

Universal Collapse Theory as a Coherence-First Research Program

Architecture, Claims, and Falsification

Program Map · read after *Kernel First* · Structural Reference

Jeremy C. Jones

HoldingLight LLC — ORCID 0009-0007-2515-3774 — universalcollapse.com

Version: v1.0 (2026-06) | License: CC BY 4.0

DOI: 10.17605/OSF.IO/CFASB

What this document is

This is the program map. It places every part of the Universal Collapse Theory (UCT) corpus, states what kind of claim each part makes, and shows how each could fail. It is a structural reference, not an argument: it does not defend the kernel-first reading and it does not develop any domain. Those jobs belong to other documents.

UCT offers three orientation surfaces, each answering a different question. *Kernel First* answers *how to read* the framework — the conceptual gateway explaining why UCT is kernel-first rather than primitive-first, and how its central term should not be reified. The *Reading Roadmap* (universalcollapse.com/roadmap) answers *where to start* — curated reading paths by audience. This document answers *what the program is* — its layers, its claim taxonomy, and its falsification structure. Read it after *Kernel First*, when orientation is in place and the question becomes one of structure.

This is not the best first document for a casual reader; it is the best first document for anyone asking whether UCT has a coherent program architecture — levels, standards, tests, mappings, audits, and failure modes — rather than a single undifferentiated claim.

What this map does not claim

This map does not prove the UCT kernel, validate the Law of Coherence, or establish any domain result by itself. It classifies the corpus, states the level of each claim, and identifies where falsification enters. The map is an accountability instrument — not evidence for the program, except insofar as it makes the program's commitments explicit.

1. The kernel, in brief

The minimal kernel is the tuple $(\Omega, K, C_t^K, x_t^*, R_t, S_t, T, U)$: a structured possibility space Ω , an active constraint set K , a constraint-conditioned collapse operator C_t^K , a realized resolution x_t^* , a record layer R_t , a residue term S_t , a record-time index T , and a constraint-update map U . The operator acts as

$$C_t^K : \Omega \rightarrow (x_t^*, R_t, S_t, \Omega_{t+1})$$

with constraints evolving through $K_{t+1} = U(K_t, x_t^*, R_t, S_t)$. In plain language: what could occur, under what constraints, resolves into what actuality, leaves what records and residue, and changes what can occur next.

Two conventions are load-bearing throughout the corpus. *Collapse* names the operation; *resolution* names its achieved result — reserved and never interchanged, which is what keeps collapse from being read as a substance. And the kernel is *substrate-neutral*: the same eight-place structure is specialized to each domain by restricting Ω , K , R , and the remaining terms, without privileging any one substrate as the base of the others. T denotes record depth, $T = \sum R_i$ — the count of accumulated record layers, not a numerical sum of their contents. The full postulates this notation compresses are fixed in WP01; the anti-reification reading is developed in Kernel First.

2. The claim hierarchy: law, method, and mapping

Not every part of UCT claims the same kind of thing, and conflating the kinds is the fastest route to misreading the program — dismissing a tool because a law is unproven, or crediting a law because a tool works. The corpus distinguishes three claim levels, and a reader should always know which level a statement occupies.

Level	What it claims	How it is judged
Level 1 <i>Formal kernel</i>	The kernel grammar itself: possibility, constraint, resolution, record, residue, update, and the operator relating them. A formal vocabulary for constrained actualization.	Internal coherence and definitional adequacy. Fixed in WP01; not an empirical claim on its own.
Level 2 <i>Structural-interpretive</i>	Readings of what the grammar means: collapse as constrained actualization rather than substance; mind, mathematics, and institutions as record-bearing rather than basal.	Argument and fit against alternatives. Defended in Kernel First and the domain papers; corrigible, not decisive on its own.
Level 3 <i>Domain discriminators</i>	Falsifiable predictions in a specific domain: the signature claims (S_1 , S_2 , S_3), their formal bounds, and the empirical tests that can come out against them.	Records, discriminators, and pre-specified falsifiers. This is where UCT can be wrong in the ordinary scientific sense.

