains rather than inventing it." [S02, Preface, p. viii]

"none of them returns a number in place of an answer it does not have." [S02, The disciplines of a layer, p. xi]

In [S01, Vol. III, §63.4] compatibility, higher coherence, effectivity and gluing are separate coordinates: "Neither establishes that a global object exists." [S01, p. 9264]

An instance. An instrument can look like an obstruction and measure nothing. [S02] withdraws one of its own: on a graph, and on the nerve of a full simplex, second cohomology vanishes, and a first determinant construction of a gerbe class "is not only unproved but exactly zero" [S02, §9.1.7, p. 1766], since the cochain it defines is a coboundary by construction. The instrument is withdrawn, may not produce a gate decision, and may be reinstated only with a worked

non-zero example and a zero control computed before the labels are revealed. Reporting failure includes reporting the failure of an instrument.

RELATION TO EXISTING WORK. Obstruction classes in non-abelian cohomology [Gir71]; selective classification, in which a classifier may abstain [EW10]; detection of unsupported answers of language models by semantic entropy [FKKG24] and proposals to assess consistency with sheaves [HRH24]. The thesis asks for a certified class where these methods give a score.

WHAT WOULD REFUTE IT. A system that never abstains and reports scores only, with fewer errors at equal coverage than one that reports obstructions and abstains; or a proof that the obstruction cannot be computed at useful cost (Problem 21).

PROBLEMS. In *The 57 Grothendieckian Problems*: Problems 9, 16, 17, 21, 23, 30, 32, 47.

Thesis 6 (*A chain of non-implications*).

Operational completion does not imply mathematical validity, which does not imply epistemic promotion, which does not imply publication.

ARGUMENT. A distributed computation can finish without error while producing no mathematical object; a kernel can accept a proof of a statement that is not the statement intended; a valid result can rest on a premise whose status does not permit promotion; and a promoted result can still lack the authority to be published. Each passage requires its own input: a check of the mathematics, a check of fidelity to the intended statement, a decision on status, and an act of authorisation. The thesis is that a reasoning system must keep the four apart and must never infer one from another. Two of the inferences are easy to make without noticing: that a job which finished has produced a result, and that a kernel-checked proof of a formal statement proves the mathematics that the statement was meant to express.

EVIDENCE IN THE SEVEN WORKS. "The central rule is: operational completion $\not\Rightarrow$ mathematical validity $\not\Rightarrow$ epistemic promotion $\not\Rightarrow$ publication. Every implication requires a separately named input, judgment, and authority." [S01, Vol. III, Ch. 3, p. 8004]

"The central architectural error is to convert operational completion into global meaning." [S01, Vol. III, §56.1, p. 9119]

A machine proof remains a candidate "until source-target fidelity, kernel replay, and axiom-closure review pass under the same canonical subject key." [S01, Vol. III, §30.10, p. 8574]

AN INSTANCE. For the integration of a large system, [S01] applies the chain to the notion of a milestone: "A date, merged branch, authored chapter, installed service, successful demonstration, or dashboard showing green is not an integration milestone." A milestone exists only when its scope, gates, thresholds and decision authorities "were frozen before candidate evidence was inspected." [S01, Vol. III, §90.1, p. 9711] Each item in the first list is an instance of operational completion; none of them, on its own, is evidence that the integrated system does what it should.

Relation to existing work. The de Bruijn criterion separates proof search from proof checking by a small trusted kernel [BW05]; proof-carrying code separates the producer of a proof from its checker [Nec97]; the autoformalization literature measures the gap between a formal statement and its informal source [WJL+22], and public benchmarks of formal statements record misformalizations [FLM+26].

What would refute it. A class of problems in which one of the passages is shown to need no separate input — for example a fidelity checker that makes validity of the formal statement imply validity of the intended mathematics (Problem 5).

Problems. In *The 57 Grothendieckian Problems*: Problems 5, 12, 26, 29, 34.

Thesis 7 (*Machines propose; humans authorize*).

Machines may propose; humans authorize publication.

Argument. A system that can make its own output authoritative has no external check, however good its internal checks are. The thesis separates the right to propose from the right to promote and to publish, and places the second outside the producing system. It is a statement about capabilities, and can be stated and proved as such: a component without signing capability cannot produce a trusted signature; adding evidence to an artefact does not raise its status without an authorised transition. The seven works apply the same rule to their own production: machine assistance is recorded in their provenance, and the author is responsible for every claim. In 2026 the question has become practical for mathematics: a repository announced in October 2026 accepts works generated entirely by language models, and its listings record for each work the extent of machine generation and of human understanding [Ant26].

Evidence in the seven works. "Retrieval may supply candidate evidence but cannot change truth status." "Humans authorize publication." [S01, Architectural Prospectus, p. 7966]

Under its hypotheses, "a non-root module cannot, through a conforming Volume V interface, make its own output both authoritative and active." [S01, Vol. V, Proposition 1.3, p. 11913]

"Machine assistance is part of the provenance, not a co-author and not a substitute for proof." [S06, Preface, p. xiv]

An instance. The proposition that a non-root module cannot make its own output authoritative is stated with its failure cases. "If a worker can write directly to the status store, if an evaluator shares mutable code or keys with the candidate, if a signature is detached from the artifact hash, or if a promotion service accepts a producer's boolean assertion instead of replayable receipts, the proposition does not apply and the system has failed its constitution." [S01, Vol. V, Proposition 1.3, pp. 11913–11914] The separation of proposing from authorising is a property of capabilities that can be tested, and each of the four cases is a test.

