

# THE ZERO PARADOX

*ZP-C: Information Theory*

*Version 1.4 | April 2026*

*Supersedes v1.3 | L-RUN formalized; TQ-IH answered; T-BUF added; AX-1 promoted to Candidate Theorem*

This document is self-contained within information theory and discrete analysis on  $Q_2$ . The topological structure of  $Q_2$  — specifically total disconnectedness (ZP-B T5), the clopen ball hierarchy, and the binary existence axiom (AX-B1) — is imported from ZP-B as a dependency. Every claim is marked as Derived, Axiomatic, Defined, or Candidate.

Version 1.4 changes: (1) Lemma L-RUN formalized: the act of execution is a non-null state change, derived from AX-B1 and D7. (2) Test Question TQ-IH answered negatively by L-RUN — no program outputs  $\perp$  without a non-null intermediate configuration state. (3) Candidate Theorem T-BUF added: at  $P_0$ , execution is structurally guaranteed and that execution state is  $\varepsilon_0$  in the semilattice. (4) AX-1 status updated from Axiomatic to Candidate Theorem.

## I. Kolmogorov Complexity and the Incompressibility Threshold

Let  $x$  be a binary string of length  $n$ . The conditional Kolmogorov complexity is:

$$K(x|n) = \min \{ |p| : U(p, n) = x \}$$

where  $U$  is a fixed universal Turing machine and  $|p|$  is the length of program  $p$ .

### Definition D1 — Incompressibility Threshold

$$P_0 = \lim_{n \rightarrow \infty} K(x|n) / n$$

$P_0$  is the algorithmic entropy rate of  $x$ . When  $P_0$  reaches its maximum,  $x$  has exhausted its capacity for self-description.

### Remark R1 — Scope of $P_0$

What  $P_0$  establishes (Derived): the point at which  $x$  becomes incompressible.

What  $P_0$  does not establish (prior versions: Axiomatic): that incompressibility causes the Binary Snap. Version 1.4 updates this: the generative claim is now a Candidate Theorem (T-BUF), not a bare axiom. See Section V.

## II. Informational Work: State Representations and JSD

### 2.1 The Representation Principle

### Principle RP-1 — Minimum Sufficient Probabilistic Representation

The minimum sufficient probabilistic representation of a binary ontological state is a point-mass (Dirac) distribution over  $\{0,1\}$ : all probability mass is assigned to the value the state occupies, and none to the other value.

Justification: A point-mass distribution is the unique distribution that is (a) faithful — assigns non-zero probability only to the ontologically actual value, and (b) non-redundant — carries no uncertainty about which state is occupied. Any distribution with mass at both values represents partial existence and partial non-existence simultaneously. AX-B1 excludes this.

Status: PRINCIPLE — representational commitment parallel to MP-1 in ZP-B.

### Theorem T1 — State Representations are Uniquely Derived

Given AX-B1 and RP-1, the distributions are unique:

$P = (1, 0)$  (Null State: all mass at 0 — non-existence)

$Q = (0, 1)$  (First Atomic State: all mass at 1 — existence)

Proof: AX-B1 establishes two ontological states. RP-1 requires point-mass representation of each. The Null State occupies value 0:  $P = (1,0)$ . The First Atomic State occupies value 1:  $Q = (0,1)$ . No distribution  $(p, 1-p)$  with  $0 < p < 1$  is consistent with AX-B1.  $P$  and  $Q$  are unique. Status: DERIVED from AX-B1 and RP-1. ✓

### Theorem T1b — JSD = 1 bit

For  $P=(1,0)$  and  $Q=(0,1)$  with  $M=(1/2, 1/2)$ :

$D_{KL}(P\|M) = 1$  bit,  $D_{KL}(Q\|M) = 1$  bit,  $JSD(P\|Q) = 1$  bit

$E = JSD(P\|Q) = 1$  bit [information-theoretic, dimensionless]

Status: Derived from AX-B1 and RP-1 via T1.  $E$  is not physical energy in joules. Dimensional bridge belongs to ZP-E as BA-1.

### Definition D3 — Dirac Measure $\delta_0$

$$\int_{\Omega} f d\delta_0 = f(0)$$

$\delta_0$  places unit mass at 0. Compatible with the discrete topology of  $\{0,1\}$  and with the totally disconnected topology of  $Q_2$  (ZP-B T5).  $\delta_0$  governs behaviour exactly at  $x = 0$ .