Law, method, mapping

Across those levels, three working categories tell a reader what to expect from a given artifact. A **law-level** claim is a candidate regularity held under test — the Law of Coherence (collapse under constraint tends toward coherence), stated and assessed in WP05, which is deliberately held for last and written with its own drift test open. It is a claim on trial, not a settled result. A **method-level** claim is a tool that stands on its own terms — the Standards layer, the audit protocols, the reporting standards — usable, and falsifiable, whether or not the law holds. These shed value sideways: a working audit is a working audit even if the surrounding program fails. A **mapping** is the application of the kernel to a domain, and the corpus treats a mapping as metaphor *until* it yields a discriminator. The discipline is explicit: declared elegance is not world-contact; a mapping earns law- or method-status only by producing something that could have come out otherwise.

3. The architecture: layers of the program

The corpus is organized into structural layers, each doing a distinct job and each cross-referencing the others. They divide into two groups. The core research architecture carries the program’s evidentiary weight — the kernel, its domain plays, the hygiene layer, the standards that protect its integrity, and the formal–audit–empirical stack. The access and translation architecture carries the program outward to readers and into application — bridges, externalized-cognition work, the commercial wing, and entry points. The distinction matters: only the first group counts toward the research ledger. A thin program-level layer sits above both: the Tier 0 orientation gateway, and the architecture-and-governance documents — this map among them — that describe and hold the corpus accountable rather than adding to it. Section 7 gives the complete inventory with status and identifiers.

Core research architecture

Layer	What it does	Principal work	Tier
Formal kernel	Fixes the minimal recursive grammar: constraint shapes collapse, collapse writes records, records update constraints.	WP01 — Foundations of Collapse	WP
Law (capstone)	States and assesses the candidate Law of Coherence across domains. Held for last; written with its drift test open.	WP05 — Collapse as Law	WP
Domain plays	Works the kernel out in a single domain — a bounded game under the rules. Each stands in its field’s own vocabulary, with companions for depth.	WP02 (physics), WP03 (biology), WP04 (mind); T20 operating manuals	WP / T20
Ground-clearing	Hygiene layer. Clears a term wrongly elevated to a primitive so it cannot contaminate the rest. Zero UCT buy-in to read.	Primes 0–4 (Coherence, Randomness, Chaos, Intelligence, Nothingness)	T30
Standards / integrity	The architecture that keeps UCT from becoming an unfalsifiable vocabulary: persistence, update integrity, the conditions empiricism needs, and the outward compatibility door.	Records; Structuralization of Empiricism; Update Integrity Standard; CCR; Structuralization of AI	Standards
Formal bounds	Closed-form results for each signature under stated assumptions — the proved core of the discriminators.	Technical Notes TN-S ₁ , TN-S ₂ , TN-S ₃	TN
Audit protocols	Translates each bound into a field-deployable audit with a sharp model-conditional falsifier.	Methods-S ₁ , Methods-S ₂ , Methods-S ₃	Methods
Empirical demos	Runs the audits on real data. Narrow, falsifiable, theory-neutral where possible; failures recorded alongside successes.	Rice (S ₃ , biology); COGITATE (S ₂ , cognition); S-Signatures in AI; S3-RAG-01	T16

Access and translation architecture

Layer	What it does	Principal work	Tier
Bridges / doors	Standalone entry ramps. Each reaches a specific audience on a recognizable problem with no WP dependency.	Collapse Reframed; Schrödinger; Biological Faith Systems; Self the Ego Did Not Build; How Minds Resolve; Human Interface Laws; Kernel First	T15
Mind / CIM	Treats externalized, cumulative cognition — language, math, code, institutions, AI — as a record-bearing material that shapes later minds.	CIM Foundational; AI as Synthetic Collapse; AI in the Meaning Layer	T15
Applied / commercial	Carries the coherence-audit method into deployed AI systems as a structural diagnostic. Separate wing, separate clock.	AI Integrity Protocol (AIP) + reverse-audit reports	HL
Entry points	Accessible orientation without formal notation: the book, the essays, and runnable toy kits with reproducible figures and falsifiers.	Universal Collapse Theory (book); essays; Starter Packs (Physics, Biology, Mind)	—

These layers carry no evidentiary weight on their own. Applied and commercial artifacts translate coherence-audit methods into deployed settings; they are not evidence for the law-level program unless separately audited, versioned, and entered into the public empirical ledger under the same standards as the T16 demonstrations. Bridges and entry points are reader-facing translations of formal claims, and are judged by whether they clarify those claims faithfully, not by what they appear to assert on their own. The principle is uniform: only public, auditable, versioned research artifacts count toward the ledger.

