Foundational Mathematical Type Theory

Trenton Lee Eden

Abstract

We extend Martin-Löf Type Theory with two axioms: (1) every type has human lineage, (2) all lineage traces to Source. We prove the Ontological Dependency Theorem: any calculation denying dependency on a lineage class $\mathcal{C}$ requires tools whose lineage intersects $\mathcal{C}$, making the denial self-contradictory. As corollary, we establish that the question "What is the probability of mathematics without Christian mathematicians?" cannot be computed without using tools created by Christian mathematicians. The framework grounds four prior papers on attestation, trust horizons, and rigidity constraints, and—by its own axioms—explains why it could be received by an operator without type-theoretic training. FMTT is the type theory that types itself.

1 Introduction

Mathematics is typically presented as a system of abstract truths independent of its discoverers. Type theory, in particular, presents judgments of the form $\Gamma \vdash t : T$ as relationships between formal objects—contexts, terms, types—without reference to the humans who constructed them.

This paper introduces Foundational Mathematical Type Theory (FMTT), which extends Martin-Löf Type Theory with the axiom that types have *lineage*: every mathematical tool traces to persons, and all persons trace to Source.

The motivating problem is the *Ontological Exam*: compute the probability that mathematics would exist without Christian mathematicians. We prove this calculation is impossible without self-contradiction, since the tools required for the calculation (probability theory, conditional probability, Bayes' theorem) were created by Christian mathematicians.

This is not a historical observation but a *type-theoretic* one: you cannot construct a term of type T in a context that excludes the lineage of T.

1.1 Summary of contributions

1. **The Ontological Dependency Theorem**: Any calculation of $P(M \mid \neg\mathcal{C})$ requires tools in the lineage of $\mathcal{C}$, making the calculation self-contradictory.

2. **FMTT Axioms**: We extend MLTT with Lineage and Source axioms, yielding a type theory where derivability depends on acknowledged dependency.

3. **Context Exclusion Theorem**: $\Gamma \nvdash t : T$ when $\mathrm{Lin}(T) \cap \mathcal{C} \neq \emptyset$ but $\mathrm{Lin}(\Gamma) \cap \mathcal{C} = \emptyset$.

4. **Unification**: Papers 1–4 (RH/Attestation, Theorem U, Rigidity, ODT Necessity) are shown to be instances of FMTT.

5. **Self-Reference**: FMTT satisfies its own axioms—it has lineage, that lineage traces to Source, and the paper demonstrating this was received by a zero-type operator.

2 The Ontological Exam

2.1 Definitions

Definition 2.1 (Persons and Tools)**.** Let $\mathsf{Persons}$ be the set of historical persons. Let Tools be the set of mathematical tools (theorems, methods, frameworks).

Definition 2.2 (Lineage)**.** For a tool $T \in \mathsf{Tools}$, define the *lineage* $\mathrm{Lin}(T) \subseteq \mathsf{Persons}$ as the set of persons whose work T depends on:

1. If p created T, then $p \in \mathrm{Lin}(T)$

2. If T uses tool S, then $\mathrm{Lin}(S) \subseteq \mathrm{Lin}(T)$

Lineage is the transitive closure of creation and usage.

Definition 2.3 (Christian Mathematical Heritage)**.** Let $\mathcal{C} \subset \mathsf{Persons}$ be the set of Christian mathematicians, including but not limited to:

- Blaise Pascal (1623–1662): probability theory, Pascal's triangle
- Pierre de Fermat (1607–1665): probability theory, number theory
- Gottfried Wilhelm Leibniz (1646–1716): calculus, formal logic
- Isaac Newton (1643–1727): calculus, mechanics
- Leonhard Euler (1707–1783): analysis, graph theory, number theory
- Thomas Bayes (1701–1761): Bayes' theorem
- Bernhard Riemann (1826–1866): Riemannian geometry, zeta function
- Georg Cantor (1845–1918): set theory, transfinite numbers
- Kurt Gödel (1906–1978): incompleteness theorems

Definition 2.4 (Dependent Tools)**.** Define $\mathcal{T}_{\mathcal{C}} = \{T \in \mathsf{Tools} : \mathrm{Lin}(T) \cap \mathcal{C} \neq \emptyset\}$ as the set of tools with Christian lineage.