### Remark R2 — Hamming Cross-Validation

$$d_H(P, Q) = 1 \text{ [Hamming, on } \{0,1\}]$$

The agreement between  $d_H = 1$  and  $E = 1$  bit is a consistency check, not a proof that Hamming distance and JSD are equivalent in general. Both are computed by independent methods on the same AX-B1-derived distributions.

### III. The Discrete Surprisal Field on $Q_2$

#### Remark R3 — Why the Smooth Embedding Remains Retired

ZP-B T5 establishes  $Q_2$  is totally disconnected. ZP-B T2 establishes every ball is clopen. Importing smooth calculus operators onto a totally disconnected space imports a smoothness assumption that contradicts ZP-B. The smooth embedding, MO-1, and P1 from v1.1 remain retired.

#### Definition D4 — Discrete Surprisal Function I

For  $x \in Q_2$  with  $x \neq 0$  and  $P(x) > 0$ :

$$I(x) = -\log_2 P(x)$$

As  $v_2(x) \rightarrow +\infty$  ( $x$  approaches 0 in the 2-adic metric),  $I(x) \rightarrow +\infty$ : states 2-adically close to 0 are informationally extreme.

The probability measure  $P$  on  $Q_2 \setminus \{0\}$  is the branching measure induced by the binary ball hierarchy of  $Q_2$ : at each level  $k$ , the two sub-balls of  $B(0, 2^{-k})$  each receive half the probability mass of their parent ball.

#### Definition D5 — Discrete Surprisal Difference Operator DF

For  $x, y \in Q_2 \setminus \{0\}$ :

$$DF(x, y) = I(y) - I(x) = \log_2[P(x) / P(y)]$$

DF is antisymmetric:  $DF(x, y) = -DF(y, x)$ . No smoothness is assumed; DF is defined entirely pointwise.

#### Definition D6 — Discrete Circulation (Extended)

FINITE CASE: A discrete loop  $\gamma_n = (x_0, x_1, \dots, x_n, x_0)$  in  $Q_2 \setminus \{0\}$  returning to its start. Circulation:  $C(DF, \gamma_n) = \sum_{i=0}^n DF(x_i, x_{i+1})$  where  $x_{n+1} = x_0$ .

Note: By telescoping,  $C(DF, \gamma_n) = I(x_0) - I(x_0) = 0$  for all finite loops. DF is conservative on all finite loops.

INFINITE CASE: An infinite sequence  $\sigma = (x_1, x_2, \dots)$  in  $Q_2 \setminus \{0\}$  where  $v_2(x_i) = i$  for all  $i \geq 1$ . Partial circulation:  $C_n(DF, \sigma) = I(x_{n+1}) - I(x_1)$ .

Circulation of  $\sigma$ :  $C(DF, \sigma) = \lim_{n \rightarrow \infty} C_n(DF, \sigma)$  if the limit exists or diverges.

An infinite sequence  $\sigma$  exhibits non-conservative behaviour if  $C(DF, \sigma)$  diverges.

#### Theorem T2 — DF Exhibits Non-Conservative Behaviour on Infinite Sequences Approaching 0

DF is conservative on all finite loops ( $C = 0$  by telescoping). On infinite sequences through the ball hierarchy approaching 0, the circulation diverges:  $C(DF, \sigma) \rightarrow +\infty$ .

### Theorem T2 — DF Exhibits Non-Conservative Behaviour on Infinite Sequences Approaching 0

Proof: Let  $\sigma = (x_1, x_2, \dots)$  with  $x_i = 2^i$ . Then  $P(x_i) = 2^{-i}$ , so  $I(x_i) = i$ . Partial circulation  $C_n = I(x_{n+1}) - I(x_1) = n$ . As  $n \rightarrow \infty$ :  $C(\text{DF}, \sigma) = +\infty$ .

Divergence arises from the unbounded depth of the 2-adic ball hierarchy (ZP-B). Finite conservation and infinite divergence are distinct regimes; no contradiction arises.

Status: DERIVED from ZP-B ball hierarchy structure and branching measure on  $Q_2$ . OQ-C1 closed.  
✓

## IV. Execution as State: The Hardware Lemma

### Definition D7 — Machine Configuration

A machine configuration  $c$  is a complete description of a Turing machine at a given moment: the current state of the control unit, the position of the read/write head, and the contents of all tape cells.