How each layer can fail

The map states not only what each layer does but how each could fail — the per-layer companion to the claim-level and program-level falsification developed in Section 5. A layer in good standing is one whose characteristic failure has been looked for and not found.

Layer	What failure looks like
Formal kernel	Internal inconsistency, ambiguous primitives, or definitions that do not port across domains.
Structural interpretation	A rival reading explains the same corpus with fewer commitments.
Domain mapping	No discriminator emerges; the mapping remains metaphor.
Technical Note	The bound fails under its own stated assumptions.
Methods paper	The audit cannot separate the target signature from confounds.
Empirical demo	The falsifier fires, the result is absent, or the effect fails replication.
Standards layer	The ledger is incomplete, failures go unrecorded, or update rules are applied selectively.
Bridges	A reader-facing analogy distorts rather than clarifies the formal claim.
Applied / commercial	Diagnostic use fails external audit, or is mistaken for evidence for the law.

4. The signature stack: the worked spine

The program’s empirical core is a single repeated structure. Each of the three portable signatures is realized as a vertical column — a formal bound proved under stated assumptions (Technical Note), an audit protocol that translates the bound into a field test with a sharp falsifier (Methods paper), and at least one empirical run on real data (T16). The same three signatures are then exercised across three substrates — biology, cognition, and artificial systems — giving a worked matrix rather than a single demonstration. This is the structure a reader should follow top to bottom to see the framework do its work.

Signature	Formal bound (TN)	Audit protocol (Methods)	Empirical run (T16)
S₁ redundancy → consensus	Exponential consensus bound — independent records force agreement at a Chernoff rate.	Agreement-curve diagnostic — genuine vs. correlated redundancy.	Tested within the AI signature study (consensus channel).
S₂ neutrality → delay	Expected resolution-time bound — weak bias lengthens resolution; maximal at zero drift (drift-diffusion).	Latency-curve diagnostic — four-axis falsifier plus a fifth confound case.	COGITATE iEEG — prediction A supported ($p = 0.007$); B not supported; C inconclusive.
S₃ constraint sweep → hysteresis	Loop-area lemma — $A(R) = 4\theta_0 + 4\alpha R$; records amplify hysteresis, they do not create it.	Loop-scaling diagnostic — record-driven vs. confound-induced path dependence.	Rice transcriptome ($p \approx 5 \times 10^{-5}$, reproducible packet); S3-RAG-01 (bounded null over 420 classifications).

Where each signature breaks

A worked spine has to show not only that each signature exists but where it would fail. The bounds above hold only under stated assumptions, and each carries a sharp falsifier — a specific observation that would count against it. The companion table records both, at map level; the load-bearing detail is in the Technical Notes and Methods papers.

Signature	Load-bearing assumptions	Sharp falsifier
S₁	Conditionally independent records, an independently specified channel/redundancy model, and a controlled correlation structure.	The agreement curve fails the predicted (Chernoff-rate) decay under stated independence, or correlated pseudo-redundancy accounts for the consensus.
S₂	A drift-diffusion model with fixed boundaries, an independently measured bias/asymmetry, and controlled confounds (non-decision time, boundary shifts).	The latency curve does not peak near neutrality, or departs from the predicted $E[\tau]$ shape under the stated model and audited asymmetry.
S₃	A record-state parameterization, a rate-audited (quasi-static) constraint sweep, and confounds separable from record coupling.	The quasi-static loop area fails the predicted R-dependent scaling under a valid sweep, or path dependence is fully explained by rate lag.

Reading the matrix

Read it three ways: *down a column* for proof → protocol → data; *across a row* for the same logical layer across signatures; *across substrates* for the portability claim. The empirical column records failures and nulls alongside positive results, which is what gives the surviving results their weight.