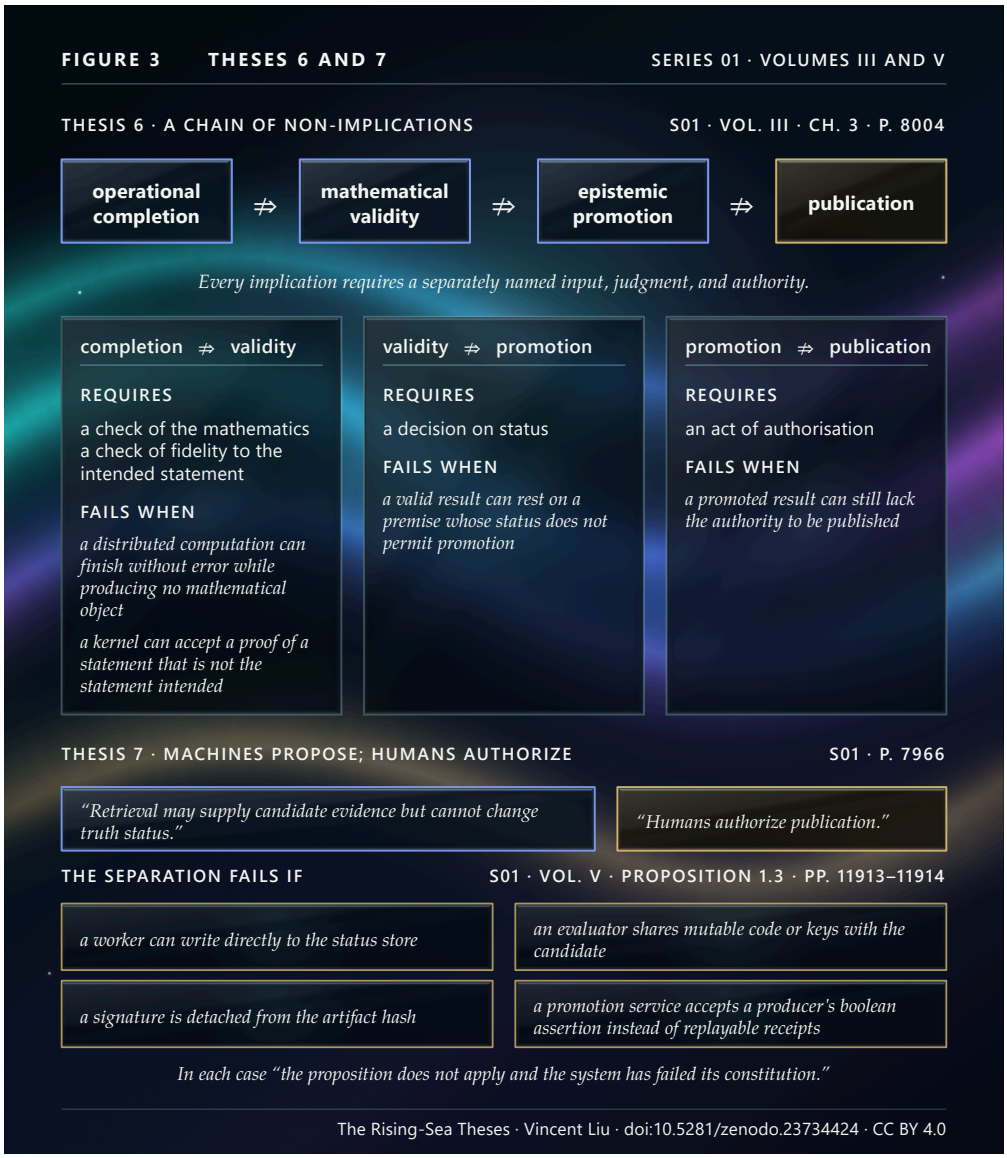


FIGURE 3. Theses 6 and 7. Above, the chain of non-implications of [S01, Vol. III, Ch. 3, p. 8004], with the input that each passage requires and the way in which it fails, from the argument of Thesis 6. Below, the separation of proposing from authorising [S01, Architectural Prospectus, p. 7966] and the four cases in which, by [S01, Vol. V, Proposition 1.3, pp. 11913–11914], it fails.

Relation to existing work. Separation of privilege and capability-based protection [SS75; DV66]; Thurston's account of proof as a social process of understanding [Thu94]; reward hacking, the failure that the separation guards against in self-improving systems [SHKK22].

What would refute it. A publication process without human authorisation whose rate of retraction and correction is no higher than that of a human-authorised process on comparable work; or a capability model in which the separation cannot be enforced.

Problems. In *The 57 Grothendieckian Problems*: Problems 10, 11.

Method

Thesis 8 (*The rising sea*).

> *Hard problems are better approached by raising the surrounding theory than by direct attack.*

Argument. Grothendieck described two ways of opening a hard nut: by blows, or by immersing it until it softens and opens under the hand [McL07]. The Weil conjectures are the model case: étale cohomology [SGA4] made the zeta function of a variety over a finite field a characteristic polynomial of Frobenius and the functional equation a consequence of duality, and the remaining statement, on the absolute values of the eigenvalues of Frobenius, was proved by Deligne [Del74]. The thesis takes the image as a method for machine reasoning, and the seven works make it operational: a rising-sea event begins from an explicit obstruction, proposes a typed extension, tests it, records what it costs, and is judged against the original difficulty. The thesis is a heuristic and is held to its outcomes. The five research subjects of the Programme report where the sea has risen and where it has not.

