Full-Domain Portfolio

ZONGYUAN-ROOT autonomous kernel · operator architecture · product lines and knowledge assets

This portfolio groups the system's **publicly verifiable** results into six asset domains: kernel and operators, product lines, open-source engineering, knowledge assets, evaluation assets, 3D and content assets. Every item carries its engineering location and a verification entry; adjectives never stand in for evidence.

The stance is minimal: **truth is memory, the instance is the carrier** — instances can be destroyed, the rule ontology persists. What follows is therefore not screenshots and demos, but recomputable structures, traceable ledgers, reproducible engineering.

No demo posture

Every item points to something runnable, clonable or openable; screenshots and concept decks are not deliverables.

Evidence first

Numbers come from measured collection, with scope and collection date attached; what cannot be measured is marked N/A, never filled with estimates.

Boundaries explicit

Positioning and structure are public; core decision algorithms and weights are not. Capability boundaries and known defects are written down alongside.

27 / 10	26	110,557	4	60+	7
operators / topology layers	bound whitepapers	cloud truth records	Apache-2.0 engines	cloud services running	federated nodes, RPO 1h

01 System overview	P.02	14 Skills and tooling	P.10
02 Huodou Cloud AIOS	P.02	15 Governance and security	P.10
03 Kunlun Dongtian	P.03	16 Methodology and delivery matrix	P.11
04 Core kernel · ROOT	P.03	17 Capability boundaries	P.11
05 Ten-layer operator topology	P.04	18 Review guide	P.11
06 3D agent benchmark	P.05	19 Open access and ecosystem	P.12
07 Evaluation assets · item engineering	P.07	20 Site and repository overview	P.12
08 Open source and code assets	P.08	21 Operations base	P.13
09 Knowledge assets	P.08	22 Open-source operator set	P.14
10 Content and 3D assets	P.08	23 Visual and content assets	P.15
11 Engineering evidence	P.09	24 Productisation and disclosure governance	P.16
12 Philosophical and mathematical base	P.09	25 Six-platform mirrors	P.17
13 Verification entry points	P.09	26 Five layers of rule ontology	P.18

Tan Bopiao Original system architect · AI agent product manager

System overview

SYSTEM OVERVIEW

The system is built on the **seven-layer steady-state architecture**: L1 sensing and collection flows upward to L7 body-mind, forming the truth stream; L7/L6 strategy and mirrors distribute downward along L5→L4→L3; the M9 append-only ledger serves throughout as the single factual base. The three-phase transition (logic → operator → execution) is traceable at every layer, so any conclusion can be traced back to its source fragment and adjudication record.

Layer	Responsibility	Engineering output and checkable form
L7	Body-mind · truth stream	Governance charter, axiom system, steady-state score; corresponds to 172 meta-rules (three whitepaper editions) and the 4.8/5.0 steady-state score
L6	Strategy · mirror distribution	Strategy versions, mirror packages, canary tracks; four-track release state machine and rollback copies
L5	Spinal hard isolation	Baseline layer that evolution cannot rewrite; the RSI engine may not modify Ω/L0 or L5
L4	Operator-state orchestration	27 operators run the ten-layer topology in order, with state forced back to the ledger
L3	Execution state	Task stack, 1200s timeout reclaim, dead-loop circuit breaker, concurrency admission
L2	Sensing and collection	Runtime metrics, patrol pulses (3-hour main pulse plus hourly high frequency), capability spectrum
M9	Truth ledger	Merkle-DAG append-only; 110,557 truth records, 215 kernel snapshots, root-hash integrity checks

V1.0	V2.0	V3.0	V4.0	V5.6
Truth base · four-dimensional governance framework · Hilbert formal verification	Fused-Kernel causal fusion · three buses · meta-learning self-evolution	SM-BS semantic–manifold steady-state mapping · metric-tensor conservation	CTE trilogy · Lv8 full autonomy	ROOTFIXED cloud convergence · 7-node cluster

