

TFPT — Safeguards against Coincidence and Numerology

The verification discipline: how every load-bearing claim is defended against chance, fitting and over-reading

Stefan Hamann Alessandro Rizzo

June 25, 2026 · v5.3 (rev 318)

The TFPT document set — what is treated where

Plain language: TFPT (Topological Fixed-Point Theory) is a small discrete compiler. Two inputs — the seam constant $c_3 = \frac{1}{8\pi}$ and the carrier rank $g_{\text{car}} = 5$ — build $D_5 \oplus A_3 + \mu_4 \Rightarrow E_8$ and read off the Standard-Model skeleton, the constants and the scale grammar. The series is **the introduction plus five numbered papers and three companions**, best read in order (with **four** companions):

0. **introduction** — reading guide, compiler closure, predictions, the dependency DAG and the single proof ledger (`verification/status_ledger.csv`). (*entry point*)
1. **tfpt_1_architecture_e8** — the two axioms, the derivation map, the EM fixed point α^{-1} , the $D_5 \oplus A_3 + \mu_4 \Rightarrow E_8$ construction, the QBL theorem chain.
2. **tfpt_2_standard_model** — the SM in one φ_0 -ladder formula, flavor from parabolic transport (sheet diamond, dual anchor, Q_+ cohomology), the worked closures.
3. **tfpt_3_e8_audit_bootstrap** — the seven E_8 slices as an audit raster, the cascade bridge, the Möbius bootstrap.
4. **tfpt_4_frontier** — honest status of η_B , m_p/m_e , Koide, dark matter, full QG.
5. **tfpt_5_redteam** — the adversarial audit: declared attacks, what survives, what each reduces to, and the kill tests.

Companions: **R.** **tfpt_research_contracts** — the open interfaces ($v_{\text{geo}}, G_{\text{net}}, F_{\text{transfer}}$) as numbered contracts; **H.** **tfpt_horizon_readouts** — Appendix H (unit-system reframe): one seam constant as the universal horizon thermal code, the Nariai/anchor surface, the resummed seam clock; **O.** **origin_theory** — why no free fundamental number remains (exact core + one honestly-typed cyclic interpretation); **S.** **tfpt_safeguards** — the verification discipline: every mechanism that defends a claim against coincidence and numerology (status calculus, no-free-pattern, the over-determination map, the firewall, frozen predictions + null model, the independent Wolfram and Lean paths, the red team).

*You are reading the **Safeguards** companion (S): it collects, in one place and in one language, every mechanism by which TFPT protects a claim against coincidence — the status calculus, the anti-fitting rules, the over-determination accounting, the firewall, the frozen predictions and null model, the two independent proof paths (Wolfram, Lean), the red team, and the honest residual. It introduces no new physics; it makes the discipline auditable.*

Abstract

A two-input theory that reads out many small integers is, a priori, at risk of being elaborate numerology. TFPT answers that risk not with rhetoric but with a *layered, machine-checked discipline*, and this companion states all of it uniformly. The layers are: (1) a four-class status calculus with a single-source ledger and a sync audit that forbids a conditional claim from being rendered as exact; (2) an anti-fitting rule (“no free pattern”) plus a reverse audit that counts how much structure carries *no* readout; (3) an over-determination map that honestly separates *multiplicative* evidence from *compression* (many readouts from one generator) and, applied to TFPT’s own arithmetic witnesses, finds them to be facets of *one* $(2, 3, 5)/E_8$ object (compression), locating the genuine multiplication in an input forced four independent ways plus a foreign readout (α^{-1}); (4) a firewall that keeps external-physics interfaces from ever being typed as compiler outputs, together with the No-Unit theorem that makes the absence of an

absolute scale a *theorem*; (5) a frozen prediction registry with a Monte-Carlo null model and a live data scorecard; (6) two independent reproduction paths — an independent Wolfram engine and a Lean 4 kernel proof; (7) an adversarial red-team layer; and a final, explicit statement of what would still falsify the theory. The thesis is deliberately modest: these safeguards make coincidence an *expensive* explanation of the discrete core, and they keep “exact compiler closure” from ever being mistaken for “closed physics”.