$c_0$ : the initial configuration — the machine exists but has not yet begun execution (no instruction has been fetched).

$c_1$ : the first running configuration — the machine has fetched and begun executing its first instruction.

Note: D7 does not import physics. It is a structural distinction between two discrete machine states within the standard Turing model.

### Lemma L-RUN — Execution is a Non-Null State Change

The transition from  $c_0$  (initial: not yet running) to  $c_1$  (running: first instruction fetched) is a discrete, irreducible state change in the machine configuration.

Proof:

Step 1 —  $c_0$  and  $c_1$  are distinct configurations by D7: the control unit is in a different state (idle vs. executing).  $c_0 \neq c_1$ .

Step 2 — By AX-B1, a state either exists or it does not. The machine at  $c_1$  occupies a configuration distinct from  $c_0$ . The transition  $c_0 \rightarrow c_1$  is a binary state change in the sense of AX-B1.

Step 3 — The transition is irreducible: there is no intermediate configuration between  $c_0$  and  $c_1$ . The first instruction fetch is the minimal unit of execution.

Step 4 —  $c_0$  corresponds to  $\perp$  (no distinguishable structure).  $c_1$  is strictly above  $\perp$  in the semilattice: the machine configuration has gained content (an active execution context) not present in  $c_0$ .

### Lemma L-RUN — Execution is a Non-Null State Change

Conclusion:  $c_1 \neq \perp$ . The act of execution is itself a non-null state, regardless of what the output tape contains. ✓

Status: DERIVED from AX-B1 and D7. No Coding Theorem required. No output-tape contents required.

### Remark R4 — Configuration and Output Are Independent

D-TQ-2 (Output Definition): A computation's output is defined by the contents of its tape/register at termination.

L-RUN establishes that configuration state and output state are independent. A program may terminate with null output while still having passed through non-null configuration states during execution.

The question "can a program output  $\perp$  without any non-null intermediate state?" is answered by looking at the execution trace, not the output tape. This distinction is load-bearing for T-BUF and TQ-IH.

## V. The Test Question and the Buffer Overflow Theorem

### Test Question TQ-IH — Can a Program Output $\perp$ Without a Non-Null Intermediate State?

Question: Does there exist a program  $p$  such that  $U(p) = \perp$  and the execution trace  $\tau(p)$  contains no configuration  $c_i$  with  $c_i \neq \perp$ ?

Answer: No.

Proof:

Step 1 — Any program that executes must pass through  $c_1$  (by definition of execution, D7).

Step 2 — By L-RUN,  $c_1 \neq \perp$ .

Step 3 — Therefore the execution trace  $\tau(p) = (c_0, c_1, \dots)$  contains  $c_1$ , which is a non-null configuration state, regardless of what appears on the output tape at termination.

Step 4 — The output of  $p$  being  $\perp$  does not imply that all configurations in  $\tau(p)$  are  $\perp$ . Output state and configuration state are distinct (R4).

Conclusion: No program can produce  $\perp$  as output without passing through a non-null intermediate configuration state. TQ-IH answered negatively. ✓

Status: DERIVED from L-RUN, D7, and R4. No Kolmogorov machinery required. AX-1 derivability pathway now open.

### Candidate Theorem T-BUF — Incompressibility Forces Non-Null Execution State

Statement: At the incompressibility threshold  $P_0$ , the Binary Snap  $\perp \rightarrow \varepsilon_0$  is a structural consequence of execution, not an external trigger.

Step 1 —  $P_0$  identifies the configuration  $x$  at which  $K(x|n)/n = 1$ : the configuration string is incompressible. (D1)

Step 2 — An incompressible configuration that instantiates a computation must execute. The configuration at  $P_0$  is a live machine state. To be at  $P_0$  is to be a configuration string that is executing. (See DA-1 in ZP-E)

Step 3 — Any execution passes through  $c_1$  (L-RUN).  $c_1 \neq \perp$  (L-RUN conclusion).

Step 4 — In  $(L, v, \perp)$ , this non-null configuration state is  $c_1 = \perp \vee \varepsilon_0$ . By ZP-A D2, this is the Binary Snap.