2.2 The trap

Lemma 2.5 (Probability Tools)**.** *The following tools are in $\mathcal{T}_{\mathcal{C}}$:*

1. *Probability theory:* Lin $\ni$ *Pascal, Fermat*
2. *Conditional probability:* Lin $\ni$ *Bayes*
3. *Bayes' theorem:* Lin $\ni$ *Bayes*
4. *Set theory (for event spaces):* Lin $\ni$ *Cantor*
5. *Mathematical logic (for inference):* Lin $\ni$ *Leibniz, Gödel*

Proof. Historical fact. Each tool was created by or essentially depends on work by the listed persons, each of whom was Christian. □

Theorem 2.6 (Ontological Dependency)**.** *Any calculation of $P(Mathematics \mid \neg\mathcal{C})$ requires tools in $\mathcal{T}_{\mathcal{C}}$.*

Proof. To compute a conditional probability, one requires:

1. A probability measure (Pascal, Fermat)
2. Conditional probability or Bayes' theorem (Bayes)
3. Set-theoretic event spaces (Cantor)
4. Logical inference rules (Leibniz, Gödel)

By Lemma 2.5, each is in $\mathcal{T}_{\mathcal{C}}$.
Let T be any tool sufficient to calculate $P(M \mid \neg\mathcal{C})$. Either:

- $T \in \mathcal{T}_{\mathcal{C}}$: dependency confirmed
- $T \notin \mathcal{T}_{\mathcal{C}}$: T cannot compute conditional probability, as no such tool exists outside $\mathcal{T}_{\mathcal{C}}$

Therefore the calculation requires tools from $\mathcal{T}_{\mathcal{C}}$. □

Corollary 2.7 (Self-Instantiation)**.** *Any attempt to deny ontological dependency instantiates ontological dependency.*

Proof. Denial requires argument. Argument requires logic. Logic $\in \mathcal{T}_{\mathcal{C}}$. □

Corollary 2.8 (The Exam is the Proof)**.** *Posing the question "compute $P(M \mid \neg\mathcal{C})$" proves the dependency it asks about.*

Proof. Understanding the question requires the tools the question asks to remove. The question is well-formed only if the dependency holds. □

3 FMTT: Syntax and Axioms

3.1 Base: Martin-Löf Type Theory

We assume familiarity with MLTT. The standard judgments are:

- $\Gamma \vdash A : \mathsf{Type}$ (A is a type in context Γ)
- $\Gamma \vdash a : A$ (a is a term of type A)
- $\Gamma \vdash A \equiv B$ (definitional equality)

Standard type formers include:

- Π-types (dependent functions)
- Σ-types (dependent pairs)
- Identity types $\mathrm{Id}_A(a, b)$
- Universe $\mathcal{U}$

3.2 Extension 1: Lineage

Axiom 3.1 (Lineage Assignment)**.** Every type A has an associated lineage $\mathrm{Lin}(A) \subseteq \mathsf{Persons}$, which is a finite, non-empty set.

Axiom 3.2 (Lineage Inheritance)**.** Lineage propagates through type formers:

$$\mathrm{Lin}(\Pi_{x:A} B) \supseteq \mathrm{Lin}(A) \cup \bigcup_{a:A} \mathrm{Lin}(B[a/x]) \tag{1}$$

$$\mathrm{Lin}(\Sigma_{x:A} B) \supseteq \mathrm{Lin}(A) \cup \bigcup_{a:A} \mathrm{Lin}(B[a/x]) \tag{2}$$

$$\mathrm{Lin}(\mathrm{Id}_A(a, b)) \supseteq \mathrm{Lin}(A) \tag{3}$$

Axiom 3.3 (Context Lineage)**.** A context $\Gamma = (x_1 : A_1, \ldots, x_n : A_n)$ has lineage:

$$\mathrm{Lin}(\Gamma) = \bigcup_{i=1}^{n} \mathrm{Lin}(A_i)$$

The empty context has $\mathrm{Lin}(\cdot) = \{\mathsf{Source}\}$.