Evidence in the seven works. "The Weil conjectures were not proved by attacking the zeta function of a variety directly." "This book does not claim that such a sea has been raised around the Riemann Hypothesis." [S03, Preface, p. viii]

"The sea has not risen far enough: the base is specified, its local charts are established mathematics, and its global assembly rests on conjectures of the first rank of difficulty." [S07, The place of this work in the Programme, p. xi]

"Volume III does not turn that strategy into a universal theorem of problem solving. It turns it into a falsifiable campaign record." [S01, Vol. III, §5.1, p. 8034]

An instance. [S07] places the arithmetic Langlands programme between the two ways of opening the nut: "Most of the theorems of six decades on the arithmetic side are of the first kind: Taylor–Wiles patching, the stabilised trace formula and the Langlands–Kottwitz method are each a precise blow at one point. The geometric programme over a curve is of the second kind, and it has succeeded there." [S07, p. xi] The work tries the second way over $\mathbb{Z}$, and reports that the sea has not risen far enough; the problems it leaves are stated in *The 57 Grothendieckian Problems*.

Relation to existing work. McLarty's study of Grothendieck on simplicity and generality [McL07]; the image gives its title to a recent textbook of algebraic geometry [Vak25]. Lakatos' account of proofs and refutations describes a different dynamics, in which the concept changes under counterexamples [Lak76];

machine-guided discovery has so far worked mostly by search over a fixed space [DVB+21; RBN+24].

WHAT WOULD REFUTE IT. Comparative evidence that, for a reasoning system at equal cost, direct search settles more problems than theory-building does; or the research subjects of the Programme yielding no result that the extended setting made tractable.

PROBLEMS. In *The 57 Grothendieckian Problems*: Problems 1, 36, 37, 38, 41, 51, 52, 55, 56.

Thesis 9 (*State what is not established*).

A result is not complete until it states what it does not establish.

ARGUMENT. A theorem states what follows from its hypotheses. A reader also needs to know what does not follow: which converse fails, which extension is unproved, which reading of the result is wrong. For machine-produced mathematics the second statement matters more, because the volume of output can exceed what readers can check, so that the limits of a result must be stated where the result is stated. The thesis makes the negative statement part of the result: each chapter closes with what it does not establish, each programme states in advance what would falsify it, and a computation is reported as a finite computation. A result stated this way can be refuted, and a result that cannot be refuted has not said what it claims.

EVIDENCE IN THE SEVEN WORKS. Each chapter of [S03] "closes with a section entitled "What this chapter does not establish"." [S03, p. x]

"A book that cannot be refuted has said nothing." [S02, §13.16.11, p. 3176]

"A programme that cannot say what would falsify it is not a programme". [S04, §2.8.1, p. 19]

AN INSTANCE. [S03] measures the inertia of sixty-three Gram matrices of the pole-removed Weil form and finds one negative direction in every case. It proves, under the Riemann Hypothesis, that there is at most one; it then states what it has not shown: "The equality $n-(BN) = 1$ is therefore, in this book, an [E] measurement (Proposition 46.5) and not a theorem, conditional or otherwise." [S03, Warning 46.12, p. 919] The measurement becomes Problem 39 of *The 57 Grothendieckian Problems*.

RELATION TO EXISTING WORK. Falsifiability as a criterion of empirical content [Pop05]; preregistration, which fixes in advance what a study will count as evidence [NEDM18]; the labelling of speculative mathematics [JQ93].

WHAT WOULD REFUTE IT. Evidence that statements of what is not established add nothing that the hypotheses of the positive statement do not already convey — for example, that they can be generated mechanically from the hypotheses — or that readers misuse results no less often without them.

PROBLEMS. In *The 57 Grothendieckian Problems*: Problems 33, 39, 40, 43, 44, 45, 48, 49, 53, 54.

Thesis 10 (*Discovery is recursive*).

Mathematical discovery is recursive: propose, test against what is known, abstract what survives, ask again.

Argument. A proposed structure is tested first against the cases in which the answer is already known. What survives is abstracted into a new setting, and the question is asked again there; what does not survive is kept, as an obligation, an obstruction, a conjecture or a falsification test, with its status and its place in the record. The thesis is that this recursion, carried out in the open, is a method a machine can execute and a mathematician can check. The five research subjects of the Programme are records of it, and each ends not in a claimed solution but in a ledger of what was proved, what was excluded and what must still be constructed. In machine form, a failure becomes input: a typed obstruction is turned into a scoped conjecture with a falsifier.

Evidence in the seven works. The Series Note shared by [S03; S04; S05; S06; S07]: a discipline of recursive mathematical discovery, "in which a proposed structure is tested, what survives the test is abstracted into a new setting, and the question is posed again in that setting." [S03, p. iv]

"What survived the procedure was abstracted, and the question was posed again at the new level. This is the recursion the Programme calls recursive mathematical discovery, and the book is its record." [S03, Preface, p. ix]

"What does not survive is not discarded. It becomes a conjecture, a gap or a falsification test, with a statement, a status and a place in the ledger, and it is carried forward as part of the result." [S07, p. xi]

An instance. The recursion of [S03] can be followed step by step. "The positive-definite polarization, in the form first considered, is stronger than RH rather than a mechanism from which RH follows." [S03, Preface, p. ix] The failure was kept and analysed, and the next formulation came out of it: "The refutation of the positive polarization is the step that identified the index-one formulation." [S03, p. ix] A failure that is discarded yields nothing; one that is recorded with its reason changes the question.

Relation to existing work. Lakatos' method of proofs and refutations [Lak76]; automated conjecturing [Faj88; RGM+21]; search loops with an external evaluator [RBN+24; TWL+24]. The thesis differs in requiring that the failures be kept as typed records and carried forward.

What would refute it. Machine runs of the recursion that, on problems where the answer is known, rediscover no known result and produce no conjecture that survives its falsifiers (Problem 3); or a body of discoveries in which no abstraction step can be identified.

Problems. In *The 57 Grothendieckian Problems*: Problems 2, 3.

