Type Theory from Distinction: Simply Typed Lambda Calculus as Structural Necessity

Andrei Kozyrev

January 2026

Abstract

Modern mathematics and computer-verified proof systems are built on type theory, yet type theory itself is usually presented as an axiomatic starting point: types, equality, and functions are assumed rather than explained. We show that simply typed lambda calculus arises necessarily from a single meta-assumption: the existence of distinction ($\Delta \neq \emptyset$). From this assumption we construct a minimal calculus of distinctions in which types emerge as equivalence classes of indistinguishable elements, equality is induced by the absence of a distinguishing operation, and functions are exactly those mappings that preserve indistinguishability. Within this framework, the full soundness of simply typed lambda calculus is derived—not postulated—and a collapse theorem shows that if distinction is absent, all mathematical structure degenerates to a trivial system. The result is fully formalized and machine-verified in Agda, with no postulates or unproven assumptions.

1 Introduction

Type theory underlies formal verification, dependent programming languages, and much of contemporary foundations of mathematics. Yet the standard presentation treats types, equality, and functions as primitive notions, introduced by axiom or convention. This raises a foundational question: why is type theory a legitimate framework at all?

We address this gap by showing that simply typed lambda calculus (STLC) is not an arbitrary choice but a *structural necessity*—it arises from the mere possibility of making distinctions.

Contributions

1. A minimal calculus based on a single meta-assumption: distinction exists ($\Delta \neq \emptyset$).
2. A derivation of STLC soundness from this assumption.
3. A collapse theorem showing that absence of distinction eliminates all mathematical structure.
4. A complete Agda formalization with zero postulates.

2 Preliminaries and Scope

We assume no prior type theory. The meta-assumption is:

(A0) There exist entities that can be distinguished: $\exists x, y.\ x \not\sim y$.

This is not a choice of axioms but a necessary condition for any non-trivial formal system. We make no claims about physics, consciousness, or metaphysics—only about the structure of formal reasoning.

3 Distinction Calculus

Definition 1 (Distinction Structure)**.** *A* distinction structure *consists of:*

- *A universe U of entities*
- *A collection D of distinguishers*
- *For each $d \in D$, a set of labels L_d and an observation function* $\mathrm{eval}_d : U \to L_d$
- *Distinguishability:* $x \# y \iff \exists d.\ \mathrm{eval}_d(x) \neq \mathrm{eval}_d(y)$
- *Indistinguishability:* $x \sim y \iff \neg(x \# y)$

Theorem 2 (A1: Equivalence)**.** *The relation $\sim$ is an equivalence relation.*

Definition 3 (Δ-morphism)**.** *A function $f : U \to U$ is a* Δ-morphism *if it preserves indistinguishability:*

$$x \sim y \implies f(x) \sim f(y)$$

Theorem 4 (Category DD_0)**.** *Types (equivalence classes under $\sim$) and Δ-morphisms form a category.*

4 Types, Equality, and Functions

The key insight: these are not postulated but derived.

- **Type** = equivalence class $[x]_\sim$
- **Equality** = $x =_\Delta y \iff x \sim y$
- **Function** = Δ-morphism

No other structure is compatible with distinction.

5 Interpretation of STLC

5.1 Syntax

$$\begin{aligned} \text{Types:} \quad & A, B ::= \iota \mid A \Rightarrow B \\ \text{Contexts:} \quad & \Gamma ::= \varepsilon \mid \Gamma, A \\ \text{Terms:} \quad & t ::= x \mid \lambda x.t \mid t\ u \end{aligned}$$

5.2 Semantics

$$\begin{aligned} [\![\iota]\!] &= \text{BaseObj} \quad (U \text{ with } \sim) \\ [\![A \Rightarrow B]\!] &= [\![A]\!] \Rightarrow_{\text{obj}} [\![B]\!] \\ [\![\varepsilon]\!] &= \top_{\text{obj}} \\ [\![\Gamma, A]\!] &= [\![\Gamma]\!] \times_{\text{obj}} [\![A]\!] \end{aligned}$$

6 Soundness Theorem

Theorem 5 (Soundness)**.** *For every well-typed term $\Gamma \vdash t : A$, the interpretation $[\![t]\!]$ is a well-defined Δ-morphism:*

$$[\![t]\!] : \text{DDHom}([\![\Gamma]\!], [\![A]\!])$$

Proof outline. By induction on term structure:

- **Variables**: Lemma `lookup-resp` shows lookup preserves $\sim$.
- **Abstraction**: Lemma `curry` shows currying produces Δ-morphisms.
- **Application**: Lemma `appHom` shows application preserves $\sim$ via transitivity.

Full proof in Agda: `Distinction.Soundness`. □

7 Collapse Theorem

Theorem 6 (Collapse)**.** *If $D = \emptyset$, then:*

1. *$\forall x, y.\ x \sim y$ (total indistinguishability)*
2. *There exists exactly one type*
3. *All morphisms are equal*

Interpretation: Nontrivial mathematics requires $\Delta \neq \emptyset$. This is not a choice but a necessary condition.

8 Formal Verification

The complete formalization consists of 8 Agda modules (∼900 lines):

`Core.agda`	Primitive notions
`Axioms.agda`	A1, A2
`Types.agda`	Setoid structure
`Morphisms.agda`	Δ-morphisms
`Category.agda`	DD_0 laws
`Collapse.agda`	Theorem 9.1
`STLC.agda`	Syntax
`Soundness.agda`	Full soundness proof

Verification: `agda DDCalculus.agda` completes with exit code 0. No `postulate`, no `sorry`.

9 Discussion

What this does not claim

- We do not claim to have solved the foundations of mathematics.
- We do not claim this is the only possible foundation.
- We make no claims about physics or metaphysics.

Relation to dependent type theory

STLC is the minimal case. Extension to Π/Σ types via fibrations is future work.

Implications

Type theory is not a discretionary framework. It is a structure forced by the possibility of distinction. To reject it, one must deny that distinction exists—a position incompatible with formal reasoning itself.

Acknowledgments

[To be added]

References

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

[2] The Agda Team. *Agda Documentation*. `https://agda.readthedocs.io/`