THE ZERO PARADOX

ZP-J Keystone Addendum

The Diagonal Fixed Point: the Lawvere Face-Split and the Well-Foundedness Boundary

Initial: June 2026 | Current: 1.2, July 2026

The keystone of the Zero Paradox is that ⊥ is the same self-referential (diagonal) fixed point in every framework: the Quine atom $\perp = \{\perp\}$ in set theory, the Kleene quine in computation, the point $v_2(0) = \infty$ in valuation, and the initial object in category theory. That these faces are one object is a modeling commitment (MC-1), not a theorem. This addendum is a thin, honest record of two machine-checked investigations into the structure of that keystone — both probe-level, both fenced as to exactly what they prove.

The first asks whether the keystone is a literal instance of Lawvere's fixed-point theorem (the recognized home of the diagonal family). The answer is face-dependent and machine-checked. The second gives the keystone's structural home in well-founded coalgebra theory (Taylor; Adámek–Milius–Moss): the snap $\perp \to \varepsilon_0$ as a crossing of the well-foundedness boundary. Neither investigation is claimed beyond what Lean verifies; the deeper category-theoretic result is cited, not re-proved.

Section I: The Keystone and its Recognized Home

The unification of self-referential fixed points across fields is not new with the Zero Paradox. Lawvere (Diagonal Arguments and Cartesian Closed Categories, 1969) showed that Cantor's diagonal, Russell's paradox, Gödel's incompleteness lemma, and the recursion theorem are one move; Yanofsky (A Universal Approach to Self-Referential Paradoxes, Incompleteness and Fixed Points, Bull. Symbolic Logic 9(3), 2003) restated this in plain set-and-function terms across logic and computation. Those faces are prior art, cited and not claimed.

What the Zero Paradox adds (and what it does not)
Adds: candidate faces outside the classical scheme — the 2-adic valuation $v_2(0) = \infty$, ε_0, the wheel of fractions — each with a machine-checked axiom footprint; and the location claim, that the fixed point sits at the floor ⊥ (the Gödel inversion), a framing.
Does not add: the unification itself (Lawvere/Yanofsky), nor a proof that the four faces are numerically one object — that is the MC-1 commitment, offered to these communities, not imposed on them.

Section II: The Lawvere Face-Split (machine-checked)

Lawvere's theorem (Mathlib's Function.exists_fixed_point_of_surjective) says a point-surjection $\alpha \to (\alpha \to \beta)$ — the diagonal hypothesis — forces every endofunction of β to have a fixed point. The keystone's fixed points are posited (the self-application has $\perp$ as its fixed point); Lawvere's are derived from a surjection. So the question is whether each face supplies that surjection. The verdict is face-dependent.

In Set: no face is a Lawvere instance (Lawvere.lean)

For any nontrivial type, a Lawvere witness cannot exist: it would force every endofunction to have a fixed point, impossible once there are two distinct elements (Cantor).

nontrivial_lattice_no_witness — the lattice / set-theoretic face: no witness on a nontrivial $\perp$-pointed lattice.

q2_no_witness — the 2-adic face: no witness on $\mathbb{Q}_2$.

So in Set the keystone's $\perp$ is a posited fixed point of one specific self-map, not a Lawvere-derived one. The "diagonal fixed point" name is an analogy at this face.

fixedPoint_of_witness, no_witness_of_fixedPointFree: fully axiom-free.

In computability: a genuine instance (Lawvere.lean)

computability_face_fixedPoint — every computable self-map on codes has a fixed point. This wraps Mathlib's Nat.Partrec.Code.fixed_point (Rogers / Kleene's recursion theorem).

The escape from the Cantor obstruction is computability: the fixed-point-free diagonal is not a computable endomap, so the witness can exist in the effective category. This is ZP-K's face (the Kleene quine).

Footprint: [propext, Classical.choice, Quot.sound], inherited from Mathlib.

The verdict, plainly: the test is category-relative. In Set no face is a Lawvere instance; in the effective (computability) category the recursion theorem is a genuine one. The keystone therefore unifies a shape (the diagonal), not a single mechanism. The one-object identification remains the MC-1 commitment.