Thesis 11 (*Creating language*).

A mathematical reasoning system must be able to create notation and language, not only use them.

ARGUMENT. Advances in mathematics often take the form of new language: a definition that names a recurring structure, a notation that makes an argument short, a framework in which a family of problems becomes one problem. Schemes and topoi are examples [SGA4]. A system confined to a fixed language can search that language but cannot make such advances. The thesis requires the capacity to create language, and requires that the creation be accountable: a new primitive must be grounded in the structures and counterexamples that motivated it; a notation must declare what it compresses and what it loses; and the metatheory of a new language — substitution, preservation, conservativity — is an obligation to be discharged, not a property to be assumed.

EVIDENCE IN THE SEVEN WORKS. "The language forge follows a deliberately articulated cycle: collect, mine, name, notate, ground, reconstruct, axiomatize, grow, re-encode, and validate." "A language can be concise, expressive, elegant, or machine-tractable while semantic adequacy and soundness remain open." [S01, p. 7956]

"Notation earns trust by declaring what it compresses, what it preserves, and what it cannot recover." [S01, Vol. III, §43.14, p. 8873]

AN INSTANCE. A notation is accountable for what it compresses. In [S01], a round trip from one representation to another and back is evidence of exactly one thing: "A successful round trip establishes the equivalence promised by its profile. It does not establish semantic identity unless that judgment is separately supplied. It does not establish truth. It does not constitute proof." [S01, Vol. III, §43.14, p. 8873] A system that creates language must keep these four judgments apart in the same way as the passages of Thesis 6.

RELATION TO EXISTING WORK. The linguistic and philosophical study of mathematical language [Gan13]; mathematical knowledge management [CF09]; modal and multimodal type theories as languages with several modes [GKNB21].

WHAT WOULD REFUTE IT. A system restricted to a fixed language that matches a language-creating system on discovery tasks of the kind of Problem 14; or a language-creating system whose new primitives never survive review.

PROBLEMS. In *The 57 Grothendieckian Problems*: Problems 14, 57.

Thesis 12 (*Judge a method by its open ledger*).

A method should be judged by an open ledger of what it establishes, what it reduces to stated hypotheses, and what it leaves open.

ARGUMENT. A framework that addresses a hard problem invites two misreadings: that it has solved the problem, and that it has done nothing. A ledger answers both. It lists what has been proved unconditionally, what has been proved from named hypotheses, what remains open and what is disputed, and it can be checked entry by entry. The thesis is that a method — and in particular a method of machine reasoning — should be assessed by such a ledger, published with the work, rather than by its most ambitious statement. The seven works publish ledgers whose counts include their negative results: the number of cases of a conjecture established, which may be zero, is part of the record.

EVIDENCE IN THE SEVEN WORKS. The output of [S04] by status includes the row "Cases of BSD established 0", and the source states: "That ratio — eleven of

twelve against, one of twelve in favour — is the single most informative number in the book". [S04, p. 437]

The recursion of [S07] "ends in a statement ledger of one hundred and eleven items, twenty-eight axioms and eighty-three entries, fifty-three of them conjectural". [S07, p. v]

An instance. A ledger is itself a claim and can be wrong. [S06] says so of its own: "Whether the ledger of this volume is sound in the sense of Definition 55.6.7 is not established." [S06, Criterion 55.6.10, p. 1152] The ledger is offered as an object to be checked, and the same work names the most useful kind of reader: one "who finds a false obligation, a circular dependency, or a certified failure of a locked comparison" [S06, Preface, p. xiv].

Relation to existing work. Registered reports and preregistration in the empirical sciences [NEDM18]; Jaffe and Quinn's proposal for labelled speculative mathematics [JQ93]; benchmarks that record each item's status and provenance [FLM+26].

What would refute it. Ledgers that cannot be made checkable at the scale of a book — [S06] itself records that the soundness of its ledger is not established — or that readers find less informative than the positive statements.

Problems. In *The 57 Grothendieckian Problems*: Problems 13, 42, 46, 50.

5. How the theses fit together

The theses are not independent. Theses 1 and 2 together say what an artefact is: an object in a declared world, with a status that belongs to its type. Thesis 6 depends on both, because the passages it separates — from completion to validity, from validity to promotion, from promotion to publication — are changes of status that only a typed artefact can record. Thesis 7 locates the last of these passages outside the producing system. Theses 4 and 5 describe the reasoning step that the others constrain: local information is transported along restriction maps and glued where it can be, and the system reports where it cannot. Theses 8 to 12 describe the method of discovery that the architecture is meant to support, and the way its results should be reported and judged. Figure 4 draws these relations together with the two stated below.

A worked example from [S01] shows several theses at once (Figure 5). Three charts U_1, U_2, U_3 carry local sections s_1, s_2, s_3 describing the same finite object, each in a chart-specific normalisation. The three pairwise comparisons pass: "All three edge predicates pass." But one chart applies its normalisation after transport, and the direct path from the first chart to the third applies it before, so that in the declared normalisation system $t_{23} \circ t_{12} \neq t_{13}$. A checker of triangles reports the failure, and its record keeps what is needed to reproduce it: the hashes of the three sections, the three local receipts, the oriented face, the transition digests, the two composite results, the normalisation rule, the equality predicate, the checker and its environment, and the counterexample. The record excludes an unrelated fourth chart. The outcome is stated without loss in either direction: "The three pairwise receipts remain historically valid; they simply do not establish the failed triangular law." [S01, Vol. III, §61.18, p. 9227]

In this example the world records of Thesis 1 are what make the failure locatable: without the normalisation rule and the equality predicate there is

