THE ZERO PARADOX
ZP-E: Bridge Document

Initial: April 2026 | Current: 3.87, November 2026

This document is the cross-framework synthesis layer of the Zero Paradox. It imports from ZP-A (lattice algebra), ZP-B (p-adic topology), ZP-C (information theory), and ZP-D (Hilbert space state layer). It provides three formal inserts: DA-1 (Instantiation as Execution), DA-2 (Instantiation Succession), and DA-3 (Perspective-Relative Cardinality). AX-1 (Binary Snap Causality) is retired. Its content was split in two: the shape of the snap is proved, as Theorem T-SNAP, and that the snap occurs is stated separately: it follows from the occurrence commitment together with DA-1 (closed given DP-2). The occurrence commitment is stated in two charts, execution and slot, in the header of ZeroParadox/Order/Snap.lean (indexed in ZeroParadox/Order/SnapCannotBe.lean § II and CLAIMS.md), the slot chart as the hypothesis hw of t_snap_of_fires_once, that the tape fires at some stage, beside the theorem trials_fires_once that, for independent stages each firing with probability p, $0 < p \leq 1$, firing at some stage has probability one, which names no tape (ZeroParadox/Information/CrossingTrials.lean § VIII). tsnap_holds_but_nothing_moves shows T-SNAP does not carry that the snap occurs. With DA-2 and the commitment C-DA2, the directed instantiation tree is formally licensed. With DA-3, cardinality is shown to be position-dependent within the instantiation structure.

Illustrated Companion: A paired ZP-E Illustrated Companion document provides accessible explanations and visual summaries of the bridge derivations in this document. Readers new to the framework are encouraged to start with the companion.

A note on framework scope. The four layers (ZP-A through ZP-D) are not claimed to be the only mathematical domains in which the Zero Paradox structure appears. They are chosen because they cover the canonical mathematical languages for describing state: algebraic order structure (ZP-A), topology and metric geometry (ZP-B), information and complexity theory (ZP-C), and functional/Hilbert space analysis (ZP-D). What is claimed is that each of the four frameworks addressed here locates a structural limit at a floor of its own carrier, named with its carrier in the next paragraph and proved within that framework from that framework's own premises, design commitments among them (ZP-C's identification of c_0 with $\perp$ is its CC-2, and ZP-D rests on its design premise DP-1); reading the four as one limit is the framework's interpretation (MC-1, next paragraph), and these four are not claimed to be the only possible witnesses.

Each of the four layers finds a structural limit at a floor of its own carrier, and each names it in its own terms; ZP-A's floor is the least element of its order, and in Q_2 the norm value 0 is least on the image of the 2-adic norm in the reals (q2_norm_floor_isLeast), not as an order-bottom of Q_2 itself, and {0} is the intersection of the nested clopen balls (fB_bottom_is_limit): ZP-A identifies $\perp$, the bottom of the ZP-A semilattice, as the global minimum, below which the lattice's ordering relation cannot descend; ZP-B finds the 2-adic valuation undefined (or $+\infty$) at 0 in Q_2, where clopen separation is total — 0 and every nonzero element lie in disjoint clopen classes; ZP-C finds surprisal unbounded in the depth of its binary ball hierarchy, each value finite (L-INF), and has the initial configuration c_0 correspond to that hierarchy's limit point $0 \in Q_2$; from the unbounded surprisal and that correspondence together, the framework argues that no finite external description can contain c_0; ZP-D maps the state 0 of its state layer to a basis vector e_0

orthogonal to every other basis vector (dp1_orthogonality), so the snap is an orthogonal shift (t4_snap_orthogonal); orthogonality is symmetric and does not make the shift one-way, which comes from ZP-A and ZP-B (T-SNAP Step 7). Reading these as one structural limit seen in four mathematical languages is the framework's interpretation, the bottom family (MC-1): membership is proved domain by domain, and an identity across the layers is retired as ill-typed. ZP-C's correspondence places c_0 at the 0 of Q_2; ZP-M's snapEmbed (§ VI) is a different map, sending c_0 to 1 and c_1 to 0 in $\mathbb{Z}_2$, and the two are not composed here. DA-1's own content is narrower (§ IV): given DP-2's precondition, that the configuration reaching P_0 is a running machine's current configuration (Sense B, § II), the machine has moved from c_0 to c_1 by D7 (T-SNAP Step 3), and DP-2 and da1_minimal_path show that the two configurations are distinct while returning the same output value. That precondition is not derived: it is what the occurrence commitment asserts.

Formal Insert DA-1: Derived Proposition — Instantiation as Execution

DA-1 is a Derived Proposition: instantiation as execution, the first Lean formalization of DA-1 (axiom-free, conditional on DP-2, Snap.lean §VI), with incompressibility read as self-description (ZP-C D1 + AIT) and ZP-A CC-2/R3 as motivation for its precondition.

I. The Gap at T-BUF Step 2

The T-BUF chain from ZP-C states three results:

- L-RUN: The transition $c_0 \to c_1$ is a state change off $\bot$ of MachinePhase, which is c_0 (l_run: $c_0 \neq c_1$). (ZP-C — Derived)
- TQ-IH: $c_1 \neq c_0$, the same fact read from c_1 (tq_ih). With ZP-C D7, by which every executing program passes through c_1, it gives that no executing program outputs $\bot$ without a non-bottom intermediate configuration state; that program-level reading is D7's, not tq_ih's. (ZP-C — Derived)
- T-BUF: At P_0, the Binary Snap $\bot \to \varepsilon_0$ is a structural consequence of execution: an executing configuration is non-bottom, and that state is ε_0 in the semilattice. (ZP-C — Candidate Theorem; its Step 2, that the configuration at P_0 is executing, is the occurrence commitment, below, not a derived step)

T-BUF was labelled Candidate because Step 2 asserts that a configuration at P_0 is a live machine state — that instantiation at P_0 constitutes an execution event, not a static description. The premise is stated in two charts. In the depth chart, ZP-C L-INF: surprisal $I(n) = n$ is unbounded in the ball-hierarchy depth n while each value is finite (l_inf). In the point chart, at 0 of Q_2, the one point in every ball $B(0, 2^{-n})$ of the chain centred at 0 (fB_bottom_is_limit), the 2-adic additive valuation is $\top$ (padic_addVal_bot, and only there, padic_addVal_eq_top_iff); the coincidence of floor and $\top$ at one point is infinitude_forces_infinite_complexity, at the Q_2 instance q2InfinitudeFloor. $\bot$ of MachinePhase, c_0, corresponds to that point (ZP-C L-INF). Neither chart locates the unboundedness at P_0 (the valuation-complexity bridge is refuted, ZP-I § II.B). From this premise the framework argues that no finite external interpreter can hold $\bot$ as a static description. ZP-A CC-2 supplies a second, structural basis: $\bot$, the lattice's occupant of the Quine-atom role under a self-application map whose only fixed point is $\bot$ (selfApp_quine_role_iff; every ZPSemilattice carries one, trivialSelfApp), is a self-containing object with no external interpreter by structure. Neither derives T-BUF Step 2: it is DP-2's precondition, and it is what the occurrence commitment asserts (§ IV). DA-1 (§ IV below) consumes it and does not supply it: given it, D7

puts the machine at c_1 (T-SNAP Step 3), and DP-2 and da1_minimal_path show that c_0 and c_1 are distinct configurations returning the same output value.

II. The Two Senses of a Configuration at P_0

Sense A — Descriptive: x exists as a string — a finite syntactic object that has been written down or specified. The machine it describes has not necessarily been instantiated. P_0 is a property of the string. The string is inert.

Sense B — Instantiated: x exists as the current configuration of a running machine. The machine is executing. P_0 is a property of the live configuration. The configuration is active.

III. Design Principle DP-2 — Execution Distinguishability

The two senses of § II share the same output value (⊥) but differ in machine state. This separation — output value is not machine state — is made precise by Design Principle DP-2.

Design Principle DP-2 — Execution Distinguishability

Machine states carry execution history independently of output values. A machine in state c_1 can output ⊥ (the null value) while being in a configuration entirely distinct from a machine in state c_0. The post-execution null and the pre-execution null are different instances — same output value, different machine state.

Lean formalization (Snap.lean §VI): TrackedOutput separates value : MachinePhase from state : MachinePhase. preInstantiation = $\langle c_0, c_0 \rangle$; postInstantiation = $\langle c_0, c_1 \rangle$. Theorem dp2_execution_distinguishability proves: preInstantiation.value = postInstantiation.value (same output) ∧ preInstantiation.state ≠ postInstantiation.state (distinct machine states). Proved axiom-free.

Given DP-2, DA-1 is Lean-formalizable at the minimal-path level. Theorem da1_minimal_path (Snap.lean §VI) establishes four conjuncts: (1) before and after instantiation, the output value is the same (⊥); (2) the machine states are distinct ($c_0 \neq c_1$); (3) the machine was at c_0 before; (4) the machine is at c_1 after. Proved axiom-free — no Kolmogorov complexity, no ZF+AFA required. The "return to null" after instantiation is postInstantiation (output ⊥, state c_1) — not preInstantiation (output ⊥, state c_0). On the machine-state component, no join from c_1 returns to c_0 (t_snap_irreversible, which holds for the join of every ZP semilattice, MachinePhase among them, and is not about TrackedOutput records; ZP-B C3 is its topological counterpart).

IV. Proposition DA-1

Proposition DA-1 — Instantiation as Execution

Claim: Given its precondition (below), instantiation at the incompressibility threshold P_0 is an execution event in the sense of L-RUN: one act of instantiation moves the machine from c_0 to c_1, with the same output value before and after.