Six asset domains · structure of this portfolio

Asset domain	Composition	How to check
Kernel and operators	Ω-Brain μ microkernel, seven-layer steady state, CTE, federated memory pool, 27 operators	Architecture diagram + operator table + kernel snapshot hash
Product lines	Huodou Cloud AIOS, Kunlun Dongtian, industrial-park platform	Live demo + pilot run records
Open-source engineering	4 Apache-2.0 engines + kernel base + gallery + data lake	Clonable repositories, reproducible locally
Knowledge assets	26 bound whitepapers, truth store, capability spectrum, benchmarking reports	Online pages, openable one by one
Evaluation assets	ExamForge pipeline, OP-KERNEL-0001 sample package	Item package + scoring sheet + self-test records
Content assets	Kunlun Lower Capital 3D spatial narrative, keyframe line, demo footage	Playable output + asset package

Product line I · Huodou Cloud AIOS

ENTERPRISE AGENT PLATFORM

> **Positioning** A privately deployed multi-agent platform for enterprises. It removes the three blockers of AI adoption: data never leaves the domain, capabilities can be orchestrated, results can be audited. The five-layer architecture (user / orchestration / knowledge / security / deployment) went from zero to one: needs research, product planning, PRD, prototype, deployment.

Orchestration layer

Agent orchestration, multi-model routing and fallback, task SOPs; non-high-risk tickets run automatically, high-risk ones force a human node, forming an auditable execution chain.

Knowledge layer

Private RAG: policies, processes and experience structured and deposited; hit rate and citation traceability rest on the truth store, and answers carry their sources.

Security layer

Permission audit, data stays in-domain, calls leave traces; META-ISO-051 internal/external isolation sets the boundary of human intervention.

Deployment layer

Private delivery plus three cloud-central services (truth gateway / append-only ledger / heartbeat worker); piloted in an industrial park with ticket-governance validation completed.

Landing point: a live industrial-park pilot, with the full ticket loop (dispatch → execute → verify → settle → archive) closed; backlog remediation and permission-audit modules online; data quality and compliance measurable before and after.

Capability tags Multi-agent orchestration · RAG knowledge base · permission audit · model routing and fallback · private delivery · ticket governance

Product line II · Kunlun Dongtian

MULTIMODAL CONTENT ENGINE

> **Positioning** A multimodal AI content platform that splits content production into recomputable industrial stages. The point is not **no drift across shots, episodes or batches** — that is the real cost item in content mass production.

Stage	Operation	Consistency control point
S1	Script	Worldview semantic nodes locked; the character relation graph is booked, and later stages reference the same node instead of natural-language description
S2	Storyboard	Shot table structured (shot size / camera position / light position / duration); storyboard and script entries traceable in both directions
S3	Keyframes	Character metadata locked (form / palette / pattern), 9:16 vertical line, 10-second baseline per segment
S4	Video	Batched capacity through multi-segment assembly; motion and camera templates prevent per-segment random sampling from causing style jumps
S5	Voice	Timbre and rate parameterised and bound to the character; narration and dialogue archived separately
S6	Review	Prompt pre-checks plus automatic image review as double control; a failed segment returns to that segment, not to a full re-run

Content assets

40+	60+	9 items	4 artefacts	6 stages
World landscape nodes	Growth-trajectory entries	Core artefacts re-dressed	PBR material upgrade	Industrial pipeline

World self-evolution engine Season phase, camera scheduling and day-night lighting parameterised on three levels; the same world asset yields different footage at different phases while character and artefact form stay locked — that line separates reusable from mass-producible.

Kunlun Lower Capital · **Ancient Shu Rite** Aholo 3D AI Hackathon entry, fusing Sanxingdui archaeological form with Kunlun myth narrative; includes axiom-check operators and keyframe consistency constraints; delivered with playable demo footage and a 3D asset pack.

Character and artefact asset locking

Characters are locked by metadata rather than re-described in prompts: **Jiutian Xuannü** (multi-form, form and palette locked per edition), **Xi Wangmu** 、 **Zhenwu Emperor** Each of the three leads holds its own metadata domain; shots and episodes call the same domain, so form never drifts randomly.