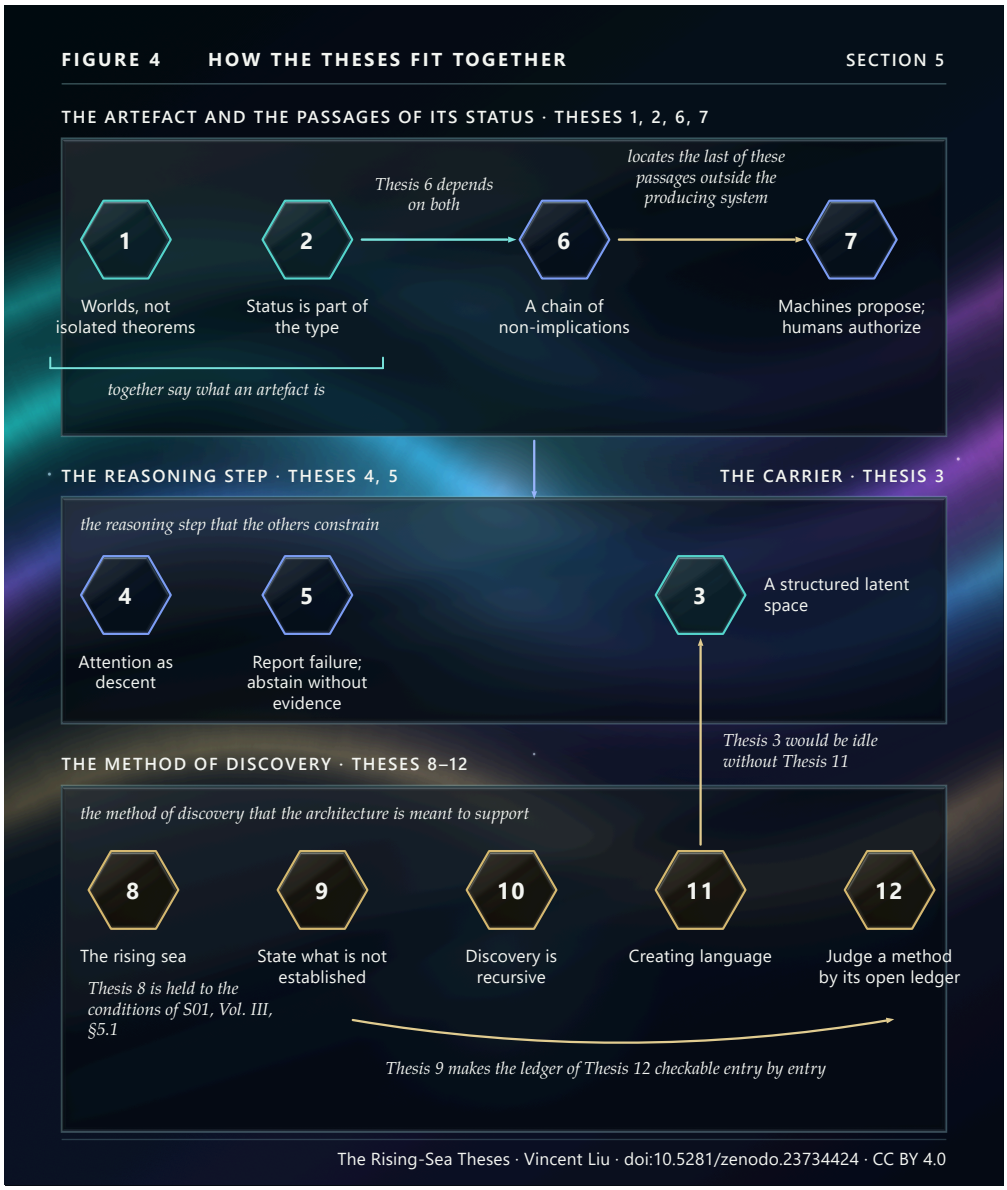


FIGURE 4. The relations between the theses that this section states, each labelled with its words. Nodes are coloured by group as in Figure 2 and link to the theses.

nothing to compare. The statuses of Thesis 2 are what keep the pairwise receipts valid while the triangle fails. Thesis 5 is the report itself, an obstruction with a location rather than a score. Thesis 6 is the refusal to read three successful comparisons as a successful gluing, and Thesis 10 is what happens next: the failure becomes the input of a proposal to change the normalisation bridge or to split the cover, which must itself be admitted and rechecked. Nothing in the example is deep mathematics. Its point is that each of these steps can be skipped, and that skipping any one of them turns true local statements into a false global one.

The theses also constrain one another. Thesis 3, that reasoning needs a structured carrier, would be idle without Thesis 11: a carrier whose vocabulary is fixed cannot represent the structures that a new problem requires. Thesis 9, that a result must state what it does not establish, is what makes the ledger of Thesis



FIGURE 5. The worked example of this section, after [S01, Vol. III, §61.18, p. 9227]: three charts carry local sections of one finite object; the three pairwise comparisons pass and the triangular law fails. On the right, what the record of the failure keeps and what it excludes; below, the step at which each of Theses 1, 2, 5, 6 and 10 acts. Each panel of the lower row links to its thesis.

12 checkable entry by entry. And Thesis 8, the rising sea, would be indistinguishable from changing the question if it were not held to the conditions that [S01] attaches to it: that an extension which deletes the predicate expressing a contradiction, forgets the invariant detecting failure, weakens equality until disagreement disappears, or renames an unknown as a primitive has not resolved the obstruction [S01, Vol. III, §5.1, p. 8034].

6. Limits of the evidence

The evidence offered for the theses is of three kinds, and its weight differs.