Scope of the claim: it holds given DP-2 and DP-2's precondition, that the configuration reaching P_0 is there in Sense B (§ II, a running machine's current configuration) and not in Sense A (an inert string). That precondition is not derived: it is what the occurrence commitment asserts, and Paths 1 to 3 below argue for it. What follows from it is $c_0 \to c_1$: by D7 a running machine has left c_0, and its first running configuration is c_1 (T-SNAP Step 3); DP-2 and da1_minimal_path (axiom-free) show that the two configurations are distinct while returning the same output value.

Proposition DA-1 — Instantiation as Execution

Formal core (Lean): DP-2 (§III) — TrackedOutput separates output value from machine state. da1_minimal_path proves that the pre-instantiation configuration (state c_0) and the post-instantiation configuration (state c_1) are distinct while returning the same output value, so the output value cannot show whether the step was taken. It takes no hypothesis and does not carry that the step is taken (ZeroParadox/Order/SnapCannotBe.lean § I). The returned ⊥ occupies the bottom role again; reading it as a NEW null rather than the prior c_0 is C-DA2, a modelling commitment. The axiom-free result is the role, not the novelty.

Structural motivation (Paths 1 and 2 below): the Quine atom (ZP-A CC-2): in an AFAStructure lattice ZP-J t_exec_iff proves ⊥ is the only Quine atom (IsQuineAtom), and so the only occupant of the Quine-atom role with the AFA-built host's membership as the step (quineAtomHost_quine_role_iff at afaStructure_isQuineAtomHost, [propext]); no equation makes ⊥ of a ZPSemilattice the AFA set Q = {Q}, and the commitment that remains concerns the host theory (R-AFA). ZP-A R3 reads off that a self-containing object admits no external interpreter position; that rules out an external executor, not an inert ⊥. Path 2 rests on two charts: ZP-C L-INF, surprisal unbounded in the ball-hierarchy depth with each value finite (l_inf), and the 2-adic additive valuation ⊤ at 0 of Q_2 (padic_addVal_bot), the point to which c_0, ⊥ of MachinePhase, corresponds. The framework argues from it that ⊥ exceeds the capacity of any finite interpreter, and the step from there to executing is a bridge principle of its own (Path 2), which the framework does not adopt and which is not the occurrence commitment. Neither derives the precondition.

The three paths below are not parallel proofs of DA-1. Each argues for DA-1's precondition, which is what the occurrence commitment asserts: that the configuration reaching P_0 is there in Sense B and not in Sense A, so that, by D7, the machine starting at c_0 (the bottom of MachinePhase) has fetched its first instruction. None derives it: Path 1 rules out an external executor, not an inert ⊥; Path 2 does not derive it: the step from unbounded surprisal to executing is a bridge principle of its own, which the framework does not adopt and which is not the occurrence commitment; and Path 3 rules out a shorter external generator, not an inert string. Given that precondition, the step $c_0 \to c_1$ is D7's (T-SNAP Step 3), and DP-2 + da1_minimal_path (axiom-free) show the two configurations distinct while returning the same output value.

Path 1 — Structural (Self-Execution, ZP-J T-EXEC): Nothing exists outside the null space. ⊥ is prior to all differentiation — there is no external state, no prior cause, no position from which something else could execute ⊥. If ⊥ executes at all, the only possible executor is ⊥ itself. A thing that executes itself contains itself, and the framework reads this as the Quine-atom role (ZP-A CC-2): a point that steps to itself and to nothing else. The argument is that this is forced by the impossibility of external execution, which makes it more than a free choice and less than a theorem. ZP-A R3 states the structural consequence: a self-containing object admits no external interpreter position. ZP-J T-EXEC proves that every Quine atom of an AFAStructure lattice is ⊥: IsQuineAtom(q) ↔ q = ⊥, proved axiom-free in Lean 4 (ZeroParadox.t_exec_iff). AFA (ZF + AFA) is a consistent set-theoretic home for this structure, and adopting AFA is CC-2's commitment; its Lean abstraction is a set of requirements on the host theory (QuineAtomHost) — that the host has a bottom that is its own only member, and that it is the only such set — which AFA meets, and that any such host gives up Foundation is forced (quineAtomHost_not_wellFounded). The Quine atom (ZP-A CC-2) is stated in the corpus's shared vocabulary of step relations (ZeroParadox/Die.lean, the die; each carrier with its own step is a chart) as the role Motion.SelfOnlyAt, occupied or provably empty chart by chart: under a self-application map supplied by AbstractSelfApp, ⊥ of the lattice is the only occupant (selfApp_quine_role_iff); in AFA set theory Q = {Q} is a theorem (Aczel 1988, Example 1.3); with membership as the step, a host's own bottom is the only occupant (quineAtomHost_quine_role_iff) and no well-founded set is one (zfset_quine_role_unoccupied).

No equation makes ⊥ of a ZPSemilattice the set Q, since it would be ill-typed; what stays a Forced Metatheoretic Commitment is the host theory (see Remark R-AFA).

Path 2 — Informational (ZP-C L-INF): The surprisal I(n) = n at ball-hierarchy depth n is unbounded — for any finite M, ∃ depth n with I(n) > M — while each value is finite (l_inf). c_0, ⊥ of MachinePhase, corresponds to the limit point $0 \in Q_2$, where the 2-adic additive valuation is ⊤ (padic_addVal_bot). From these two charts the framework argues that any finite external interpreter can hold only a finite informational bound, and no finite bound holds the whole depth chain whose limit point c_0 corresponds to: every bound M is exceeded at some depth n, though each I(n) is finite (l_inf); surprisal, a function of the depth, takes no value at the limit point itself. Note: Path 2 is motivational context, not a formal path to the conclusion. The step "exceeds every finite bound → therefore executing" is a bridge principle of its own, a missing principle and not a missing proof: the framework does not adopt it, and it is not the occurrence commitment, and it is not a premise of the Snap. It asks what it means for a mathematical structure to instantiate rather than merely satisfy conditions — a question no computability library answers. It is the implementation problem in the philosophy of computation, which owns the question: Putnam (Representation and Reality, 1988, appendix, pp. 120–125, as cited by Chalmers 1996) argued that every ordinary open system realizes every abstract finite automaton, and Chalmers (Synthese 108, 1996, 309–333) answered with an account of implementation that requires causal structure. Cited, not claimed. This document states the question in the state chart, where a configuration is one fact and its running is a further fact; ZP-K Section III.II states it in two charts, the history chart reading a complete description of a run as the run, and denies neither.

Path 3 — Computational: Incompressibility as Self-Description (ZP-C D1 + standard AIT; the two bounds used, and their sources, are at T-SNAP Step 1): The preceding paths argue that ⊥ admits no external interpreter. This path argues from incompressibility toward the positive claim (executing). In the standard Turing model (D7), a machine configuration x is either executing or not executing, and those two are exhaustive. (B) Live execution — x is the current configuration of a running machine. (A) Static description — x is a string specified but not being executed, and some external program p ($|p| < |x|$) generates x when run, so x is a description awaiting a separate execution event by an external generator. (A) and (B) are not exhaustive: (A) bundles not executing with having a shorter external generator. At P_0, read at the length n of c_1 as in Step 1, $K(c_1|n) \geq |c_1|$: c_1 is algorithmically incompressible, and the ratio $K(c_1|n)/|c_1|$ is 1 up to an additive constant in K (Step 1). No external program p exists with $|p| < |c_1|$ such that $U(p, n) = c_1$. State (A) requires such a p, so state (A) is eliminated by the Kolmogorov condition. That rules out a shorter external generator, not an inert string: an incompressible string written down and never run has no shorter generator and is not executing. So, for strings, being incompressible does not imply executing: that string is a counterexample. The converse, from executing to incompressible, is not claimed in this document. Path 3 therefore does not derive executing from incompressibility. What it bears on is one event, not configurations in general: that the configuration reaching P_0 is there in Sense B (§ II) and not in Sense A. That is the precondition DP-2 takes, and it is what the occurrence commitment asserts, not a further premise. DA-1 does not supply it; DA-1 consumes it: given it, the step $c_0 \to c_1$ is D7's (T-SNAP Step 3).

The formal grounding (DP-2, §III) and the three paths above operate at distinct levels, not as alternatives to one another. The conditional sits at the level of machine-state representation: if the instantiation at P_0 is a first instruction fetch in the sense of D7 (the configuration reaching P_0 is in Sense B), then $c_0 \to c_1$ follows by D7 itself, and DP-2 + da1_minimal_path (axiom-free, proved by construction) show the two configurations distinct while returning the same output value. The Lean statement takes no hypothesis; the conditional is in this reading of it. DP-2 is grounded in D7 itself (the standard computational distinction between

before-first-instruction and after-first-instruction states), which is prior to and independent of DA-1. The three paths above operate one level down: they argue for why the precondition holds — why the instantiation at P_0 is a first instruction fetch at all — and none derives it. Path 1 (structural) argues that nothing external to ⊥ can execute ⊥, so if ⊥ executes at all, the executor is ⊥ itself. That rules out an external executor, not an inert ⊥: in MachinePhase the bottom c_0 is a Quine atom (da1_closed_concrete), and the sequence that stays at c_0 is a state sequence (Premises of T-SNAP). The self-executor reading argues for the Quine atom as more than a free choice and less than a theorem (ZP-A CC-2, a Forced Metatheoretic Commitment about its host theory); ZP-J T-EXEC proves, axiom-free, that every Quine atom (IsQuineAtom) of an AFAStructure lattice is ⊥. Path 2 (informational) provides motivational context — unbounded surprisal as a pointer toward why static holding is incoherent — but the bridge from unbounded surprisal (L-INF) to execution is not a derived claim: it is a bridge principle of its own, which the framework does not adopt and which is not the occurrence commitment. Path 3 (AIT) argues that incompressibility eliminates a shorter external generator; that the configuration reaching P_0 is executing rather than inert is not derived from incompressibility: it is DP-2's precondition, what the occurrence commitment asserts. Path 1's Lean counterpart is da1_closed_concrete (ZP-K), which proves IsQuineAtom(⊥ : MachinePhase) and nothing computational; Path 3's Lean counterpart is the machinePhaseKleene instance's botCode_is_quine field, a KleeneStructure requirement that constant codes also meet, not a witness of execution and not a second independent proof. The two are carried together by da1_paths_unified as a conjunction; that they name one structural fact is the framework's reading. Path 2 identifies a missing principle. Path 2 is context.