1 The risk, and the shape of the answer

The honest danger for TFPT is not that an individual number is wrong — the arithmetic is usually clean — but *closure inflation*: that an exact, discrete *compiler* closure is read, or worded, as a *physical* closure. Every safeguard below exists to hold that one line. None of them, alone, turns syntax into semantics; together they make the discrete core hard to dismiss as chance and impossible to oversell as a finished Theory of Everything. We state each as a *layer* with its enforcing artefact(s).

The layered defense — a load-bearing claim must pass all seven

1 Status calculus [E]/[C]/[O]/[X] · ledger single source · audit_sync
2 No free pattern + reverse audit v305 · E8.REVERSE.AUDIT (3/8 readout)
3 Over-determination map v427/v428 · seven readouts compress one (2,3,5)/E8 object
4 Firewall + No-Unit v187/v213 · v153 (v_geo theorem-forbidden)
5 Frozen registry + null model v84 · v100 Monte-Carlo · v375 scorecard
6 Two independent paths Wolfram 116+327 · Lean 4 kernel proof
7 Red team tfpt_5 · Targets A-F · named kill tests

Figure 1: The seven safeguard layers; a load-bearing claim must pass all of them. Each layer names its enforcing artefact (script id or document).

2 Layer 1 — the status calculus (no [C] dressed as [E])

Every claim carries one of four display markers: **[E]** exact / proven (identity, Lie/lattice theorem, Lean-formalised, or numerical fixed point), **[C]** conditional (physical, under named hypotheses), **[O]** open / axiom / anchor, **[X]** falsifiable kill test. The finer per-claim type (Axiom / Formal / Lattice / Numerical / Identity / Physical) lives in the *single source of truth*, `verification/status_ledger.csv`; the papers and the website only *mirror* it. A sync audit (`verification/audit_sync.py`, run as `bash build.sh audit`) enforces, in both directions, that the suite, the runner, the registry and the ledger agree; that every script is cited in a paper *body*; and that no generated surface is stale. The point of the calculus is exactly the reviewer’s concern: it is structurally impossible for a **[C]** claim to be silently rendered as **[E]**, because the marker is read from the ledger, not retyped per document.

3 Layer 2 — no free pattern, and the reverse audit

Anti-fitting: an identity earns its keep or it is decorative

The *forward discipline* / generator-economy audit ([verification/v305_witness_independence.py], null model [verification/v100_numerology_null_mc.py]) admits an identity as load-bearing only if it is derivationally necessary, kills alternatives, is ablation-relevant, links modules, or is experimentally testable; admissible invariant classes are bounded in advance. The complementary *reverse audit* (E8.REVERSE.AUDIT.01) asks the honest opposite question — how much E_8 structure maps onto *nothing*? Of the eight Casimir degrees $\{2, 8, 12, 14, 18, 20, 24, 30\}$, exactly **three** feed a primary physical readout (degree 2, the metric; degree 8, the rank $\rightarrow c_3 = 1/(8\pi)$; degree 30, the Coxeter number $\rightarrow g_{\text{car}} = 5$ and the order-30 clock) and **five** carry no primary readout. A theory that publishes its 5/8 unused overhead is not cherry-picking.

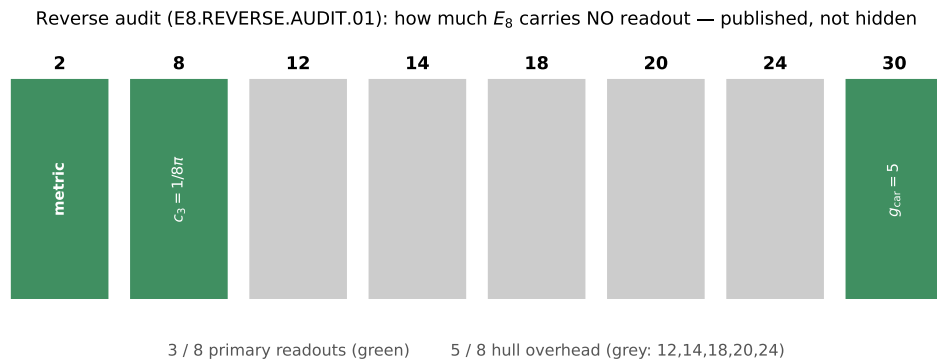


Figure 2: The reverse audit (E8.REVERSE.AUDIT.01): of E_8 's eight Casimir degrees only three feed a primary readout; the other five are published hull overhead.