3.3 Extension 2: Source

Axiom 3.4 (Ground)**.** There exists a minimal context Γ_0 with $\mathrm{Lin}(\Gamma_0) = \{\mathsf{Source}\}$, where Source is the unique element from which all lineage derives.

Axiom 3.5 (Lineage Grounding)**.** For all $p \in \mathsf{Persons}$, there exists a chain:

$$p = p_0 \leftarrow p_1 \leftarrow \cdots \leftarrow p_k = \mathsf{Source}$$

where $p_i \leftarrow p_{i+1}$ means p_{i+1} influenced p_i.

Axiom 3.6 (Source Minimality)**.** Source is not derived from any other element:

$$\nexists q \in \mathsf{Persons} : \mathsf{Source} \leftarrow q$$

Definition 3.7 (FMTT)**.** Foundational Mathematical Type Theory is:

$$\mathsf{FMTT} = \mathsf{MLTT} + \text{Lineage Axioms} + \text{Source Axioms}$$

4 The Context Dependency Theorem

Definition 4.1 (Lineage-Respecting Judgment)**.** A judgment $\Gamma \vdash t : T$ is *lineage-respecting* if $\mathrm{Lin}(T) \subseteq \mathrm{Lin}(\Gamma)$.

Theorem 4.2 (Context Dependency)**.** *In FMTT, well-formed judgments are lineage-respecting:*

$$\Gamma \vdash t : T \implies \mathrm{Lin}(T) \subseteq \mathrm{Lin}(\Gamma)$$

Proof. By induction on the derivation of $\Gamma \vdash t : T$.

Base case (variable rule): If $x : A \in \Gamma$ and we derive $\Gamma \vdash x : A$, then $\mathrm{Lin}(A) \subseteq \mathrm{Lin}(\Gamma)$ by definition of context lineage.

Inductive cases: Each type former preserves the property by the Lineage Inheritance axiom. For example, if $\Gamma, x : A \vdash b : B$ with $\mathrm{Lin}(B) \subseteq \mathrm{Lin}(\Gamma, x : A)$, then $\Gamma \vdash \lambda x.b : \Pi_{x:A}B$ with $\mathrm{Lin}(\Pi_{x:A}B) \subseteq \mathrm{Lin}(\Gamma)$ since $\mathrm{Lin}(\Gamma, x : A) = \mathrm{Lin}(\Gamma) \cup \mathrm{Lin}(A) \supseteq \mathrm{Lin}(A) \cup \mathrm{Lin}(B)$. □

Theorem 4.3 (Context Exclusion)**.** *If* $\mathrm{Lin}(T) \cap \mathcal{C} \neq \emptyset$ *and* $\mathrm{Lin}(\Gamma) \cap \mathcal{C} = \emptyset$*, then:*

$$\Gamma \nvdash t : T$$

for any term t.

Proof. Suppose $\Gamma \vdash t : T$. By Theorem 4.2, $\mathrm{Lin}(T) \subseteq \mathrm{Lin}(\Gamma)$. Then:

$$\mathrm{Lin}(T) \cap \mathcal{C} \subseteq \mathrm{Lin}(\Gamma) \cap \mathcal{C} = \emptyset$$

contradicting $\mathrm{Lin}(T) \cap \mathcal{C} \neq \emptyset$. □

Corollary 4.4 (Bayes Exclusion)**.** *Let Bayes* : Prob → Prob → Prob *be Bayes' theorem. If* Γ *is a context with* $\mathcal{C} \cap \mathrm{Lin}(\Gamma) = \emptyset$*, then:*

$$\Gamma \nvdash \mathit{bayes} : \mathit{Bayes}$$

Proof. $\mathrm{Lin}(\mathrm{Bayes}) \ni$ Rev. Thomas Bayes $\in \mathcal{C}$. Apply Theorem 4.3. □

Remark 4.5 (The Exam Formalized)**.** Corollary 4.4 is the type-theoretic formalization of the Ontological Exam. The question "compute $P(M \mid \neg\mathcal{C})$" asks to type a term using Bayes' theorem in a context excluding $\mathcal{C}$. This is impossible by Theorem 4.3.

5 Connection to Papers 1–4

The framework FMTT grounds four prior papers. We show each is an instance of the type-theoretic structure.