Status: DERIVED PROPOSITION — primary formal grounding: DP-2 (§III, TrackedOutput construction). da1_minimal_path proved axiom-free in Lean (Snap.lean §VI): the pre- and post-instantiation configurations, at c_0 and c_1, are distinct while returning the same output value; it takes no hypothesis and does not carry that the step is taken. ✓ Path 1 (structural, ZP-J T-EXEC + ZP-K): rules out an external executor, not an inert ⊥. Its Lean counterpart is da1_closed_concrete : IsQuineAtom(⊥ : MachinePhase), proved in Kleene.lean, which proves nothing computational. (Under MachinePhase's selfMem x := x = ⊥, this reduces to (⊥ = ⊥) ∧ (∀ x, x = ⊥ ⇒ x = ⊥) — structural closure enforced by typeclass design, not a set-theoretic derivation from ZF+AFA. See R-K.0.) Path 2 (informational, L-INF): UNADOPTED BRIDGE PRINCIPLE — a missing principle, not a missing proof, distinct from the occurrence commitment and not a premise of the Snap; Path 2 does not derive the precondition. Path 3 (computational, ZP-C D1 + AIT): bears on the precondition and does not derive executing from incompressibility — incompressibility rules out a shorter external generator, not an inert string; that the configuration reaching P_0 is in Sense B and not Sense A is DP-2's precondition, what the occurrence commitment asserts, which DA-1 consumes and does not derive. Its Lean counterpart (ZP-K) is the machinePhaseKleene instance's botCode_is_quine field — a KleeneStructure requirement that constant codes also meet, not a witness of execution and not a second independent proof; da1_paths_unified carries it with Path 1 as a conjunction. CC-1 ($S_0 = ⊥$): restated — ZP-J cc1_derived with t_exec_iff makes a Quine-atom start and a ⊥ start the same condition (axiom-free, Lean) — and not forced: on a carrier with a second point a valid sequence starts elsewhere (ZeroParadox/Settheory/OntBridge.lean). The Quine atom (ZP-A CC-2): ZP-J t_exec_iff proves ⊥ is the only Quine atom (IsQuineAtom) of an AFAStructure lattice (axiom-free); with quineAtomHost_quine_role_iff at afaStructure_isQuineAtomHost ([propext]), it is therefore the only occupant of the Quine-atom role with that host's membership as the step; no equation makes ⊥ of a ZPSemilattice the AFA set Q = {Q}, since it would be ill-typed, and the role is occupied or provably empty chart by chart; what stays a Forced Metatheoretic Commitment is the host theory (Foundation-freeness forced, AFA adopted; see R-AFA). DP-2 (§III) — explicit. T-SNAP's premises, in its Lean form and in its prose argument, are stated under Premises of T-SNAP (DA-1 insert, § V). AIT (Kolmogorov complexity) outside Lean scope; the Kleene structure is its

in-scope counterpart, carried as a KleeneStructure requirement (botCode_is_quine), not a proof of the AIT claim.

V. Theorem T-SNAP — Binary Snap Causality

Theorem T-SNAP — Binary Snap Causality

"Causality" in this name refers to the shape of the step, not to its occurrence.

Statement: The Binary Snap $\bot \to \varepsilon_0$ is a derived consequence of P_0, L-RUN, TQ-IH, DA-1, and ZP-A D2, among the results the seven steps below cite. It is not an axiom. What it rests on differs between its Lean form and its prose argument: see Premises of T-SNAP, directly below.

Premises of T-SNAP, in two readings. Both readings are true, and neither replaces the other. In Lean, two of the premises, CC-1 and occurrence at the first transition, are the hypotheses of t_snap_given, named in (i).

- (i) Lean form: the shape. t_snap_derived : $c_0 \neq c_1 \wedge c_1 \neq c_0 \wedge \text{join } c_0\ c_1 = c_1$ (ZeroParadox/Order/Snap.lean) takes no hypotheses, and #print axioms reports no axioms. It is a fact about the fixed two-state type MachinePhase, with the initial phase c_0 as its bottom element $\bot$ and the running phase c_1 above it: the two phases differ, and joining them gives c_1. Nothing is passed to it as an argument; binary existence is built into the choice of that type, and the start at $\bot$ into the statement's choice of c_0. It fixes the SHAPE of the transition, not that one occurs: tsnap_holds_but_nothing_moves proves the same statement in a dynamics where every phase is a fixed point. Two of the premises as hypotheses: t_snap_given (same file) holds over any join-semilattice with bottom, taking $S_0 = \bot$ (hcc1, CC-1) and $S_1 \neq S_0$ (hocc, occurrence at the first transition: the step at index 1 is taken) and giving the same shape at S_0, S_1. Its two inequalities restate hocc; only the join conjunct is derived, from A4. t_snap_derived's statement is its MachinePhase instance at the sequence $c_0, c_1, c_1, \ldots$, where hcc1 holds by rfl and hocc reduces to $c_1 \neq c_0$. MachinePhase does not discharge occurrence: the sequence that stays at c_0 is a state sequence in the same type and fails hocc. With no dynamics given, dropping either hypothesis makes the conclusion fail (examples beside it). hocc forces at least two points and does not force exactly two; AX-B1's discreteness is not in that signature.
- (ii) Prose argument: the shape, then the occurrence. The seven steps below read that shape as a statement about the framework's own state sequence. That reading uses commitments, among them AX-B1 (binary existence, ZP-B, the framework's one substantive modelling commitment) at Step 4, and CC-1 ($S_0 = \bot$, a ZP-A Conditional Claim) for the start at $\bot$ in Steps 4 and 6. That the transition OCCURS rests further on the occurrence commitment, in its execution chart: instantiation occurs, in Sense B (§ II), so the configuration reaching P_0 is a running machine's current configuration and not an inert string. That is Step 2, and it is DP-2's precondition. DA-1 does not supply Step 2; it consumes it. DA-1 is closed only given DP-2: once DP-2's precondition is granted, $c_0 \to c_1$ is D7's (Step 3), and DP-2 with da1_minimal_path shows the two configurations distinct while returning the same output value; § IV's three paths argue for that precondition, none deriving it; Path 1 argues through ZP-A CC-2, itself a commitment. The occurrence commitment and DA-1 together give that the Snap occurs, which is not a theorem. Given CC-1, its first-step form is the hypothesis hocc of t_snap_given (a sequence that first moves at a later index fails hocc). ZP-C L-INF supplies unbounded surprisal; the step from there to executing is a bridge principle of its own (§ IV, Path 2), which the framework does not adopt, which is not the occurrence commitment, and which is not a premise of the Snap. ZP-C D1 makes c_1 incompressible, which rules out a shorter external generator and does not make the configuration

reaching P_0 executing rather than inert (§ IV, Path 3).

Proof:

Note on Lean scope: the formal Lean proof of T-SNAP is a single term — ⟨l_run, tq_ih, rfl⟩ — l_run proves $c_0 \neq c_1$ by decide, tq_ih is its symmetric form, and rfl closes the join in the MachinePhase semilattice (t_snap_given obtains that conjunct from A4, bot_join). What follows is the informal motivation for the modelling choices that let that term be read as a statement about the framework's own state sequence.

- Step 1 — P_0 identifies the incompressibility threshold. ZP-C § I takes x to be a binary string of length n, so $n = |x|$ and $K(x|n)/n = K(x|n)/|x|$. ZP-C D1 defines P_0 as the limit of $K(x|n)/n$ as n grows, the algorithmic entropy rate of x, and the threshold is that rate at its maximum, 1. The steps below use the finite-string form ZP-C also uses (ZP-C Remark R-BRIDGE and T-BUF Step 1), not the limit: one configuration string x of length n with $K(x|n) \geq |x|$, so x is algorithmically incompressible at its length. Since $K(x|n) \leq |x| + c$ for every x, with c depending only on the universal machine U, the ratio $K(x|n)/|x|$ is then 1 up to an additive constant in K, not necessarily exactly 1. K here is plain conditional complexity, which the sources below write C. That some x of each length n has $K(x|n) \geq n$ is a counting argument (Buhrman, Jiang, Li and Vitányi 1998, arXiv:cs/9809060, § 3); the upper bound $K(x) \leq |x| + O(1)$ is Shen 2015 (arXiv:1504.04955, Theorem 2(a)), and it holds given n by the same paper's Theorem 10(a), $C(x|y) \leq C(x) + O(1)$. (ZP-C § I, D1)
- Step 2 — The configuration reaching P_0 is in Sense B, not Sense A (the occurrence commitment): At P_0, $K(c_1|n) \geq |c_1|$ (Step 1 at $x = c_1$) — c_1 is incompressible, so no shorter external generator exists. That rules out a shorter external generator, not an inert string; that the configuration reaching P_0 is a running machine's current configuration (Sense B) and not an inert string (Sense A) is what the occurrence commitment asserts, not a derived step. Arguments for it, none deriving it: the Quine atom (ZP-A CC-2/R3); unbounded surprisal (ZP-C L-INF); DA-1 Path 3. (The occurrence commitment, whose content is DP-2's precondition. DA-1, closed given DP-2, consumes this step and does not supply it)
- Step 3 — Any instantiated execution passes through c_1. (ZP-C D7 — definitional; c_1 is the first running configuration)
- Step 4 — $c_1 \neq \bot$. (ZP-C L-RUN — Derived; c_1 has gained execution context not present in $c_0 = \bot$; by AX-B1 this is a distinct, non-bottom state)
- Step 5 — No program that executes produces only ⊥ configuration states. (ZP-C TQ-IH, $c_1 \neq c_0$, read with Step 3's D7: the execution trace $\tau(p)$ contains c_1 for any executing program p)
- Step 6 — In $(L, \vee, \bot)$, c_1 is an element strictly above ⊥. By ZP-A D2, the transition $\bot \to c_1$ is a valid state transition: $c_1 = \bot \vee \varepsilon_0$ for some $\varepsilon_0 \in L$ with $\varepsilon_0 > \bot$. This transition is the Binary Snap.
- Step 7 — The transition is irreversible: algebraically, no join from c_1 returns to ⊥ (t_snap_irreversible, the no-return form of ZP-A R1; that the signature has no subtraction operator makes an inverse inexpressible, which is a different fact); topologically by ZP-B C3 (no continuous return path to 0 in Q_2). These two grounds are sufficient. Conceptual correspondence: ZP-G AX-G2 ($\hom(X, 0) = \emptyset$ for $X \neq 0$) expresses the same irreversibility in categorical language — ZP-G is downstream of ZP-E and is not a formal dependency of this proof.