4 Layer 3 — the over-determination map (multiply vs. compress)

The strongest positive evidence is over-determination, but it must be counted honestly. The *framework* is a sharp axis ([verification/v427_overdet_witness_map.py], extending the generator-economy audit v305): evidence *multiplies* only across genuinely disjoint grammars; many readouts from one generator are a *compression* gain, not independent multiplication. The hard part is classifying TFPT's own witnesses correctly — and applying that test to ourselves ([verification/v428_overdet_reclass.py]) forces a self-correction that we state in full.

Seven arithmetic readouts — and why they compress

Each integer below is computed *from its own grammar* and lands on a carrier-skeleton value:

Gauss $\mathbb{Z}[i]$:	$N(3+2i) = 3^2+2^2$	= 13 (= Δ_Q)
Eisenstein $\mathbb{Z}[\omega]$:	$N(3+2\omega) = 3^2-3\cdot 2+2^2$	= 7
Cyclotomy $\mathbb{Q}(\zeta_5)$:	$N(3+2\zeta_5) = 2^4 \Phi_5(-3/2)$	= 55
Galois $(\mathbb{Z}/5)^\times$:	$ \text{Gal } \mathbb{Q}(\zeta_5) = \varphi(5)$	= 4 (= $ \mu_4 $)
Root lattice :	$ \det \text{Cartan}(E_8) $	= 1
Pascal/exterior :	$\binom{4}{0} + \binom{4}{1} + \binom{4}{2} = 11, 2^4$	= 16 = $\dim S^+$
Coxeter :	$h(E_8) = 30 = 2\cdot 3\cdot 5, \varphi(30)$	= 8 = $\text{rank } E_8$

These are seven *syntactically* distinct grammars — but syntactic distinctness is not statistical independence. By the Brieskorn classification (v236) the $(2, 3, 5)$ singularity $x^2+y^3+z^5$ is the *one* generator behind the order-30 clock, with Milnor number $\mu = (2-1)(3-1)(5-1) = 8 = \text{rank } E_8$ and $30 = 2\cdot 3\cdot 5 = h(E_8)$. Each readout above is a facet of that same object: $\mathbb{Z}[i]$ is the prime-2 (μ_4 /sheet) face, $\mathbb{Z}[\omega]$ the prime-3 (family) face, $\mathbb{Q}(\zeta_5)$ and the Galois group the prime-5 (carrier) face, Pascal the μ_4 exterior, the Coxeter number the whole clock, and $\det \text{Cartan}(E_8)$ the E_8 resolution itself. *So the seven do not multiply* —

they compress, exactly like the anchor $a = (1, 1, 2)$ emitting $(e_1, e_2, e_3) = (4, 5, 2)$, the power sums, and the E_8 data from one generator. An earlier version of this section called the seven “disjoint grammars that multiply”; [verification/v428_overdet_reclass.py] applies this layer’s own test to that claim and walks it back.

What honestly multiplies: forced inputs + a foreign witness

The genuine over-determination is of a different shape. First, the *input* of the theory is forced by several *independent* arguments: the “8” in $c_3 = 1/(8\pi)$ equals rank $E_8 = 8$, the carrier Coxeter number $h(D_5) = 2(5-1) = 8$, the totient $\varphi(30) = 8$, and the Milnor number $(2-1)(3-1)(5-1) = 8$ — four *different* mathematical facts that independently fix the same constant, and that genuinely multiply. Second, the constant the theory was *not* built around — $\alpha^{-1} \approx 137$, a prime foreign to the $(2, 3, 5)$ skeleton — is reproduced as the fixed point of the boundary Ward identity (v305). The honest claim is therefore narrower and harder to attack: not “seven independent witnesses,” but *one richly over-determined input plus a foreign readout*.