Conclusion: At  $P_0$ , execution is structurally guaranteed. Execution guarantees a non-null configuration state. That state is  $\varepsilon_0$  in the semilattice. AX-1 is derivable from  $P_0 + L-RUN + TQ-IH + ZP-A D2$ . ✓

Status: CANDIDATE THEOREM — structurally complete within ZP-C. Cross-framework integration (DA-1) owned by ZP-E. The gap between incompressibility and forced transition is closed at the configuration level.

### Remark R5 — Updated Status of AX-1

Prior to v1.4: AX-1 (Binary Snap Causality) was labeled Axiomatic in ZP-C.

As of v1.4: AX-1 is a Candidate Theorem. The derivation pathway:  $P_0$  (D1) identifies the threshold. L-RUN establishes that execution at the threshold constitutes a non-null state change. TQ-IH establishes that no program avoids this. ZP-A D2 establishes that a non-null state change from  $\perp$  is the Binary Snap.

Remaining work: DA-1 (Definitional Alignment) must formally tie instantiation of  $P_0$  to an execution event. This is owned by ZP-E.

Status label: CANDIDATE THEOREM — gap identified and named (DA-1). Closed in ZP-E DA-1 insert.

## VI. Open Items Register for ZP-C v1.4

Item	Status	Description
S1: Distribution stipulation	Closed — T1	T1 derives P and Q from AX-B1 and RP-1.
OQ-C1: Non-conservation	Closed — T2 rebuilt	Telescoping critique resolved; infinite divergence proven within extended D6.

Item	Status	Description
Smooth embedding, MO-1, P1	Retired	Remain retired from v1.2. Inconsistent with ZP-B.
RP-1: Representation principle	Principle — explicit	Bridge between AX-B1 and probabilistic tools.
D7: Machine configuration	Defined	Foundation for L-RUN and TQ-IH.
L-RUN: Hardware Lemma	Derived — Lemma	Execution is a non-null state change. Derived from AX-B1 and D7.
TQ-IH: Test Question	Closed — Negative	No program can output $\perp$ without a non-null intermediate configuration state. Proven by L-RUN.
T-BUF: Buffer Overflow Theorem	Candidate Theorem	Incompressibility forces non-null execution state. DA-1 bridge in ZP-E closes this fully.
AX-1: Binary Snap Causality	Candidate Theorem	Derivation pathway formalized. Closed as T-SNAP in ZP-E DA-1 insert.

## VII. Validation Status

Component	Status / Notes
$K(x n)$ and $P_0$ (D1)	Valid — standard algorithmic IT
R1: Scope of $P_0$	Valid — updated: AX-1 now Candidate Theorem, not bare axiom
RP-1: Representation Principle	Valid — Principle; explicit bridge; resolves reviewer gap in T1
T1: Distributions from AX-B1 + RP-1	Valid — Derived; reviewer gap closed
D2: JSD definition	Valid — standard
T1b: $E = \text{JSD} = 1$ bit	Valid — Derived from AX-B1 + RP-1
D3: Dirac measure $\delta_0$	Valid — standard; compatible with discrete $Q_2$ topology
R3: Smooth embedding retired	Valid — Retired; inconsistent with ZP-B
D4: Discrete surprisal $I(x)$	Valid — pointwise on $Q_2 \setminus \{0\}$ ; branching measure stated
D5: Difference operator DF	Valid — antisymmetric; no smoothness assumed
D6: Circulation (extended)	Valid — finite and infinite cases both defined; finite conservation acknowledged
T2: Non-conservation rebuilt	Valid — Derived; telescoping critique addressed; divergence at $\sigma \rightarrow 0$ established

Component	Status / Notes
R2: Hamming cross-validation	Valid — Consistency check
D7: Machine configuration	Valid — Defined; standard Turing model; no physics imported
L-RUN: Hardware Lemma	Valid — Derived from AX-B1 and D7
R4: Configuration vs. output independence	Valid — load-bearing distinction for TQ-IH and T-BUF
TQ-IH: Test question answered	Valid — Derived by L-RUN; no Kolmogorov machinery required
T-BUF: Candidate Theorem	Candidate — structurally complete in ZP-C; DA-1 bridge in ZP-E closes fully
R5: AX-1 status updated	Valid — AX-1 is Candidate Theorem; prior Axiomatic status corrected