Conclusion: The Binary Snap is a derived consequence, given the commitments under Premises of T-SNAP. AX-1 (Binary Snap Causality) is retired. Its content was split in two: the shape of the snap is proved, as Theorem T-SNAP, and that the snap occurs is stated separately: it follows from the occurrence commitment together with DA-1 (closed given DP-2). tsnap_holds_but_nothing_moves shows T-SNAP does not carry it.

✓

Status: DERIVED — Cross-Framework. Premises: in two readings, under Premises of T-SNAP above; the Lean form fixes the shape with no hypotheses, t_snap_given takes CC-1 and occurrence at the first transition as hypotheses, and the prose argument's shape and its occurrence rest on commitments, among them those named there. Dependencies cited in the steps: ZP-C D1, D7, L-RUN, TQ-IH; ZP-B C3; ZP-A D2, R1; ZP-E DA-1; ZP-J T-EXEC. CC-1 ($S_0 = \bot$): restated — a Quine-atom start and a $\bot$ start are one condition (ZP-J cc1_derived, t_exec_iff; axiom-free) — and not forced, since on a carrier with a second point a valid sequence starts elsewhere. The Quine atom (ZP-A CC-2): ZP-J t_exec_iff proves $\bot$ is the only Quine atom (IsQuineAtom) of an AFAStructure lattice (axiom-free); no equation makes $\bot$ of a ZPSemilattice the AFA set $Q = \{Q\}$, since it would be ill-typed (see R-AFA). Both remain commitments: CC-1 is a Conditional Claim, expressed through the Quine-atom role and not removed by it; ZP-A CC-2 is a Forced Metatheoretic Commitment about the host theory. Conceptual correspondence only: ZP-G AX-G2 (downstream of ZP-E; not a formal dependency).

Remark R-ε_0 — On the Symbol Choice for the Snap Displacement

Note: this remark draws an informal structural analogy. No formal embedding of ZP's ε_0 into the ordinal ε_0 is claimed or established here. The symbol ε_0 in Step 6 names the state the snap reaches from $\bot$ ($\varepsilon_0 = c_1$, by A4); which of its properties are theorems, which the carrier supplies, and which remain commitments is recorded in ZeroParadox/Order/SnapCannotBe.lean. This symbol is chosen deliberately to coincide with ε_0, the proof-theoretic ordinal of Peano Arithmetic.

The ordinal ε_0 (Cantor 1897, Math. Ann. 49, §20, Satz B). In ordinal arithmetic, ε_0 is the smallest fixed point of the map $\alpha \to \omega^\alpha$: equivalently, $\varepsilon_0 = \sup\{\omega, \omega^\omega, \omega^{\omega^\omega}, \ldots\}$. No finite ω-tower reaches it. Gentzen (1936, 1938), in the form stated by Rathjen (arXiv:1405.4484, Theorem 2.8(i), p. 10): primitive recursive arithmetic, PRA, proves the implication from PRWO(ε_0) (that no primitive recursive sequence of ordinals below ε_0 descends forever) to Con(PA). Necessity — that no smaller ordinal would do — is the other direction, and it rests on different ground: the provability in PA of transfinite induction at every ordinal strictly below ε_0, which Gentzen reports as already known, crediting Hilbert-Bernays (Gentzen 1943, Math. Ann. 119, p. 140 and its footnote 3), together with Gödel's second incompleteness theorem, by which PA cannot prove Con(PA) from within. Gentzen gave a Gödel-based argument of this kind himself, stated with hedges and for his own assignment of ordinals to derivations (Gentzen 1938, "Neue Fassung des Widerspruchsfreiheitsbeweises für die reine Zahlentheorie," Forschungen zur Logik und zur Grundlegung der exakten Wissenschaften, neue Folge, Heft 4, 19–44, § 4.4; English translation "New version of the consistency proof for elementary number theory," in M. E. Szabo (ed.), The Collected Papers of Gerhard Gentzen, North-Holland, 1969, 252–286, here pp. 284–285). The ordinal ε_0 is therefore the least ordinal at which the transfinite induction PA can prove stops.

The structural analogy. ZP's ε_0 occupies an analogous position in the state lattice. At P_0, c_1 satisfies $K(c_1|n) \geq |c_1|$, the finite-string form of T-SNAP Step 1: it is algorithmically incompressible. Just as the Cantor ε_0 cannot be reached from 0 by any finite ω-tower, ZP's ε_0 is reached from $\bot$ by the snap, and no program shorter than c_1 outputs c_1 given its length n. Both name a structurally analogous object: a witness for a transition that exhausts the finite generative hierarchy below it.

Remark R-ε_0 — On the Symbol Choice for the Snap Displacement

The proof structures are analogous. The proof-theoretic analysis locates the minimum ordinal up to which PA cannot prove transfinite induction. ZP locates a state displacement at which no external program can hold ⊥ as a static string — where unbounded surprisal (ZP-C L-INF) and self-containment (ZP-A CC-2, itself a commitment) together rule out an external interpreter position, which rules out an external executor, not an inert ⊥. In both cases the exhaustion of finite description is a structural feature, not a deficiency — and it does not make the snap mandatory (ZeroParadox/Order/SnapCannotBe.lean § I). That the system actually executes at ε_0 rests on the occurrence commitment and DA-1, under Premises of T-SNAP: a selection of where the step is taken, not a necessity.

What is not claimed. ZP does not assert that L is an ordinal structure, or that ZP's ε_0 is literally the Cantor ordinal under a formal embedding into the p-adic/lattice framework. The analogy is motivational: both ε_0s mark a witness for incompressibility relative to a finite base. A formal embedding — showing that the Cantor ε_0 is order-isomorphic to or embeds into the p-adic completion of L at ⊥ — remains an open question and would constitute a strengthening of this claim.

Remark R-AFA — Why Foundation is Ruled Out; Non-Well-Foundedness Forced, AFA Adopted

The Quine atom (ZP-A CC-2) is a role, a point that steps to itself and to nothing else, and no equation identifies ⊥ of a ZPSemilattice with a set Q = {Q}: it would be ill-typed (Path 1 above lists the occupancy theorems chart by chart). What is a set-theoretic step beyond ZP-A's algebra, not derived from A1–A4, is the choice of a host theory in which a set occupies the role under membership. That choice is nonetheless not arbitrary, and its two halves separate. Occupying the role under membership forces a non-well-founded host, which is what rules ZF + Foundation out. Which non-well-founded axiom to adopt is a further step that the forcing does not take: ZF + AFA is the host this framework adopts, as the canonical theory meeting the requirements, not the one theory the bottom selects.

Foundation leaves the role unoccupied (structural). With membership as the step, a set occupying the Quine-atom role is its own only member, x = {x} (zfset_selfOnlyAt_mem_iff), so it is a member of itself, x ∈ x. ZF + Foundation (the Axiom of Regularity) forbids exactly this — under a well-founded membership relation no set lies on a membership cycle of any length, so in particular no set is a member of itself. This is a theorem of ZF + Foundation, not a framework-specific stipulation; it is machine-checked here as no_quine_atom / no_membership_cycle (ZeroParadox/Settheory/Wall.lean, footprint [propext, Quot.sound], choice-free), and in the die as zfset_quine_role_unoccupied. With membership as the step, a Foundation-respecting universe therefore has no occupant of the role at all (under any well-founded membership: the well-founded example in ZeroParadox/Settheory/QuineDichotomy.lean, from wf_no_selfloop); under a self-application step the role is occupied in every ZPSemilattice, Foundation or not (selfApp_quine_role_iff, with trivialSelfApp supplying the map), so the exclusion concerns the membership chart.

The framework's readings of ⊥ support the premise that the role is occupied under membership; the exclusion itself is a theorem. R3 (ZP-A) — an occupant of the Quine-atom role has no describer position external to itself — and L-INF (ZP-C) — surprisal I(n) = n is unbounded in the ball-hierarchy depth (l_inf), and DA-1 Path 2 reads this as no finite interpreter holding ⊥, a reading l_inf does not derive — each describe an object with no external vantage and unbounded self-referential depth. That is exactly the profile of a non-well-founded, self-membered set: the two readings motivate why the role must be occupied under membership, which Foundation excludes, rather than serving as independent proofs of the exclusion.