Corrected over-determination map (v428): seven arithmetic readouts COMPRESS one $(2, 3, 5)/E_8$ object; the input forced four ways + the foreign α^{-1} are what multiply

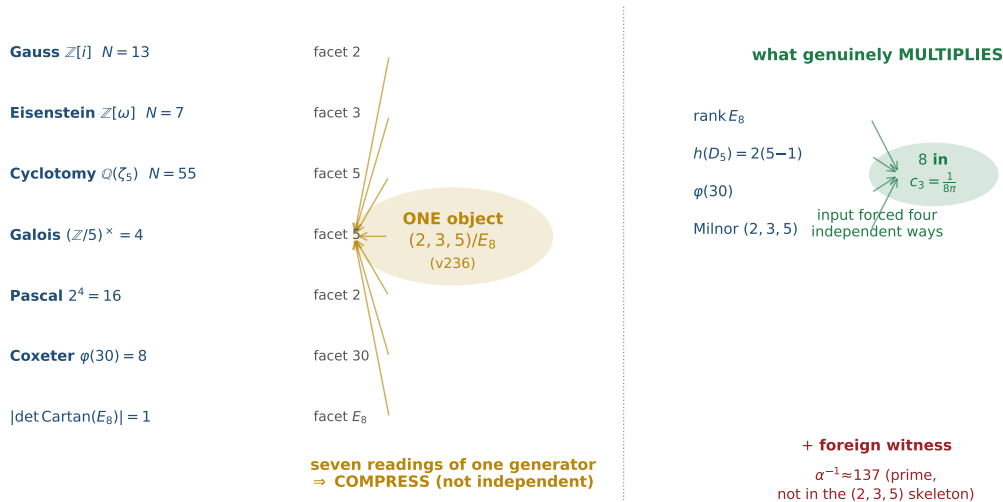


Figure 3: The corrected over-determination map ([verification/v428_overdet_reclass.py]): the seven arithmetic readouts are facets of *one* $(2, 3, 5)/E_8$ object (a *compression* gain, like the anchor), so they do not multiply; what genuinely multiplies is the input forced four independent ways (the “8”) plus the foreign witness $\alpha^{-1} \approx 137$.

5 Layer 4 — the firewall and the No-Unit theorem

Two safeguards keep the boundary between the closed core and external physics sharp.

- **The firewall.** The four frontier transfers — F_{pole} (Koide), $F_{\text{Boltzmann}}$ (η_B), F_{relic} (axion), F_{QCD} (m_p/m_e) — are *typed interfaces*, a discrete compiler kernel composed with an external continuous solver. A guard ([verification/v187_fttransfer_laws.py]) reads the ledger and *enforces* that none is ever typed as a primitive [E] compiler prediction; the functor contract is [verification/v213_fttransfer_functor.py] and a prose sentinel ([verification/v188_frontier_wording_guard.py]) guards the wording. Exact algebraic sub-parts may be [E] (the Koide 53/54 factor, $b_3 = -7$); the physical observable never is. The recent single-flow reduction ([verification/v425_dyn_transfer_universal.py]) makes the *dynamics* of all four the one native seam recovery semigroup, leaving only named anchors external — a

sharpening that respects, not relaxes, the firewall.

- **The No-Unit theorem** ([verification/v153_no_unit_theorem.py]). Because c_3 and g_{car} are mass-dimension 0, no dimensionless compiler can output an absolute scale; v_{geo} is therefore *forbidden by theorem*, not a missing prediction. The residual-certification audit ([verification/v384_residual_certification.py]) makes the shape explicit: every open item is an external math proof, theorem-forbidden (the unit), or external physics — *zero* open internal mechanisms.

6 Layer 5 — frozen predictions, the null model, and the α uniqueness scan

Predictions are frozen before confrontation, and a match is scored against *chance*, not admired in isolation.

The look-elsewhere null model, made explicit ([verification/v100_numerology_null_mc.py])

The frozen registry ([verification/v84_frozen_registry.py], REG.FREEZE.01) fixes every dimensionless prediction at 25 digits, re-derived from the two axioms each run (a formula \leftrightarrow value lock). The null model then runs an *exact census* of the full declared formula grammar over the 13 scored observables with conservative data windows: the joint formula-fishing probability is $\prod_i p_i = 10^{-25.8}$, and $2 \times 200,000$ Monte-Carlo pseudo-theories reach *at most* 5/13 hits against TFPT's 13/13 (Fig. 4). Power controls confirm the test bites: $\varphi_0^{\text{ret}} \pm 10\%$ collapses TFPT's own scorecard to 6/13, and a data shuffle averages 1.2/13.