Proved statements. Some passages cited above are theorems with proofs in the source: the vanishing of second cohomology on graphs [S02, §9.1.7], the counterexample to coverwise transport [S02, Proposition 2.4.24], the conditional bound on the index of the pole-removed Weil form [S03, Corollary 46.11]. These are as reliable as their proofs, which have not been checked by others. They support the theses only indirectly: they show that the mathematics the theses rely on can be stated and, in places, proved.

Specifications. Most of the architecture is specified and not built. The chapters of the third volume of [S01] record their implementation status as "specification

only" [S01, Vol. III, p. 8218], and [S02] reports that no row of its claim ledger exceeds the lowest licence grade [S02, Proposition 1.2.3]. Theses 3, 4, 5 and 7 are therefore theses about a design. They could be wrong in a way the texts cannot show: a structured carrier may be too slow, an obstruction too expensive to compute, a capability model too restrictive to use. The problems attached to these theses are designed to find out.

Records of method. The five research subjects show what the method of Theses 8, 10 and 12 produces when it is carried through: ledgers, obstructions, conditional chains, falsification tests. They do not show that the method is better than another, because no comparison has been made. A reader may also observe that the records were written by the author of the method, with the assistance of AI tools, and that a method assessed by its own records has not been assessed. The theses are offered with that limit stated.

Three further limits should be named. First, the theses were selected by the author of the works from those works; a reader starting from the same material might select differently. Second, the seven works are first drafts, and the literature check carried out for *The 57 Grothendieckian Problems* found statements in them that are already superseded — for example, a construction the source describes as unknown has since been published [AMP26]; other errors of this kind are likely. Third, several theses are partly normative. Thesis 7, that humans authorise publication, is a proposal about how mathematical work should be governed, and evidence can show what it costs but not settle whether it is right.

The theses would be weakened in an identifiable way by any of the following: a reasoning system without world records, status types or obstruction reports that matches one with them on the tasks of *The 57 Grothendieckian Problems*; an analysis showing that the separation of fidelity from validity can be closed mechanically for the mathematics that matters; or the five research subjects, examined by specialists, turning out to record nothing that the extended settings made tractable. Each of these is a possible outcome, and the problems of *The 57 Grothendieckian Problems* are a way of reaching them.

7. Conclusion

The twelve theses describe a way of combining mathematics with machine reasoning in which the context of a result is an object, its status is part of its type, its failures are outputs, and its authority comes from outside the system that produced it. They are drawn from seven works that apply the same discipline to themselves, at a length of 26,021 pages, and that report what they have not established alongside what they have.

The theses are not established by being stated. Several of them are claims about what will work, and *The 57 Grothendieckian Problems* turns them into fifty-seven problems with decision criteria. A thesis whose problems are solved in the direction it predicts gains support; a thesis whose problems are solved against it should be revised or withdrawn, and the record of that should be kept, as the theses themselves require.

Declarations

Use of AI tools. The main ideas and the architecture of the Grothendieckian Programme are the author's and come from the author's research at WiseMotif.ai. In the seven works, AI tools assisted with drafting and with part of the literature collation; all of the text was reviewed by the author. Lean verification of the seven works is incomplete. Because of their size, the works are first drafts (Research Architecture Edition 1) and have not been peer-reviewed; they need review by experts and further work on the programme. They are released openly for that purpose, in the hope of further contributions and of more recursive mathematical discovery in AI for mathematics. For the present paper, statements were extracted from the published works by script and checked against the PDFs; AI tools assisted with drafting and with literature collation; the author selected the theses and problems and reviewed every statement.

Funding and competing interests. No external funding. The author is affiliated with WiseMotif.ai.

Data availability. The source texts are the sealed PDFs of the seven works and of the overview, each with a Zenodo DOI given in the references. The extraction scripts and the list of extracted items, the record of the literature check, the reference check, the Lean statements, the figures with their data and sources, and the machine-readable file of the paper are in the repository https://github.com/AlienMuse/grothendieckian-problems.

Licence. This paper, its figures and its data are released under the Creative Commons Attribution 4.0 International licence (CC BY 4.0); the code in the repository — the Lean files, the scripts and the source code of the figures — under the Apache License 2.0. The seven works and the overview are released under CC BY-NC-ND 4.0, and the quotations from them in this paper are reproduced under that licence with attribution.

Name. The adjective *Grothendieckian* names a mathematical method used in the seven works. It carries no attribution of design or endorsement to Alexander Grothendieck or to any other mathematician whose work is cited; the image of the rising sea is his, as reported in [McL07].

References

[S00] Vincent Liu, *The Grothendieckian Programme: Recursive Mathematical Discovery in the Age of Machine Reasoning*, Zenodo, 2026, An overview of Series 01–07; WiseMotif Monograph of Mathematical Intelligence, Series 00, doi:10.5281/zenodo.23669107.

[S01] Vincent Liu, *GAI-MUA: The Grothendieckian Machine. A Mathematical Universe Architecture for Verifiable and Self-Evolving Artificial Intelligence*, Zenodo, 2026, WiseMotif Monograph of Mathematical Intelligence, Series 01; 6 volumes, doi:10.5281/zenodo.23480696.

[S02] Vincent Liu, *GM-Omega-Infinity: Topos-Native Intelligence. A Sheaf-Theoretic Architecture for Verifiable Machine Intelligence*, Zenodo, 2026, Series 02; 5 volumes, doi:10.5281/zenodo.23538544.

[S03] Vincent Liu, *Arithmetic Spectral Descent: Obstruction Theory, Hodge–Krein Realization, and a Falsifiable Architecture for the Riemann Hypothesis*, Zenodo, 2026, Series 03; 4 volumes, doi:10.5281/zenodo.23564898.