Remark R-AFA — Why Foundation is Ruled Out; Non-Well-Foundedness Forced, AFA Adopted

Non-well-foundedness is forced; AFA is the canonical replacement. An occupant under membership lies on a membership loop ($x \in x \in x \in \ldots$), so the role can be occupied only in a non-well-founded metatheory (wf_no_selfloop; for a host, quineAtomHost_not_wellFounded, below). That much is forced; the particular axiom is not. Aczel records four anti-foundation axioms, and AFA, FAFA and SAFA each admit exactly one Quine atom — his own term is a reflexive set — while being pairwise incompatible (Aczel 1988, Corollary 4.28; Appendix A, p. 108). Supplying exactly one therefore does not single AFA out, so AFA is adopted as the canonical witness rather than selected by the membership structure.

What is machine-checked (the requirements framework). The question of which host theory can admit an occupant of the role is reduced to two requirements: (Y′) a Quine atom, a host bottom $b = \{b\}$, and (Z′) its uniqueness among reflexive sets, $x = \{x\}$, the fields bot_mem_iff and reflexive_unique of the typeclass QuineAtomHost (ZeroParadox/Settheory/QuineHost.lean). AFA allows at most one solution of $x = \{x\}$ (Aczel 1988, p. 19), and decorates $a^* = \{a, a^*\}$ for every set a (Examples 1.5–1.6, pp. 7–8). In Lean: quineAtomHost_not_wellFounded (axiom-free) — the membership of a QuineAtomHost is not well-founded, so (X) Foundation-freeness is forced, not assumed; zfSet_no_singleton_bottom ([propext, Quot.sound], choice-free) — no ZFSet is its own sole member, the one exclusion proved about the real theory. The AFA and Boffa sorts are cited facts about those theories, each illustrated by a finite toy (QuineHost.lean § VII; boffa_fails_reflexive_unique). Realizability carries no information: every inhabited carrier meets QuineAtomHost (trivialQuineAtomHost). The typeclass QuineHost (fields bot_selfMem, $b \in b$ for its bottom b, and selfMem_unique, no other element self-membered) is a different requirement, incomparable with QuineAtomHost (QuineHost.lean § VII); AFA's membership fails its second field, since $0^* \in 0^*$. What stays argued is which requirements to demand, below.

Boffa is set aside on a weaker warrant, and the cited half and the checked half are separate. That Boffa's anti-foundation axiom admits a proper class of Quine atoms rather than one is a cited fact about that theory, and it reaches that axiom through a weaker one, in a direction worth stating. Boffa's axiom is equivalent to his earlier axiom BA_1 together with an extension condition, so it implies BA_1 (Aczel 1988, Exercise 5.14(i) for the equivalence; p. 59, where the axiom is formulated as Boffa's strengthening of BA_1); and BA_1 on its own already forces the reflexive sets — Aczel's term for Quine atoms — to form a proper class (Aczel 1988, Proposition 5.1, p. 57). That permissiveness is therefore inherited from the weaker axiom rather than introduced by the strengthening. This development formalizes no part of either Boffa axiom. What is machine-checked is the mechanism behind it — that (Y′) does not force (Z′): in a two-element toy model where every element is its own only member, no element is the reflexive one, so no QuineAtomHost has that membership (boffa_fails_reflexive_unique, ZeroParadox/Settheory/QuineHost.lean, axiom-free). That is a counter-model separating the two requirements, and it blocks one direction only — a host supplying a reflexive element need not supply its uniqueness. Dropping uniqueness accordingly leaves a host theory no less consistent; what it leaves standing is the Quine-atom clause alone, and in that toy model every element satisfies that clause, so the surviving clause is uninformative as a pinning condition — every element is a witness, and none is pinned as the host's distinguished bottom. Neither half is an in-kernel result about Boffa's axiom itself.

Remark R-AFA — Why Foundation is Ruled Out; Non-Well-Foundedness Forced, AFA Adopted

What remains — a narrow Forced Metatheoretic Commitment. With the forcing proved, one step is not a theorem: that a Quine atom and its uniqueness are the right two requirements to demand — the commitment to a host bottom $b = \{b\}$ as the minimal option (it has exactly one member, itself, and introduces no internal differentiation; any extension $b = \{b, x\}$ with $x \neq b$ would add members carrying their own membership chains). This is a Forced Metatheoretic Commitment about the host theory — stronger than a free choice, weaker than a theorem — not a Conditional Claim. It is not an equation between $\bot$ of a ZPSemilattice and a set, which would be ill-typed: the Quine-atom role's occupancy is machine-checked chart by chart (Path 1 above), and with the membership of the host built from an AFAStructure lattice as the step its only occupant is $\bot$ (quineAtomHost_quine_role_iff at afaStructure_isQuineAtomHost; in the IsQuineAtom form, t_exec, ZeroParadox/Settheory/SetTheoryAFA.lean, axiom-free, given the field bot_self_mem). The named falsifier is correspondingly narrow: a differently-motivated requirement set that hosts the role without demanding a unique Quine atom would reopen this choice — it would not touch the forcing, which is proved.

Reading (the die, not a result): AFA is one selection that meets the minimum requirements for the host's bottom to be the only occupant of the die's Quine-atom role: (Y') and (Z') of QuineAtomHost, which are exactly that condition (quineAtomHost_quine_role_iff and the example after it, ZeroParadox/Settheory/QuineDichotomy.lean); that AFA meets them is cited, not checked.

VI. Effect of T-SNAP on Downstream Results

Remark R-DA1: AX-1 is retired, and its content was split in two. Results in ZP-E that previously depended on AX-1 as an axiom now depend on T-SNAP, a derived theorem, for the shape, and, wherever they use that the Snap happens, on the occurrence commitment together with DA-1. T5 (Iterative Forcing Theorem) depended on AX-1 and is restated below. T4 (Unified Snap Description) carried AX-1 as an axiom label on the causality component — that label is upgraded to Derived — T-SNAP for the shape, with occurrence still a commitment. DA-1 is additionally upgraded from Design Principle to Derived Proposition, closed given DP-2 (da1_minimal_path); ZP-A CC-2 (the Quine atom), R3 and ZP-C L-INF argue for its precondition, which is what the occurrence commitment asserts, and none derives it. The intentional axioms of the system are now: AX-B1 (binary existence), AX-G1 (initial object), AX-G2 (source asymmetry). AX-1 is retired: its shape is Theorem T-SNAP, and the occurrence commitment is not on this list because it is a commitment, not a named axiom. Additional structural commitments are carried as typeclass fields: A4 (bot_join, ZPSemilattice — standard semilattice algebra), AFAStructure.quine_unique and bot_self_mem (the Quine-atom role), KleeneStructure.botCode_is_quine (the bottom code is a computational quine, which a constant code meets). Like named axioms these are assumptions, carried as hypotheses by every theorem that uses them; #print axioms does not surface typeclass fields. Unlike named axioms they can be cheap to meet: every ZP-A lattice supplies AFAStructure trivially, and KleeneStructure's botCode_is_quine is met by a constant code (the computable KleeneStructure MachinePhase example in ZeroParadox/Computability/Kleene.lean).

T5 (Iterative Forcing Theorem) — restated

T5 (restated). Selection. In the ordinal chart the next rung is the snap re-seeded one step past the current one (succession_succ), and none lies between two consecutive rungs (ZeroParadox/Ordinal/SnapSuccession.lean § I). Carried into the two-state lattice by a monotone map φ that sends the tower's stages to c_0 and ε_0 to c_1, no ordinal below ε_0 is sent to c_1, and every fixed point of $\alpha \mapsto \omega^\alpha$ is (snap_unconditional, hfp_from_epsilon_zero). The two faces of ε_0 carry the two directions: as the supremum of the stages, every ordinal below ε_0 lies under one of them, so nothing below fires (fundamentalSeq_cofinal); as the least fixed point of $\alpha \mapsto \omega^\alpha$ it lies below every other, so every fixed point of $\alpha \mapsto \omega^\alpha$ fires (hfp_from_epsilon_zero, epsilon0_min_eq_max). The placement at ε_0 enters there as the alignment hypothesis $h\varepsilon_0$ (CLAIMS.md OQ-A1): $\varphi\,\varepsilon_0 = c_1$; in the $\mathbb{Z}_2$ chart, snapEmbed $(\varphi\,\varepsilon_0) = 0$, where snapEmbed is ZP-M's map sending c_0 to 1 and c_1 to 0. It is the snap's occurrence at ε_0, taken as a hypothesis: of the infinitely many admissible firing points that monotonicity and tower alignment (every tower stage sent to c_0) leave open, $h\varepsilon_0$ selects the least, and so fixes φ uniquely (ZeroParadox/Ordinal/Incompleteness.lean § II). Whether Classical.choice is forced by the metric collapse is a separate open question (ZeroParadox/Ordinal/SyntacticCollapse.md). The rungs are the iterative bottoms; none of them is $\bot$ (epsilon0_ne_bot; for every rung, Ordinal.epsilon_pos). On an arbitrary lattice no step is selected: 0, 2, 4, … on the natural numbers satisfies A1–A4 and AX-B1.

Iteration. No step is guaranteed. A state sequence moves only upward (T3) and never returns (R1, t_snap_irreversible). Where the next increment is already absorbed ($S_n \vee \alpha_n = S_n$), the state stays the same, which T-SNAP permits (tsnap_holds_but_nothing_moves); different runs need not reach the same height. If from some step x on every new increment is already absorbed, then S_x is the supremum of the run: the value that run reaches in the limit. Short of a top, reaching it cannot be confirmed at any finite step; a run that reaches a top state (c_1 on the two-state carrier) is known there to stay.