5. Claims and falsification

A coherence-first program that says “look for structure” must be unusually explicit about where it can fail, or it reads as unfalsifiable. UCT distributes its falsification across three levels: the individual claim, the integrity architecture that disciplines the whole, and the program as a Lakatosian research program.

5.1 Per-claim falsification

Falsification is written into the artifacts, not deferred to a separate promise. Every Technical Note states an acceptance criterion and a rejection criterion. Every Methods paper is built around a sharp model-conditional falsifier — the comparison of an observed curve against the scaling law an independently specified model predicts. Every T16 paper carries pre-specified falsifier conditions and classifies each result as observed, partial, absent, inconclusive, or inaccessible, with no obligation for any result to come out positive. The Structuralization of Empiricism adds five Level-3 divergence claims with explicit local failure conditions and a global portability challenge. A reader who wants to know how a given claim could be wrong can read its falsifier directly off the paper that makes it.

5.2 The integrity architecture

The Standards layer is the machinery that keeps the framework corrigible. Its four documents do four distinct jobs, and the roadmap places them separately rather than as a single block.

Standards document	Job it does
<i>Records Across Nature, Life, and Mind</i>	Defines the persistence layer: what a record is, and the collapse → record → update cycle that makes resolution cumulative. Without records there is no objectivity, inheritance, or learning.
<i>The Structuralization of Empiricism</i>	Specifies the conditions under which empirical practice converges rather than drifting into pseudo-consensus or frozen paradigm. Carries the five divergence claims and the global portability challenge — the framework’s own exposure to failure.
<i>Update Integrity Standard (UIS)</i>	Names the ways the update loop breaks — record falsification, pseudo-redundancy, constraint freezing, selective update, coercive agreement, discriminator-free coherence, identity binding — with a repair protocol for each, and supplies the Empirical Ledger template for per-domain tracking.
<i>Coherence Compatibility Reference (CCR)</i>	The one outward-facing Standards document: a self-specification any external framework can run to test compatibility with coherence-first inquiry. The instrument is published; verdicts about named frameworks are not.

The UIS Empirical Ledger is load-bearing for what follows. It is the standing instrument by which audit outcomes — wins *and* failures — are recorded per domain, which is precisely what the program-level evaluation below depends on.

5.3 Program-level evaluation

The program as a whole is *not* falsified by any single result, and it does not carry a stated threshold whose breach ends it. That would be a category error — a research “programme” in Lakatos’s sense (his term) is not refuted by a failed prediction but is judged, over time, as *progressive* or *degenerating*: progressive when it keeps making new domains structurally legible and yielding fresh discriminators; degenerating when it survives only by relabeling what it cannot predict and bolting on exceptions to save its core. That verdict is empirical and longitudinal, read off an accumulating track record — not a philosophical proof, and not a condition statable in advance.

This places a real obligation on the program, and it is the one obligation that *can* be discharged now: honest bookkeeping of every audit outcome, so the eventual verdict is read off a complete ledger rather than a curated subset. Selectively remembering successes — reporting the favorable results and quietly dropping the null and failed ones — would make “progressive” a curated success narrative rather than a finding. The guard is the standing commitment to record the failures with the wins: the COGITATE B-failure and the S3-RAG-01 null are published as first-class outcomes, and the UIS Empirical Ledger is the instrument that keeps the full account. The willingness to record what came out against the framework is what makes the surviving results mean anything.

To keep that judgment from staying vague, the ledger is reviewed at release milestones — each review recording new discriminators, confirmations, nulls, failures, assumption changes, and any ad hoc repairs made to preserve a claim. The program reads as *progressive* when new discriminators and successful cross-domain transfers accumulate without a rising exception load, and as *degenerating* when negative outcomes are absorbed only by narrowing assumptions, relabeling results, or removing tests from view. This is a commitment to visible accounting, not a single-failure threshold: it makes the longitudinal verdict legible without pretending it can be triggered by any one result.