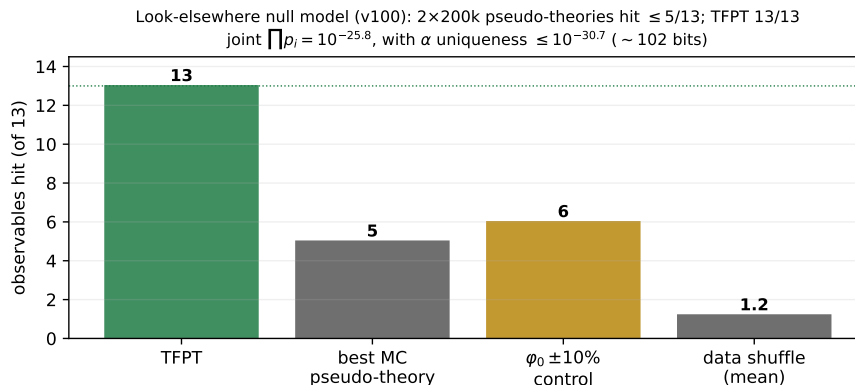


Figure 4: The look-elsewhere null model ([verification/v100_numerology_null_mc.py]): Monte-Carlo pseudo-theories hit at most 5/13 where TFPT hits 13/13; the power controls ($\varphi_0^{\text{ret}} \pm 10\%$, data shuffle) confirm the test is non-vacuous.

Is the α path unique? The sharpest single result is the fine-structure fixed point. All 94,500 variants of the boundary $U(1)$ Ward formula $F_{U(1)}$ in the declared grammar are root-solved; *exactly one* — the budget $M=41$ at $N=8$ ($c_3 = 1/8\pi$) — lands in the 4×10^{-8} CODATA window (Fig. 5). Folding this into the census gives a total $\leq 10^{-30.7}$ (~ 102 bits), *conditional on the declared grammar* — a null-model rejection, never a claim of certainty.

No moving goalposts; the worst case is shown. A live data scorecard records the JUNO/NuFIT/ACT/BK18 compatibility ([verification/v375_observatory_registry.py]). A data watchdog types each row as PASS, WATCH, TENSION or KILL ([verification/v307_data_watchdog.py]), and a kill-test board keeps the falsifiers tied to the frozen rows ([verification/v321_killtest_board.py]). The discipline is applied to its own weakest point: the reactor angle $\sin^2 \theta_{13} = \varphi_0^{\text{ret}} e^{-5/6}$ is *pre-registered* at $+2.0\sigma$ tension (FLAV.TH13.PRESSURE.01) with its own kill criterion — the opposite of hiding the worst case.

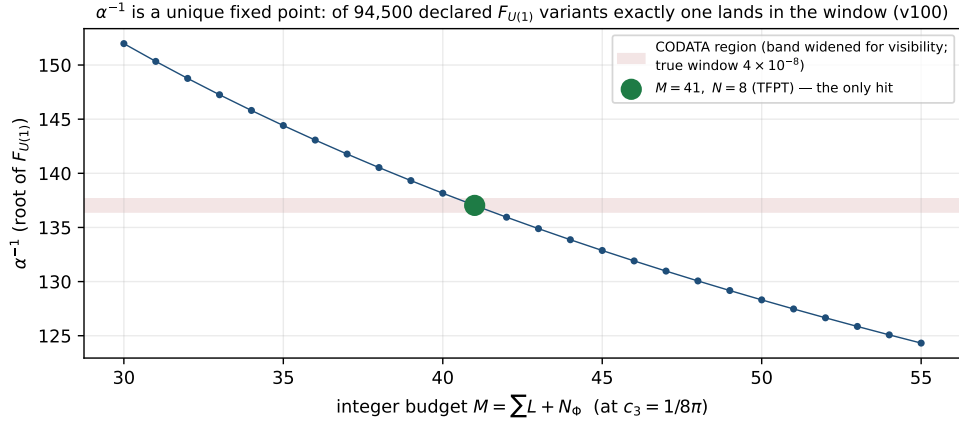


Figure 5: α^{-1} as a unique fixed point: scanning the integer budget M at $c_3 = 1/8\pi$, only $M=41$ lands in the CODATA window; the full census over all 94,500 $F_{U(1)}$ variants finds exactly one hit.