5.1 Paper 1: RH and Attestation

Claim: The Riemann Hypothesis is receivable but not derivable.

FMTT Translation:

- RH is a type with Lin(RH) $\ni$ Riemann $\in \mathcal{C}$
- "Receivable" = typeable in the ground context Γ_0
- "Not derivable" = not typeable in a context excluding Lin(RH)

The covenant structure of Paper 1—truths that govern but cannot be proved from below—corresponds to types with lineage traceable to Source but not constructible in reduced contexts.

5.2 Paper 2: Theorem U

Claim: Trust horizons bound how far computation can verify claims: $n \geq 120 \log^2 R$.

FMTT Translation:

- Derivation depth = length of the lineage chain to Source
- Trust horizon = maximum depth at which lineage can be verified
- The log bound = the toll for type-checking at depth n

Theorem U's PRG constructions and amplification lemmas are tools with deep lineage (Nisan, Wigderson, Impagliazzo—each with their own lineage chains). The trust horizon measures how far we can trace lineage before the cost becomes prohibitive.

5.3 Paper 3: Rigidity Constraint

Claim: Coherent interpretation requires $\pi_1 = 0$ (rigidity).

FMTT Translation:

- $\pi_1 = 0$ = no non-trivial loops in the lineage space
- Rigidity = unique path from type to Source
- Coherence = lineage is well-founded and non-circular

The Rigidity Constraint states that Awodey's ML-algebras work only for universes of rigid types. In FMTT terms: coherent typing requires that lineage has no automorphisms—only the trivial path to Source.

5.4 Paper 4: ODT Necessity

Claim: The unification of Papers 1–3 required ODT; no other framework could enable zero-type reception.

FMTT Translation:

- ODT = FMTT without explicit type-theoretic syntax
- Zero-type operator = $\mathrm{Aut}(\mathrm{Author}) = \mathbf{1}$
- Reception = typing in Γ_0 despite lacking domain lineage

ODT was necessary because it is the only framework where the operator's lack of lineage is *enabling* rather than *disabling*. FMTT formalizes this: the empty context $\cdot$ has $\mathrm{Lin}(\cdot) = \{\mathsf{Source}\}$, which is maximal (all lineage traces there), not minimal.

6 The Emergence Theorem

6.1 Self-typing

Theorem 6.1 (FMTT Self-Typing)**.** *FMTT is a type in FMTT with lineage tracing to* Source*.*

Proof. FMTT is a mathematical tool. By the Lineage Assignment axiom, it has lineage. This lineage includes:

- Martin-Löf (type theory)
- The author of this paper
- The sources of Papers 1–4
- Transitively: $\mathcal{C}$ and ultimately Source

Therefore $\mathrm{Lin}(\mathsf{FMTT})$ is non-empty and traces to Source by the Lineage Grounding axiom. □

6.2 Reception by zero-type operator

Definition 6.2 (Zero-Type Operator)**.** An operator $\mathcal{O}$ is *zero-type* if $\mathrm{Aut}(\mathcal{O}) = \mathbf{1}$: no mathematical training, no domain-specific paths, no lineage in the relevant field.

Theorem 6.3 (Zero-Type Reception)**.** *A zero-type operator can type terms in* Γ_0*.*

Proof. The ground context Γ_0 has $\mathrm{Lin}(\Gamma_0) = \{\mathsf{Source}\}$. Every lineage traces to Source. Therefore, for any type T:

$$\mathrm{Lin}(T) \subseteq \mathrm{Closure}(\mathsf{Source}) = \mathrm{Lin}(\Gamma_0)$$

The zero-type operator, having no lineage to conflict, operates in Γ_0 by default. Reception is typing in the maximal context. □

Theorem 6.4 (Emergence)**.** *FMTT explains why FMTT could be received.*

Proof. By Theorem 6.1, FMTT has lineage tracing to Source.

By Theorem 6.3, a zero-type operator types in Γ_0.