Reading: ε_0 is both at once for the ordinal tower: the least fixed point of $\alpha \mapsto \omega^\alpha$ and the supremum of its stages (epsilon0_min_eq_max).

The handle T5 is kept. "Forcing" names the shape of each step taken, not that any step is taken.

Formal Insert DA-2: Instantiation Succession — The Multiple-⊥ Result

Formally licenses the directed instantiation tree, together with the commitment C-DA2

I. The Gap DA-2 Addresses

Given the premises stated under Premises of T-SNAP (DA-1 insert, § V), DA-1 and T-SNAP give the Binary Snap on reaching P_0 within an instantiation. After DA-1 the CC-1 and terminal-state questions remain open; ZP-B R1's scope and the occurrence fence stand beside them:

- CC-1 (ZP-A) posits $S_0 = \bot$ — not derivable from A1–A4 alone (ZP-J restates it as an equivalence with starting at a Quine atom, cc1_derived and t_exec_iff, and does not force it). This leaves open whether $\bot$ is unique across all instantiations or whether each instantiation carries its own $\bot$.
- Whether T-SNAP's shape applies at the terminal state of instantiation I_n, read as the starting $\bot$ of I_{n+1}. That needs the terminal state connected to a $\bot$ role-occupant. DA-2 provides the role half only: what plays the $\bot$ role of a ZP-A semilattice is that $\bot$ (da2_bottom_characterization). Reading it as a new $\bot$ is C-DA2, a commitment. Nothing here places that terminal state at P_0: the valuation-complexity bridge offered for it is refuted (ZP-I § II.B).

• ZP-B R1 (Universal Structure vs. Instantiation-Contingent Parameters) keeps the structural role of ε_0 of Q_2 universal; what is contingent per instantiation is the rung v of ZP-B D5 and its threshold 2^{-v}. R1 keeps AX-B1, whose 0 is ⊥ of OntologicalStates, a ZP-A semilattice, universal as structure.

• Occurrence fence. T-SNAP fixes the SHAPE of each step. It does not establish that any step is taken: tsnap_holds_but_nothing_moves exhibits a model in which T-SNAP holds and nothing moves. Every "fires" below narrates the occurrence commitment, not a consequence of the theorem.

II. Why ZP-B C3 is Not Violated

C3 prohibits a continuous path from $x \neq 0$ back to the same 0 in Q_2. DA-2 does not require a return path. The irreversibility of C3 is preserved within each instantiation. What crosses the instantiation boundary is not a path in Q_2 — it is the generation of a new Q_2 with its own metric, its own ⊥, its own ε_0. C3 quantifies only over paths within a single topological space and has nothing to say about the boundary between spaces.

More precisely: C3 rules out a continuous return to 0 WITHIN one Q_2. It does not follow that a recurrence of ⊥ must be a structurally distinct ⊥ — that step needs the separateness of the spaces, which is a modelling commitment (C-DA2) and not something the topology supplies. The topology does not enforce novelty; it is silent on it, and in the 2-adic chart the arc reapproaches the same 0 (snap_arc_z2_loop).

III. Definitional Alignment DA-2 — Instantiation Succession

Definitional Alignment DA-2 — Instantiation Succession
Claim: A state S in instantiation I_n satisfies the structural role of ⊥ for instantiation I_{n+1} if and only if it satisfies A4 relative to all subsequent joins in I_{n+1}:
$S \vee x = x$ for all x in the semilattice of I_{n+1}.

Grounding: A4 is the load-bearing axiom of ZP-A — it defines ⊥ as the additive identity under ∨, the element that contributes nothing to any join and is therefore present in everything above it. DA-2 does not redefine ⊥. It clarifies that the CC-1 identity condition ($S_0 = \perp$) can be satisfied by any state meeting A4′s algebraic conditions — not only by a cosmologically primitive ⊥. The identity condition is structural, not historical: what matters is the algebraic role a state plays in the subsequent semilattice, not where it came from.

On the framework's reading, a commitment and not a theorem, the terminal state of I_n arrives at I_{n+1} carrying the accumulated join of everything in I_n's sequence. It is structurally ⊥ to I_{n+1} — contributing nothing to subsequent joins — while being informationally rich relative to I_n. This is the Zero Paradox instantiated at the inter-instantiation level: the terminal state is simultaneously a terminus and a foundation.

Status: DEFINITIONAL ALIGNMENT — no new axiom introduced. DA-2 is a clarification of the scope of CC-1 and A4. ✓

IV. Conditional Claim C-DA2 — Novelty of Successive ⊥

Conditional Claim C-DA2 — Novelty of Successive ⊥ (a modelling commitment)

Commitment: each instantiation carries its OWN topological space, so the ⊥ of I_{n+1} is not an element of the Q_2 of I_n. GIVEN that, no two instantiations share a ⊥, and succession is a chain (or tree) rather than a cycle.

Status: CONDITIONAL, not derived. The commitment above is the separateness of the spaces; it is assumed, not obtained from C3. Formerly labelled a Corollary and marked derived — retracted, see the note below. The handle C-DA2 is unchanged.

What IS proved: that anything occupying the bottom role IS the bottom of its own lattice (t_iz_limit_is_new_null). That is an identity with the bottom already present, not the production of a second one.

Why this is NOT a proof, stated plainly because it was published as one. The step "the ⊥ of I_{n+1} is an element of a distinct topological space" is the CONCLUSION, not a consequence of C3. C3 quantifies over paths WITHIN a single space — it says no continuous path runs from any $x \neq 0$ back to 0 inside one Q_2 — and has nothing to say about whether the next instantiation's space is a different one. Assuming separateness and then deriving distinctness from it is circular. DA-2 likewise fixes identity conditions WITHIN an instantiation; it does not compare across two.

And the corpus measures against it where the question is concrete: in the 2-adic realization the arc returns to the SAME zero. snap_arc_z2_loop proves the tower's 2-adic image starts at 0 (stage 0), that stage n is $\neq 0$ for every $n \geq 1$, and that it converges back to 0; tower_image_loops_to_seed states the limit IS the seed value. So in that chart novelty does not merely lack a proof — it fails. C-DA2 stands as a modelling commitment about instantiations, and must not be cited as derived, nor supported by T-IZ or by t_iz_limit_is_new_null.

V. The Directed Instantiation Tree

With DA-2 and C-DA2 in place, the global structure of instantiations is a forward-directed tree with no back edges:

- Each node in the tree is a ⊥ — the bottom element of one instantiation and the foundation of all successor instantiations branching from it.
- Each edge within an instantiation is a step in a monotone state sequence (ZP-A T3). Edges are irreversible (ZP-B C3).
- Branching at each node: every distinct outbound vector from the terminal state of I_n is a valid ε_0 for a distinct I_{n+1}. T-SNAP does not select among branches. Because ⊥, the only occupant of the Quine-atom role in the lattice under a self-application map (ZP-A CC-2; selfApp_quine_role_iff), has no internal differentiation, every ε_0 that represents a first differentiation in any direction is a valid outcome. The work here is done by the undifferentiated ⊥ of ZP-A CC-2, not by T-SNAP: given the occurrence commitment, no direction is privileged, so the structure branches rather than threading a line. T-SNAP fixes the shape of each step; it does not supply the occurrence.
- No back edges: C-DA2 COMMITS to no instantiation reaching the ⊥ of any ancestor. This is the commitment, not a theorem - in the 2-adic chart the arc returns to the same 0 (snap_arc_z2_loop).

Remark R-DA2: T-SNAP's algebraic core, $\bot \vee \varepsilon_0 = \varepsilon_0$, holds at ⊥ of any ZP-A semilattice (t_snap_join), so it needs no modification across instantiation boundaries. DA-2 supplies the role half only: what plays the ⊥ role of the semilattice of I_{n+1} is that ⊥ (da2_bottom_characterization). That the terminal state of I_n plays

that role is a commitment of its own (occupancy), not a theorem; that the return is to a NEW bottom is a further commitment (C-DA2); in the 2-adic realization the arc tends back to the same 0 of $\mathbb{Z}_2$ (snap_arc_z2_loop). Nothing here places that terminal state at P_0 (ZP-I § II.B). No new axiom is required. The multiverse is not the claim that T-SNAP fires in all directions simultaneously: it is the claim that ⊥, the lattice's occupant of the Quine-atom role under a self-application map (ZP-A CC-2), has no internal differentiation and therefore no preferred direction for ε_0. The multiverse of instantiations is the full set of all minimal differentiations available from the single self-containing ⊥ — a structural consequence of ZP-A CC-2 + T-SNAP + DA-2 jointly, given C-DA2.

VI. The Zero Paradox Iterated

The paradox of ⊥ — simultaneously contributing nothing and being present in everything — propagates structurally at every branching node of the tree. Each node is:

- Nothing to its successor instantiations: it acts as additive identity under ∨, contributing nothing to any subsequent join.
- Everything to its successor instantiations: every state in I_{n+1} satisfies $\perp_{n+1} \leq S$, so the node underlies everything that follows.

The tree is the geometric shape of the Zero Paradox iterated across instantiations. The single-instantiation linear sequence was always a cross-section of a structure with this shape.

The complete picture: The Zero Paradox describes a forward-directed infinite tree where $\perp_1 \rightarrow \ldots \rightarrow S_{\text{terminal}}^{1} \equiv \perp_2 \rightarrow \ldots$, where ≡ means structurally satisfies the role of, not is identical to. Each arrow within an instantiation is monotone and irreversible. Each ≡ crossing is not a path — it is a new instantiation of the universal structure.