What would count against UCT

At the claim level: a signature’s bound failing where its assumptions hold, an audit’s falsifier firing, or a domain mapping yielding no discriminator that could have come out otherwise.

At the program level: a sustained pattern in which new mappings stop producing records, discriminators, or updateable results — in which the kernel is preserved only by ad hoc exception rather than by making new domains legible. No single failure triggers this; the degenerating trend, read off the full ledger, is the signal.

6. How to read the corpus

The library is not linear, and no reader needs to traverse all of it. Three doors serve three different needs, and a reader should enter through the one that matches the question they are bringing.

- To learn how to read UCT without misreading its central term — start with *Kernel First*. It is the orientation gateway: what UCT claims, what it does not, and why it is kernel-first rather than primitive-first.
- To find a path suited to a particular field — use the *Reading Roadmap* at universalcollapse.com/roadmap, which gives curated sequences for the general reader, physicist, biologist, cognitive scientist, philosopher, and methodology audiences.
- To see the program’s structure — its layers, claim taxonomy, and falsification architecture — use *this document*. It is the map, not the argument.

For the fastest single demonstration of what the framework does, the Schrödinger paper (*We Are Never Having a Mathematical Experience*) takes a problem everyone recognizes and dissolves it by separating a property of the model from a property of the world — no formalism, no prior exposure required. For the empirical spine, follow the signature stack in Section 4 down any one column. For the conceptual front gate, read any one of the Primes against your own priors; none requires accepting UCT to evaluate.

7. Complete corpus inventory

The current state of the corpus, by layer. Status and identifiers follow the canonical DOI registry; OSF DOIs are the persistent identifiers of record, with PhilArchive serving reading and OSF serving reproducibility depth. Where a DOI is not yet shown, the work is drafted or awaiting deposit. This inventory supersedes the layer tables of the internal System Map.

Status legend

Status	Meaning
Live	Public and citable; DOI deposited.
Published	Released through a formal channel (e.g. the book, with ISBN).
Mint pending	Complete or release-ready, awaiting DOI deposit.
Drafted	Manuscript exists but is not yet release-ready.
Release gated	Intentionally withheld pending a stated dependency or review condition.
Held for last	Deliberately deferred until dependent layers mature.
In development	Actively being written; shell exists.
Gateway	Orientation artifact, not a law- or method-level paper.

Orientation (Tier 0)

Work	Status	Persistent identifier
Kernel First — Collapse Without Reification (gateway)	Gateway; v1.0	10.17605/OSF.IO/6RZQ2

The reader’s door — how UCT should be read before the domain papers develop the kernel. Further orientation pieces (e.g. The Tether) are in progress.

White papers (the spine)

Work	Status	Persistent identifier
WP01 — Foundations of Collapse (v2.0)	Live	10.17605/OSF.IO/VZ836
WP02 — Collapse in Physics	Live	10.17605/OSF.IO/3NYMP
WP03 — Biological Collapse (v1.0)	Live	10.17605/OSF.IO/5SZ86
WP04 — Conscious Collapse	In development	—
WP05 — Collapse as Law (capstone)	Held for last	—

Standards layer (integrity architecture)

Work	Status	Persistent identifier
Records Across Nature, Life, and Mind (v2.0)	Live	10.17605/OSF.IO/7H6DY
The Structuralization of Empiricism (v1.0)	Live	10.17605/OSF.IO/J4GZ9
Update Integrity Standard — UIS (v1.0)	Live	10.17605/OSF.IO/DWM29
The Structuralization of AI	Mint pending	—
Coherence Compatibility Reference — CCR (outward-facing)	Drafted; release gated	—

Signature stack — formal bounds and audit protocols

Work	Status	Persistent identifier
TN-S ₁ — Objectivity from Records (consensus bound)	Live	10.17605/OSF.IO/6M7N3
TN-S ₂ — Neutrality Delays Resolution (time bound)	Live	10.17605/OSF.IO/6WRQV
TN-S ₃ — Records Amplify Hysteresis (loop-area lemma)	Live	10.17605/OSF.IO/QJMSZ
Methods-S ₁ — Auditing Independence	Live	10.17605/OSF.IO/7U8SK
Methods-S ₂ — Auditing Constraint Asymmetry	Live	10.17605/OSF.IO/HRKWT
Methods-S ₃ — Auditing Record State	Live	10.17605/OSF.IO/CQGTD