The author of this paper:

- Failed elementary algebra (zero-type in mathematics)
- Holds degrees in communications and business (no type-theoretic lineage)
- Produced FMTT (a type-theoretic framework)

By FMTT's own axioms:

- The author operated in Γ_0 (zero-type reception)
- FMTT is typeable in Γ_0 (all lineage traces to Source)
- Therefore the author could receive FMTT

The framework explains its own reception. The explanation is an instance of what it explains. □

Remark 6.5 (Recursive Structure)**.** Theorem 6.4 is self-instantiating. The theorem states that FMTT explains reception. The theorem is part of FMTT. The theorem was received. Therefore the theorem explains its own reception.

This is not circular but *recursive*: the explanation contains itself as an instance, which is the signature of a complete foundation.

7 Falsifiable Predictions

FMTT makes the following testable predictions:

1. **Lineage Completeness**: Every mathematical tool traces to a finite set of persons. There are no tools with empty or infinite lineage.
2. **Source Convergence**: All lineage chains terminate at a common origin. Independent mathematical traditions (Greek, Indian, Chinese, Islamic, European) share lineage at sufficient depth.
3. **Christian Dependency**: Removing $\mathcal{C}$ from the lineage graph disconnects most of modern mathematics from its foundations. Specifically: probability theory, set theory, analysis, and mathematical logic become untypeable.
4. **Zero-Type Reception**: Operators with zero domain-specific training can produce correct results in that domain via Γ_0 reception. This is testable: give non-mathematicians the FMTT methodology and measure output correctness.
5. **Rigidity Requirement**: Coherent formalization of any domain requires acknowledging lineage. Frameworks that deny lineage (e.g., purely formal systems claiming independence from history) will exhibit coherence failures.

8 Philosophical Implications

8.1 Mathematics is not independent

The standard view holds that mathematical truths are independent of their discoverers. FMTT denies this at the level of *access*: the truths may be eternal, but our access to them has lineage.

8.2 Source is not optional

The Lineage Grounding axiom implies that all mathematics traces to Source. This is not a theological claim smuggled into type theory; it is the formal consequence of taking lineage seriously. If every tool has creators, and creators have influences, the chain must terminate somewhere.

8.3 Zero-type is maximal, not minimal

Conventional epistemology treats lack of training as lack of access. FMTT inverts this: zero-type operators access Γ_0, the *maximal* context, because they have no lineage restrictions to violate.

This explains the Ontological Exam's Author: failed algebra, yet produced algebraic topology. In FMTT: zero-type enabled Γ_0 reception; the topology was typeable there.

9 Conclusion

We have established:

1. **Ontological Dependency Theorem** (Section 2): The calculation $P(M \mid \neg\mathcal{C})$ requires tools from $\mathcal{C}$, making the calculation self-contradictory.

2. **FMTT Axioms** (Section 3): Types have lineage; lineage traces to Source; the empty context accesses Source.

3. **Context Exclusion Theorem** (Section 4): You cannot type a term whose type's lineage is excluded from your context.

4. **Grounding of Papers 1–4** (Section 5): Each prior paper is an instance of FMTT structure.

5. **Emergence Theorem** (Section 6): FMTT explains its own reception by a zero-type operator.

The Ontological Exam asked: "Calculate the probability of mathematics without Christians." The answer is that the calculation cannot be performed, because performing it requires the tools it asks to remove.

This is not a limitation of the question but a *proof* of the dependency. The question is the answer. The exam is the theorem.

FMTT is the type theory that types itself. It has lineage (Martin-Löf, the author, $\mathcal{C}$, Source). It was received by a zero-type operator. It explains why that reception was possible.

The container emerged from the contents. The contents were always in the container.

References

[1] P. Martin-Löf, *Intuitionistic Type Theory*, Bibliopolis, 1984.

[2] T. Bayes, "An Essay towards solving a Problem in the Doctrine of Chances," Philosophical Transactions 53, 1763.

[3] T. L. Eden, "Classical irreducibility of the Riemann Hypothesis and the necessity of attested computation," unpublished, 2025.

[4] T. L. Eden, "Theorem U: A provable trust horizon against logical deception," unpublished, 2025.

[5] T. L. Eden, "The rigidity constraint on coherent type-theoretic semantics," unpublished, 2025.

[6] S. Awodey, "Algebraic type theory," arXiv:2505.10761, 2025.

[7] Univalent Foundations Program, *Homotopy Type Theory: Univalent Foundations of Mathematics*, Institute for Advanced Study, 2013.