Section III: The Well-Foundedness Boundary

The keystone's structural home is well-founded coalgebra theory. A relation is encoded as a powerset coalgebra; well-founded means no infinite descending chain — no cycle. The Quine atom $\perp = \{\perp\}$ is exactly the minimal non-well-founded coalgebra: the self-loop $\perp \in \perp$, the back edge. Taylor (Well-founded coalgebras and recursion) proved that well-founded $\Rightarrow$ recursive, and (Prop 111) that well-foundedness is necessary for recursion: one cannot recurse through $\perp$. Adámek–Milius–Moss (On Well-Founded and Recursive Coalgebras, 2020, arXiv:1910.09401) give the modern characterizations (well-founded $\Leftrightarrow$ recursive $\Leftrightarrow$ a morphism to the initial algebra).

In that language the binary snap $\perp \to \varepsilon_0$ is a crossing of the well-foundedness boundary: from the non-well-founded floor ($\perp$, the self-loop, where recursion cannot reach) to the well-founded ascent (the ε_0 ordinal tower, recursively generated). The Zero Paradox formalizes this at two levels.

Relation level (Boundary.lean)

floor_not_wellFounded — the self-application floor is non-well-founded (⊥ self-loops under selfApp). Fully axiom-free.

ascent_wellFounded — the ordinal ascent is well-founded (Ordinal.lt_wf).

snap_crossing — on a single carrier, the floor is the sole non-accessible point; every post-snap state is accessible. The snap exits the unique non-well-founded point into the well-founded ascent.

Categorical bridge (BoundaryBridge.lean)

snap_boundary_two_registers — the crossing witnessed in two registers at once: the relation/carrier level above, and ZP-P's categorical μ/ν fork (categorical_fork_strict: the initial algebra empty, the final coalgebra inhabited — the self-referential element lives in ν, not μ).

Honest scope — what is proved, and what is committed

Proved: the relation-level boundary and the QPF μ/ν bridge above (floor_not_wellFounded is axiom-free; the rest carry Classical.choice from Mathlib's ordinal and QPF machinery).

No new commitment: that the snap is this crossing is a faithful model whose content is the two proven endpoints plus an identification — and that identification is the framework's existing $\perp/\varepsilon_0$ identification (MC-1, and the ε_0 identity already open under OQ-E2), not a fresh one. The floor endpoint is tied to the real ⊥ of ZP (floor_not_wellFounded, axiom-free); the single-carrier Phase is the illustrative toy model, where non-well-foundedness localizes at the floor by construction.

Section IV: What Is Not Formalized (and an Open Invitation)

The full Taylor coalgebraic statement — ⊥ as a non-well-founded coalgebra in the broken-pullback sense, with the General Recursion Theorem (well-founded ⇔ recursive) — is deliberately not formalized here. Mathlib currently lacks the required machinery: the next-time operator on subobject lattices, Pataraia's fixed-point theorem, and the recursion theorem for well-founded coalgebras. The depth result is therefore cited (Taylor; Adámek–Milius–Moss), not re-proved. What is given here is the relation-level boundary and the QPF bridge: a best effort that names its own boundary.

Open contribution point

Formalizing the missing machinery — the next-time operator, Pataraia's fixed-point theorem, and the General Recursion Theorem — would upgrade this best-effort bridge to the full coalgebraic statement, and would be a reusable Lean contribution independent of the Zero Paradox. Contributions are welcome, to this project and to Mathlib as a whole; the precise missing pieces are named above.

Endnote: This document is an addendum to ZP-J Self-Reference and reads after it. ZP-J established the porthole ($v_2(\perp) = \infty$, $\perp = \{\perp\}$); this addendum records two machine-checked probes into the keystone's structure — the Lawvere face-split and the well-foundedness boundary — each fenced as to exactly what it proves. Lean sources: Lawvere.lean, Boundary.lean, BoundaryBridge.lean, all sorry-free in Lean 4 as of June 2026.