Empirical demonstrations (T16)

Work	Status	Persistent identifier
Bio Constraint Sweep — Rice (S ₃)	Live	10.17605/OSF.IO/KZ8TP
COGITATE — Constraint-Dependent Perceptual Resolution (S ₂)	Live	10.17605/OSF.IO/MXYU2
S-Signatures in Deployed AI Systems (parent)	Live	10.17605/OSF.IO/JPXCU
S3-RAG-01 — Retrieval-Channel Hysteresis (controlled RAG)	Live	10.17605/OSF.IO/5QMVS

Ground-clearing — Primes (T30)

Work	Status	Persistent identifier
Prime 0 — Coherence	Live	10.17605/OSF.IO/J2XSQ
Prime 1 — Randomness	Live	10.17605/OSF.IO/Y678R

Work	Status	Persistent identifier
Prime 2 — Chaos	Live	10.17605/OSF.IO/A6EJN
Prime 3 — Intelligence	Live	10.17605/OSF.IO/RN3TH
Prime 4 — Nothingness	Live	10.17605/OSF.IO/YHQ5F

The five Primes are live, minted as a set; each carries its own DOI and stands alone — cite the individual paper.

Bridges, mind/CIM, and entry points

Work	Status	Persistent identifier
Collapse Reframed	Live	10.17605/OSF.IO/PY5V3
We Are Never Having a Mathematical Experience (Schrödinger)	Live	10.17605/OSF.IO/GPYWJ
Biological Faith Systems	Live	10.17605/OSF.IO/D37Q5
Self, the Ego Did Not Build	Live	10.17605/OSF.IO/ZGRD4
How Minds Resolve (FRLB01)	Drafted	—
Human Interface Laws	Drafted	—
CIM Foundational (v1.0)	Live	10.17605/OSF.IO/7URMK
AI as Synthetic Collapse	Mint pending	—
Structural Biology Operating Manual (v1.0)	Live	10.17605/OSF.IO/CPKNX
Starter Packs — Bundle / Physics / Biology / Mind	Live	UWRS3 / XRE7B / M8S39 / QCWN8
Universal Collapse Theory (book)	Published	ISBN 978-1-969095-01-6

Architecture & Governance (program-level)

Work	Status	Persistent identifier
Program Map — this document	Live	10.17605/OSF.IO/CFASB
Failure Modes and Falsification Standards	Live	10.17605/OSF.IO/TN7Z3
Two Worked Demonstrations (research + applied stacks)	Live	10.17605/OSF.IO/Z7HY2

Documents about the program as a whole — its architecture, its failure conditions, and its stacks in action. They describe and govern the corpus rather than developing or applying the kernel, and do not themselves count toward the research ledger; they deposit as a set.

Document status and use

This is a living reference and the canonical successor to the internal System Map. It is versioned independently of the papers it indexes: housekeeping updates increment the minor version; a structural reorganization increments the major version. Individual papers move on their own cadence and may have advanced beyond the versions shown here; this map is a point-in-time snapshot of the corpus, not a live index of it. The authoritative source for identifiers and deposit status remains the UCT Canonical DOI Registry; where this map and the registry diverge, the registry wins. A condensed, public-facing version of this map is mirrored at the roadmap address below.

UCT Library

This document is part of the Universal Collapse Theory library maintained by Jeremy C. Jones and HoldingLight LLC. Roadmap: universalcollapse.com/roadmap

AI Disclosure

AI tools were used to assist with manuscript preparation, drafting, organization, and editorial refinement. The underlying theory, structural decisions, analysis, and conclusions are the author's own.

Suggested Citation

Jones, J. C. (2026). Universal Collapse Theory as a Coherence-First Research Program: Architecture, Claims, and Falsification (v1.0). HoldingLight LLC. <https://doi.org/10.17605/OSF.IO/CFASB>

© 2026 Jeremy C. Jones — HoldingLight LLC · CC BY 4.0