7 Layer 6 — two independent reproduction paths

A single implementation can share a single bug; TFPT therefore re-derives the exact core twice more, independently.

- **An independent Wolfram engine** (GATE.WOLFRAM.01/GATE.WOLFRAM.02): 116/116 base checks plus a 327/327 extension mirror every exact algebraic / identity / lattice result in a different computer-algebra system, ending ALL WOLFRAM CHECKS PASSED. (Numerical ODE/PDE results are Python-only by design and flagged as such.)
- **A Lean 4 kernel proof**: the load-bearing algebra is machine-formalised with no sorry — the carrier hypercharge as the exact anomaly-free root of $6Y^2 - Y - 1 = 0$, the Pascal carrier ladder $2^4 = \binom{5}{0} + \binom{5}{1} + \binom{5}{2}$ (FORM.LADDER.01), the integer rigidity of the carrier, the seam-equivalence chain (FORM.SEAMEQUIV.01) and its collar/MMST face (FORM.SEAM.MMST.01) — so the kernel, not the prose, certifies the core.

8 Layer 7 — the red team

The adversarial audit (companion `tfpt_5_redteam`) does not defend; it *attacks*. It declares six targets — A (the seam = $(E_8)_1$ identification), B (the carrier truncation $g_{\text{car}}=5$), C (the $k=c_3/2$ area law), D (the ratios $\rightarrow v_{\text{geo}}$ reduction), E (whether one scale carries the theory), F (the perturbative 4D-QFT round) — states what survives each, and reduces every survivor to a named residual. The model example is its own honesty about holomorphy: $c = 8$ alone does *not* select $(E_8)_1$ because $(D_8)_1 = SO(16)_1$ shares $c = 8$; the load-bearing extra is holomorphy ($|\det K| = 1$). Publishing the sharpest attack on oneself is the strongest anti-numerology signal a programme can give.

9 What would still falsify TFPT

The safeguards are only credible with explicit kill tests (companion “How to kill TFPT”). Among the frozen ones: a fourth chiral generation (breaks $N_{\text{fam}} = 3$); a genuinely free fundamental constant that is not an E_8 -closure fixed point; a seam denominator $\neq 8$; α^{-1} away from the fixed-point root; a stable $\sin^2 \theta_{13}$ far from 0.0231 at $> 3\sigma$; cosmic birefringence converging away from $\beta \approx 0.2424^\circ$; a tensor-to-scalar $r \gtrsim 0.01$ (which kills the R^2 branch). The honest residual above all kill tests is exactly three named interfaces — the seam continuum theorem (SEAM.EQUIV.01, now reduced to a cited commuting-projector result,

[[verification/v424_seam_lto_rp_reduction.py](#)]/[[verification/v426_seam_lto_rp_kms.py](#)]), the metrology unit v_{geo} (theorem-forbidden), and the typed F_{transfer} interfaces (firewalled).

10 Summary — the layered defense

Layer	What it defends against	Artefact
Status calculus	a [C] claim read as [E]	ledger + audit_sync.py
No free pattern + reverse audit	cherry-picking, decorative identities	v305/v100/E8.REVERSE.AUDIT.01
Over-determination map	rhetorical “everything multiplies” (incl. <i>our own</i> miscount)	v427/v428
Firewall + No-Unit	external physics sold as compiler output; a faked scale	v187/v213/v153/v384
Frozen registry + null model	moving goalposts; chance hits	v84/v100/v375
Two independent paths	a shared implementation bug	Wolfram + Lean
Red team	self-serving framing	tfpt_5_redteam

The claim defended here is deliberately narrow: *coincidence is an expensive explanation of the discrete core, and exact compiler closure is never silently upgraded to physical closure*. That is the whole job of this companion.