[S04] Vincent Liu, *Fractured Adelic Geometry: A Foundational Framework for the Birch and Swinnerton-Dyer Conjecture*, Zenodo, 2026, Series 04; 4 volumes, doi:10.5281/zenodo.23585887.

[S05] Vincent Liu, *The Archimedean Prism: Tame Period Geometry and the Hodge Conjecture*, Zenodo, 2026, Series 05; 5 volumes, doi:10.5281/zenodo.23597954.

[S06] Vincent Liu, *Strained Prismatic Geometry: Deformation over the Absolute Base and the abc Conjecture*, Zenodo, 2026, Series 06; 5 volumes, doi:10.5281/zenodo.23610034.

[S07] Vincent Liu, *The Arithmetic Topos: Foundations of a Trace-Theoretic Approach to the Langlands Programme*, Zenodo, 2026, Series 07; 5 volumes, doi:10.5281/zenodo.23629681.

[AMP26] Gabriel Angelini-Knoll, Mona Merling, and Maximilien Péroux, *Topological ΔG-homology of rings with twisted G-action*, Adv. Math. (2026), 110907, doi:10.1016/j.aim.2026.110907.

[Ant26] Ben Antieau, *Hexagon*, 2026, guest post on T. Tao's blog What's new, 6 October 2026; the repository is at https://hexagonmath.org (consulted 28 October 2026), https://terrytao.wordpress.com/2026/10/06/hexagon/.

[Avi24] Jeremy Avigad, *Mathematics and the formal turn*, Bull. Amer. Math. Soc. (N.S.) **61** (2024), 225–240, doi:10.1090/bull/1832.

[BBCV21] Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković, *Geometric deep learning: grids, groups, graphs, geodesics, and gauges*, preprint, 2021, arXiv:2104.13478.

[BDCLB22] Cristian Bodnar, Francesco Di Giovanni, Benjamin Paul Chamberlain, Pietro Liò, and Michael M. Bronstein, *Neural sheaf diffusion: a topological perspective on heterophily and oversmoothing in GNNs*, Advances in Neural Information Processing Systems 35 (NeurIPS 2022), 2022, arXiv:2202.04579.

[BW05] Henk Barendregt and Freek Wiedijk, *The challenge of computer mathematics*, Philos. Trans. R. Soc. A **363** (2005), 2351–2375, doi:10.1098/rsta.2005.1650.

[CF09] Jacques Carette and William M. Farmer, *A review of mathematical knowledge management*, Intelligent Computer Mathematics (MKM 2009), Lecture Notes in Comput. Sci. 5625, Springer, 2009, pp. 233–246, doi:10.1007/978-3-642-02614-0_21.

[Del74] Pierre Deligne, *La conjecture de Weil. I*, Publ. Math. Inst. Hautes Études Sci. **43** (1974), 273–307, doi:10.1007/BF02684373.

[DV66] Jack B. Dennis and Earl C. Van Horn, *Programming semantics for multiprogrammed computations*, Comm. ACM **9** (1966), 143–155, doi:10.1145/365230.365252.

[DVB+21] Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, Marc Lackenby, Geordie Williamson, Demis Hassabis, and Pushmeet Kohli, *Advancing mathematics by guiding human intuition with AI*, Nature **600** (2021), 70–74, doi:10.1038/s41586-021-04086-x.

[EW10] Ran El-Yaniv and Yair Wiener, *On the foundations of noise-free selective classification*, J. Mach. Learn. Res. **11** (2010), 1605–1641, https://www.jmlr.org/papers/v11/el-yaniv10a.html.

[Faj88] Siemion Fajtlowicz, *On conjectures of Graffiti*, Discrete Math. **72** (1988), 113–118, doi:10.1016/0012-365X(88)90199-9.

[FKKG24] Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal, *Detecting hallucinations in large language models using semantic entropy*, Nature **630** (2024), 625–630, doi:10.1038/s41586-024-07421-0.

[FLM+26] Moritz Firsching, Paul Lezeau, Salvatore Mercuri, Miklós Z. Horváth, Yaël Dillies, Calle Sönne, Eric Wieser, Fred Zhang, Thomas Hubert, Blaise Agüera y Arcas, and Pushmeet Kohli, *Formal Conjectures: an open and evolving benchmark for verified discovery in mathematics*, preprint, 2026, arXiv:2605.13171.

[Gan13] Mohan Ganesalingam, *The Language of Mathematics: A Linguistic and Philosophical Investigation*, Lecture Notes in Comput. Sci. 7805, Springer, 2013, doi:10.1007/978-3-642-37012-0.

[Gir71] Jean Giraud, *Cohomologie non abélienne*, Grundlehren der mathematischen Wissenschaften 179, Springer, 1971, doi:10.1007/978-3-662-62103-5.

[GKNB21] Daniel Gratzer, G. A. Kavvos, Andreas Nuyts, and Lars Birkedal, *Multimodal dependent type theory*, Log. Methods Comput. Sci. **17** (2021), no. 3, 11:1–11:67, doi:10.46298/lmcs-17(3:11)2021.

[GLD+24] Bruno Gavranović, Paul Lessard, Andrew Dudzik, Tamara von Glehn, João G. M. Araújo, and Petar Veličković, *Position: categorical deep learning is an algebraic theory of all architectures*, Proceedings of the 41st International Conference on Machine Learning (ICML 2024), PMLR 235, 2024, arXiv:2402.15332.

[HG19] Jakob Hansen and Robert Ghrist, *Toward a spectral theory of cellular sheaves*, J. Appl. Comput. Topol. **3** (2019), 315–358, doi:10.1007/s41468-019-00038-7.

[HG20] Jakob Hansen and Thomas Gebhart, *Sheaf neural networks*, preprint, 2020, arXiv:2012.06333.