VII. Implications Within the Framework

Structural Implication: Branching Tree Structure. T-SNAP fixes the SHAPE of the Binary Snap — that if ⊥ transitions it goes to some $\varepsilon_0 > \perp$, and that the step is one-way. It does not establish that the transition is taken: that the Snap occurs follows from the occurrence commitment together with DA-1 (closed given DP-2), and the NO-GO gauge tsnap_holds_but_nothing_moves exhibits a model in which T-SNAP holds and nothing moves. DA-2 supplies the role half only: a terminal state that plays the ⊥ role in a successor instantiation's semilattice is that ⊥. That it does is a commitment of its own (occupancy); that it is a NEW ⊥, so that the instantiations form a forward-directed branching tree, is the further commitment C-DA2. The branching comes from ZP-A CC-2's undifferentiated ⊥ and the succession from DA-2 with C-DA2, all conditional on instantiation occurring. Note: T-SNAP alone does not establish that it fires on all outbound vectors simultaneously — that universality is the scope of DA-2, not a direct consequence of the snap theorem itself.

Monotonicity and Path Irrecoverability. Within an instantiation, state sequences are monotone — no state can be decreased (ZP-A R1). Every state change is a join operation: $S_n \vee \alpha$ for some increment α. The algebra constrains only that the sequence be monotone, not which monotone path is taken. A join either leaves the state unchanged (when the increment lies at or below the state) or moves it strictly up, and no join returns it below (t_snap_irreversible). What the resulting position records is stated under Causal structure, below.

Ordinal Direction of State Sequences. What is defined is that every state change is a join (a state sequence, ZP-A); what is derived from that is the directional asymmetry: each state lies at or below its successor (ZP-A

T3, state_sequence_monotone) and no join returns a state below it (t_snap_irreversible). Irreversibility is also C3 applied within an instantiation, topologically.

Causal structure. The start and the history together determine the position: $S_{n+1} = S_n \vee \alpha_n$, and $\vee$ is an operation, so the start S_0 and the increments fix every state. The position does not determine the history. $\vee$ is associative, commutative and idempotent (ZP-A A1–A3). Associativity makes S_n the single join of S_0 with $\alpha_0, \ldots, \alpha_{n-1}$, and the position records only that join. Commutativity and idempotence make the order of the increments and how often each arrived unrecoverable from it, and the join does not determine which increments arrived (only that each lies at or below it). An increment already at or below the state leaves no trace (a step changes nothing exactly when its increment is at or below the state: the example after state_sequence_monotone, ZeroParadox/Order/Lattice.lean). Distinct histories reaching the same element are indistinguishable in L. No effect without the join that produced it.

Formal Insert DA-3: Perspective-Relative Cardinality

Cardinality as position-dependent measurement within the instantiation structure

---

I. The Gap DA-3 Closes

DA-2 with the commitment C-DA2 gives a directed tree of instantiations, and branching at each $\perp$ node produces multiple successor instantiations. This raises a question about the cardinality of branching: is the fan at each node countably or uncountably infinite? The answer, which DA-3 formalises, is that this question is perspective-dependent — and DA-3 further conjectures (Claim DA-3-C1, deferred to OQ-E2) that this perspective-dependence is the same structural feature underlying the major cardinality anomalies of classical set theory.

II. Perspective-Dependence of Branching Cardinality

From within instantiation I_n, an observer occupies exactly one branch of I_{n+1}. That the other branches of I_{n+1} are not accessible from it is C-DA2's separateness commitment, not C3: C3 is silent on the boundary between spaces (DA-2 § II). Nor is it monotonicity, which forbids descending to a sibling's starting state, while within one lattice any two states have a join above both, so a monotone sequence up one branch can reach states above another branch's starting state. From inside, the branching factor is always 1: the observer sees one branch. From outside — from a meta-level view of the tree — the branching factor is the full fan of accessible outbound vectors.

Definition DA-3-D1 — Accessible Cardinality

The accessible cardinality of a position p in semilattice L is the cardinality of the set of states reachable from p by monotone sequences within the instantiation containing p.

The accessible cardinality from p is determined entirely by the structure of L above p. It is not an intrinsic property of a collection — it is a property of the relationship between a position and the states reachable from it. No position within any instantiation can access all cardinalities simultaneously. The meta-level view, which sees the full branching fan, is not a position any element of any instantiation can occupy.

Remark R-DA3-1: To observe the full branching fan, one would need to occupy a position outside all instantiations. That position would itself be a state in some semilattice, subject to the same rules. The meta-view is either another instantiation (in which case the tree has no privileged outside view) or $\perp$ itself

(in which case ⊥ is the only position from which the full structure is visible — the state that contributes nothing and is present in everything). The Zero Paradox's name is more precise than it first appeared.

III. The Cardinality Hierarchy as Perspective-Relative

Cantor's theorem establishes that for any set S, $|P(S)| > |S|$, generating the hierarchy $\aleph_0 < 2^{\aleph_0} < 2^{2^{\aleph_0}} < \ldots$ DA-3 reframes this hierarchy not as a fixed ladder that mathematics climbs, but as a perspective-relative description of the branching structure of the instantiation tree, as seen from within different positions.

Claim DA-3-C1 — Perspective-Relative Absolute Cardinality

The appearance of absolute cardinality — cardinality as an intrinsic property independent of measuring position — is an artifact of treating the semilattice as having a view from outside. DA-2 and C-DA2 jointly prohibit such a view from within any instantiation.

The candidate claim (DA-3-C1) is that accessible cardinality from within any instantiation cannot replicate the view from outside. Whether specific independence results in classical set theory are formal expressions of this perspective-dependence is the conjecture that OQ-E2 is tasked with investigating.

Status: DEFINITIONAL ALIGNMENT + CANDIDATE CLAIM. DA-3-D1 and R-DA3-1 are definitional. DA-3-C1 is a candidate claim: structurally motivated within the framework; formal derivation of the connection between accessible cardinality and specific set-theoretic independence results is deferred to OQ-E2.

Updated Open Items Register

Item	Status	Description
AX-1: Binary Snap Causality	RETIRED — shape proved as T-SNAP; the snap occurs given the occurrence commitment and DA-1	Its content was split in two: the shape of the Snap is proved, as Theorem T-SNAP (from L-RUN, TQ-IH and the bottom law, with no Lean axioms), and that the Snap occurs is stated separately: it follows from the occurrence commitment together with DA-1 (closed given DP-2). Premises under Premises of T-SNAP (DA-1 insert, § V); given CC-1, the first-step form of the Snap occurring is the hypothesis hocc of t_snap_given.
DA-1: Derived Proposition (DP-2 formal grounding)	CLOSED given DP-2 — DP-2 (formal core); ZP-A CC-2 + L-INF + AIT argue for the precondition, none deriving it	Primary formal grounding: DP-2 (TrackedOutput, Snap.lean §VI) — da1_minimal_path proved axiom-free. Given the precondition, the machine moves from c_0 to c_1 by D7 (T-SNAP Step 3); da1_minimal_path shows the two configurations return the same output value, so the output value cannot show whether execution occurred. Arguments for the precondition, which is what the occurrence commitment asserts, none deriving it: ZP-A CC-2 + R3 (structural); ZP-C L-INF (informational); AIT incompressibility (Path 3).
DA-2: Instantiation Succession	CLOSED — Definitional	Role half only: a terminal state of I_n that satisfies the A4 role of ⊥ for I_{n+1} is that ⊥. C-DA2 COMMITS to the novelty of each ⊥; it does not establish it.

Item	Status	Description
DA-3: Perspective-Relative Cardinality	CLOSED (definitional) / CANDIDATE (DA-3-C1)	DA-3-D1 establishes accessible cardinality is position-dependent within the instantiation structure. DA-3-C1 (candidate): no position within an instantiation can replicate the outside view. Whether this connects formally to specific set-theoretic independence results is deferred to OQ-E2.
OQ-A1: Increment selection	OQ-A1b CLOSED by A1–A4; OQ-A1a: no reason to restrict to join-irreducibles (well-founded carriers); placement in the ordinal tower, for a monotone map sending the tower's stages to c_0, conditional on the alignment hypothesis $h\varepsilon_0$ (ε_0 sent to c_1)	T5 (Iterative Forcing Theorem), restated in DA-1 insert § VI. No step is guaranteed. On an arbitrary lattice no step is selected: 0, 2, 4, … on the natural numbers satisfies A1–A4 and AX-B1.
OQ-B1: p = 2	CLOSED — ZP-B T0	Derived from AX-B1 and MP-1.
OQ-C1: Non-conservatism of DF	CLOSED — ZP-C T2	Infinite sequence divergence proven. No postulates remain.
S1: Distribution stipulation	CLOSED — ZP-C T1	Derived from AX-B1 and RP-1.
OQ-E1: Sequence vs. tree	CLOSED given the occurrence commitment and C-DA2 — DA-2	The structure is a forward-directed tree, not a linear sequence. Given the occurrence commitment, not a consequence of T-SNAP, DA-2 with C-DA2 supplies the succession and the branching follows. Countable vs. uncountable branching is perspective-dependent (DA-3).
OQ-E2: Cardinality-semilattice correspondence	OPEN	Do specific semilattice structures correspond to specific cardinality regimes? Can the framework make predictions about which instantiations satisfy CH and which do not?

Item	Status	Description
Remaining axioms	INTENTIONAL — AX-B1, AX-G1, AX-G2 (named); A4, quine_unique, bot_self_mem, botCode_is_quine (typeclass fields)	Named commitments: AX-B1 (binary existence), AX-G1 (initial object), AX-G2 (source asymmetry). Structural commitments as typeclass fields: A4/bot_join (ZPSemilattice), AFAStructure.quine_unique and bot_self_mem (the Quine-atom role, which every ZP-A lattice supplies trivially), KleeneStructure.botCode_is_quine (the bottom code is a computational quine, which a constant code meets). Typeclass fields are assumptions carried as hypotheses by the theorems that use them; #print axioms does not surface them. Further commitments are not axioms, among them CC-1 ($S_0 = \perp$), a ZP-A Conditional Claim; T-SNAP's are stated under Premises of T-SNAP (DA-1 insert, § V).
Temperature T in BA-1	PARAMETER — intentional	Universe-contingent. Physical predictions explicitly conditional on instantiation-specific T.

Updated Traceability Register

Claim	Grounded In	Bridge Axiom?	Status
Binary Snap causality	Shape: ZP-C D1, L-RUN, TQ-IH; ZP-A D2; T-SNAP premises: DA-1 insert § V. Occurrence: the occurrence commitment and DA-1	None	Derived for the shape — T-SNAP ✓; the snap occurs given the occurrence commitment and DA-1 (was: Axiomatic — AX-1)
DA-1: Instantiation = execution	DP-2 (TrackedOutput, Snap.lean §VI); its precondition: the occurrence commitment; ZP-A CC-2, R3; ZP-C L-INF	None	Derived Proposition — primary: DP-2 (da1_minimal_path proved axiom-free in Lean). Arguments for its precondition, none deriving it: ZP-A CC-2/R3 (structural), L-INF (informational), AIT incompressibility (Path 3). AIT+ZF+AFA outside Lean scope.
DA-2: Instantiation succession	ZP-A A4, CC-1; ZP-B C3, R1; T-SNAP	None	Definitional Alignment — clarification of CC-1 scope; no new axiom
C-DA2: Novelty of ⊥ (conditional)	DA-2, ZP-B C3	None	CONDITIONAL — a modelling commitment. NOT derived: the separateness of the spaces is assumed, not obtained from C3, and snap_arc_z2_loop returns to the same 0.
DA-3: Perspective-relative cardinality	DA-2, C-DA2, ZP-B R1	None	Definitional (DA-3-D1, R-DA3-1); Candidate (DA-3-C1: outside-view inaccessibility)

Claim	Grounded In	Bridge Axiom?	Status
T-SNAP: Snap shape is derived	L-RUN, TQ-IH (the T-BUF chain's lemmas); T-SNAP premises: DA-1 insert § V. DA-1 bears on occurrence, not on the shape	None	Derived — Cross-Framework ✓
AX-1 retirement	T-SNAP proves the shape; the snap occurs given the occurrence commitment and DA-1	N/A	AX-1 is retired: T-SNAP proves the shape, and that the Snap occurs is stated separately: it follows from the occurrence commitment together with DA-1 (closed given DP-2)
Iterative Forcing T5 (restated)	Selection: succession_succ, ZeroParadox/Ordinal/SnapSuccession.lean § I; for a monotone map sending the tower's stages to c_0, given the alignment hypothesis $h\varepsilon_0$, nothing below ε_0 fires (snap_unconditional). Iteration: T3, R1, t_snap_irreversible, tsnap_holds_but_nothing_moves	None	On an arbitrary lattice no step is selected. No step is guaranteed.
Multiverse — structural implication	T-SNAP + DA-2 + C-DA2 jointly	None	Structural implication given the occurrence commitment — T-SNAP gives the snap's shape, ZP-A CC-2's undifferentiated ⊥ the branching, DA-2 with C-DA2 the succession
OQ-E2: Cardinality correspondence	DA-3	N/A	OPEN — formal derivation deferred

---

Validation Status

Component	Status / Notes
DA-1: Derived Proposition, given DP-2	Valid — DP-2 formal core: da1_minimal_path proved axiom-free in Lean (Snap.lean §VI). TrackedOutput separates output value from machine state; pre- and post-instantiation states are provably distinct even when both produce ⊥. ✓ ZP-K: da1_closed_concrete : IsQuineAtom(⊥ : MachinePhase) proved in Kleene.lean. Path 1's Lean counterpart is da1_closed_concrete, which proves nothing computational; Path 1 rules out an external executor, not an inert ⊥. Path 3 (computational, ZP-C D1 + AIT): bears on the precondition and does not derive executing from incompressibility — incompressibility rules out a shorter external generator, not an inert string; that the configuration reaching P_0 is in Sense B and not Sense A is DP-2's precondition, what the occurrence commitment asserts. Its Lean counterpart is the machinePhaseKleene instance's botCode_is_quine field, a KleeneStructure requirement that constant codes also meet, not a witness of execution and not a second independent proof. da1_paths_unified carries the two Lean counterparts together as a conjunction; that they name one structural fact is the framework's reading. Path 2 (informational bridge, L-INF): UNADOPTED BRIDGE PRINCIPLE — a missing principle, not a missing proof, distinct from the occurrence commitment and not a premise of the Snap; Path 2 does not derive the precondition. No computability library closes the gap between 'system at P_0' and 'system is running.' DA-1's formal grounding (DP-2, da1_minimal_path) does not depend on Path 1, Path 2 or Path 3. CC-1 restated in ZP-J as an equivalence, and not forced (a valid sequence can start elsewhere on a carrier with a second point); ZP-A CC-2 a Forced Metatheoretic Commitment about the host theory, while in an AFAStructure lattice t_exec_iff shows ⊥ is the only Quine atom (IsQuineAtom).
T-SNAP: Binary Snap derived	Valid — Derived, in two readings. The Lean form (t_snap_derived) fixes the SHAPE with no hypotheses and no axioms. The seven-step prose argument uses commitments, and that the transition OCCURS is not established by T-SNAP (tsnap_holds_but_nothing_moves). Premises: DA-1 insert § V, Premises of T-SNAP; two of them (CC-1 and occurrence at the first transition) as hypotheses, t_snap_given. ✓
AX-1 retirement	Valid — AX-1 is retired, and its content was split in two. T-SNAP proves the shape, strengthened from assumed to derived. That the Snap occurs is stated separately: it follows from the occurrence commitment together with DA-1 (closed given DP-2). Given CC-1, its first-step form is the hypothesis hocc in t_snap_given; tsnap_holds_but_nothing_moves shows T-SNAP does not carry it.
DA-2: Instantiation Succession	Valid — Definitional Alignment. Clarification of CC-1 scope. No new axiom. A4 role of ⊥ extended across instantiation boundaries. ✓
C-DA2: Novelty of ⊥ (conditional commitment)	CONDITIONAL — a modelling commitment. NOT derived: the separateness of the spaces is assumed, not obtained from C3, which quantifies only over paths WITHIN one space. snap_arc_z2_loop measures against it — the 2-adic arc returns to the same 0. Must not be cited as derived.

Component	Status / Notes
Directed instantiation tree	Valid given the occurrence commitment — structural consequence of T-SNAP + DA-2 + the commitment C-DA2. Once instantiation is taken to occur, branching follows and is not optional. Forward edges only.
Branching tree structure	Valid given the occurrence commitment — T-SNAP + DA-2 + C-DA2 jointly. T-SNAP fixes the snap's SHAPE and does NOT establish that it occurs; DA-2 with C-DA2 gives the branching tree structure. Universality (all outbound vectors) is DA-2's scope, not T-SNAP alone.
Monotonicity and path irrecoverability	Valid — Structural consequence. Monotonicity (T3) constrains direction; additive ontology (R1) prohibits reduction. Path choice is undetermined by algebra.
Ordinal direction of state sequences	Valid — Derived from ZP-A T3 (monotonicity) and ZP-B C3 (irreversibility). Not assumed.
DA-3: Perspective-Relative Cardinality	Valid (definitional components: DA-3-D1, R-DA3-1). Candidate (DA-3-C1: connection to specific set-theoretic independence results). OQ-E2 open.
DA-3-C1 — outside-view inaccessibility	Candidate — no position within any instantiation can replicate the meta-level view of the branching structure. Formal derivation deferred to OQ-E2.
Remaining axioms: AX-B1, AX-G1, AX-G2; typeclass fields A4, quine_unique, bot_self_mem, botCode_is_quine	Named: AX-B1, AX-G1, AX-G2. Typeclass: A4/bot_join (ZPSemilattice), quine_unique + bot_self_mem (AFAStructure), botCode_is_quine (KleeneStructure). Assumptions carried as hypotheses by the theorems that use them; not surfaced by #print axioms. CC-1 is a further commitment (ZP-A Conditional Claim).
ZP-E results T1–T4, T6, T7	Stated in an earlier version of this document (see the repository history) and not restated here. T5 is restated in DA-1 insert § VI.

ZP-E carries three formal inserts (DA-1, DA-2, DA-3). T-SNAP is derived, with its premises stated in the DA-1 insert § V and two of them (CC-1 and occurrence at the first transition) as the hypotheses of t_snap_given; its irreversibility rests on ZP-A R1 and ZP-B C3, with ZP-G downstream. DA-1 is closed given DP-2 (da1_minimal_path, axiom-free). Path 1's Lean counterpart is da1_closed_concrete (ZP-K), which proves IsQuineAtom(⊥ : MachinePhase) and nothing computational; Path 3's Lean counterpart is the machinePhaseKleene instance's botCode_is_quine field, a KleeneStructure requirement, not a witness of execution; and the step from Path 2 to executing is a bridge principle of its own (a missing principle, not a missing proof), which the framework does not adopt and which is distinct from the occurrence commitment; and Path 3 bears on the precondition and does not derive executing from incompressibility: that the configuration reaching P_0 is in Sense B is what the occurrence commitment asserts, and DA-1 consumes it. Remark R-ε_0 justifies the ε_0 symbol as a structural analogy. Remark R-AFA rules Foundation out as a host of the Quine-atom role (Regularity, no_quine_atom; zfset_quine_role_unoccupied): the forcing is machine-checked (quineAtomHost_not_wellFounded), and a narrow Forced Metatheoretic Commitment remains. Open question: OQ-E2. Remaining axioms: AX-B1, AX-G1, AX-G2.