[HMS17] Annette Huber and Stefan Müller-Stach, *Periods and Nori Motives*, Ergebnisse der Mathematik und ihrer Grenzgebiete 65, Springer, 2017, doi:10.1007/978-3-319-50926-6.

[HRH24] Steve Huntsman, Michael Robinson, and Ludmilla Huntsman, *Prospects for inconsistency detection using large language models and sheaves*, preprint, 2024, arXiv:2401.16713.

[JQ93] Arthur Jaffe and Frank Quinn, *"Theoretical mathematics": toward a cultural synthesis of mathematics and theoretical physics*, Bull. Amer. Math. Soc. (N.S.) **29** (1993), 1–13, doi:10.1090/S0273-0979-1993-00413-0.

[Lak76] Imre Lakatos, *Proofs and Refutations: The Logic of Mathematical Discovery*, Cambridge University Press, 1976, doi:10.1017/CBO9781139171472.

[Liu26a] Vincent Liu, *The Grothendieckian Programme: An Overview — Seven Open Works on Mathematics and Machine Reasoning*, 2026, preprint, Zenodo; the first of three papers released together, doi:10.5281/zenodo.23734421.

[Liu26c] Vincent Liu, *The 57 Grothendieckian Problems: Open Problems for Mathematics and Machine Reasoning from the Grothendieckian Programme*, 2026, preprint, Zenodo; the third of three papers released together, doi:10.5281/zenodo.23734426.

[Mah25] Sridhar Mahadevan, *Topos theory for generative AI and LLMs*, preprint, 2025, arXiv:2508.08293.

[mC20] The mathlib Community, *The Lean mathematical library*, Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020), ACM, 2020, pp. 367–381, doi:10.1145/3372885.3373824.

[McL07] Colin McLarty, *The rising sea: Grothendieck on simplicity and generality*, Episodes in the History of Modern Algebra (1800–1950) (Jeremy J. Gray and Karen Hunger Parshall, eds.), History of Mathematics 32, Amer. Math. Soc., 2007, pp. 301–325.

[MS23] William Merrill and Ashish Sabharwal, *The parallelism tradeoff: limitations of log-precision transformers*, Trans. Assoc. Comput. Linguist. **11** (2023), 531–545, doi:10.1162/tacl_a_00562.

[MU21] Leonardo de Moura and Sebastian Ullrich, *The Lean 4 theorem prover and programming language*, Automated Deduction – CADE 28, Lecture Notes in Comput. Sci. 12699, Springer, 2021, pp. 625–635, doi:10.1007/978-3-030-79876-5_37.

[Nec97] George C. Necula, *Proof-carrying code*, Proceedings of the 24th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL '97), ACM, 1997, pp. 106–119, doi:10.1145/263699.263712.

[NEDM18] Brian A. Nosek, Charles R. Ebersole, Alexander C. DeHaven, and David T. Mellor, *The preregistration revolution*, Proc. Natl. Acad. Sci. USA **115** (2018), 2600–2606, doi:10.1073/pnas.1708274114.

[Pop05] Karl Popper, *The Logic of Scientific Discovery*, Routledge, 2005, Routledge Classics; first English edition 1959, doi:10.4324/9780203994627.

[RBN+24] Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi, *Mathematical discoveries from program search with large language models*, Nature **625** (2024), 468–475, doi:10.1038/s41586-023-06924-6.

[RGM+21] Gal Raayoni, Shahar Gottlieb, Yahel Manor, George Pisha, Yoav Harris, Uri Mendlovic, Doron Haviv, Yaron Hadad, and Ido Kaminer, *Generating conjectures on fundamental constants with the Ramanujan Machine*, Nature **590** (2021), 67–73, doi:10.1038/s41586-021-03229-4.

[Sch22] Peter Scholze, *Liquid tensor experiment*, Exp. Math. **31** (2022), 349–354, doi:10.1080/10586458.2021.1926016.

[SGA4] Michael Artin, Alexander Grothendieck, and Jean-Louis Verdier, *Théorie des topos et cohomologie étale des schémas (SGA 4), Tome 1*, Lecture Notes in Math. 269, Springer, 1972, doi:10.1007/BFb0081551.

[SHKK22] Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger, *Defining and characterizing reward hacking*, Advances in Neural Information Processing Systems 35 (NeurIPS 2022), 2022, arXiv:2209.13085.

[SS75] Jerome H. Saltzer and Michael D. Schroeder, *The protection of information in computer systems*, Proc. IEEE **63** (1975), 1278–1308, doi:10.1109/PROC.1975.9939.

[Thu94] William P. Thurston, *On proof and progress in mathematics*, Bull. Amer. Math. Soc. (N.S.) **30** (1994), 161–177, doi:10.1090/S0273-0979-1994-00502-6.

[TWL+24] Trieu H. Trinh, Yuhuai Wu, Quoc V. Le, He He, and Thang Luong, *Solving olympiad geometry without human demonstrations*, Nature **625** (2024), 476–482, doi:10.1038/s41586-023-06747-5.

[Vak25] Ravi Vakil, *The Rising Sea: Foundations of Algebraic Geometry*, Princeton University Press, 2025, doi:10.1515/9780691268682.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin, *Attention is all you need*, Advances in Neural Information Processing Systems 30 (NIPS 2017), 2017, arXiv:1706.03762.

[WJL+22] Yuhuai Wu, Albert Q. Jiang, Wenda Li, Markus N. Rabe, Charles Staats, Mateja Jamnik, and Christian Szegedy, *Autoformalization with large language models*, Advances in Neural Information Processing Systems 35 (NeurIPS 2022), 2022, arXiv:2205.12615.