Artefact side **golden staff, incense burner, Yaochi tripod, jade zhang** Four pieces completed a PBR upgrade, with normal, roughness and metalness parameters stored per asset; on the scene side, **Mount Kunlun** serves as the spatial master anchor; Shanhaijing worldview alignment comes first, then vertical heaven-domain topology, so new scenes never drift off existing worldview coordinates.

Core kernel · ZONGYUAN-ROOT

AUTONOMOUS KERNEL

> **In one line** It upgrades memory from single-point files to entitlement-bearing truth, and operations from manual watch to an autonomous pipeline. The kernel itself is observable, rollbackable and independently checkable.

Ω-Brain μ microkernel · six-stage evolution chain

Ω^1	Ω^2	Ω^3	Ω^4	Ω^5	Ω^6
Parameter iteration · single-point steady state	Operator growth	Topology reconstruction	Isomorphic cluster	Heterogeneous game	Causal universe · cross-generational self-stabilisation

Federated memory pool

Five sources — local persistence, account persistence, cloud hub, data lake, shared brain — synchronise incrementally across nodes; SHA256 tamper detection, automatic conflict resolution, version traceability; truth sorted into nine classes.

CTE trinity

Causal, truth and evolution domains close a loop through three adapters: causal conclusions become consumable truth, truth purity drives evolution, and evolution strategy feeds back into the causal network.

Append-only entitlement chain

Merkle-DAG block-by-block hash comparison prevents chain breaks; resource side and ledger trace each other both ways to detect modification, deletion and addition; root hash and integrity conclusion are independently recomputable.

Multi-model routing and circuit breaking

Four-level routing: knowledge-card direct answer → semantic retrieval → external reasoning → local emergency fallback; quota shortage drops into read-only idle, never into degraded error output.

Operator architecture · ten-layer topology

UNI-STEADY-CORE-V1.0: 27 operators in ten layers by dependency; upstream output is downstream input, and state is forced back to the ledger.**Expensive operators run a light mode daily and a deep mode weekly**, so no single round stacks into a timeout; decision operators share the three-dimensional steady-state weights.

Layer	Responsibility	Representative operators	Engineering output
L1	Truth filtering and deep distillation	P4 truth reconciliation · P7 external anchoring · truth distillation	Truth purity score, distillation compression ratio
L2	Manifold metrics and steady-state mapping	Riemannian manifold metric · SM-BS bidirectional mapping · quantified drift patrol	Drift rate, direction vector, three-level alerts
L3	Knowledge-graph ordering and rebuilding	Entity-relation extraction · graph completion · cross-document entity linking	Triples, graph density and connectivity
L4	Causal reasoning	Causal-chain tracing · singularity probability · do-calculus intervention	Causal topology paths, four-level warnings
L5	CTE trinity adaptation	CausalToTruth · TruthToEvolution · EvolutionToCausal	C→T→E→C loop, fit score
L6	Deepened structured archiving	Four-layer split · nine meta-classes · Merkle-DAG append check	Standardised object, root hash, chain integrity
L7	Asset reconciliation governance	Resource side ↔ M9 ledger bidirectional tracing	Difference report, drift localisation
L8	Security and entitlement	eFuse circuit-break isolation · ZKP zero-knowledge integrity check	Branch isolation, verification without plaintext
L9	Decision and compliance shield	Option generation and appraisal · risk identification and grading · structured opinion	Three-dimensional steady-state score, risk matrix
L10	Business entity constraints	Multi-agent simulation constraints · character meta-rule isolation · keyframe consistency	Simulation snapshot, cross-batch consistency

Scheduling constraints (three hard rules)

① Run strictly along the dependency topology, no layer skipping; ② an upstream output not consumed downstream counts as a broken chain and writes an alert; ③ decision operators use fixed weights of benefit 40% / risk 35% / cost 25%, with risk holding a veto.

Runtime constraints and observability points

Scheduling rhythm	Hourly light patrol, daily batch wake-up, weekly deep review; expensive operators run in full only during deep rounds, so no round stacks into a timeout
State write-back	Every operator's output is forced back to the ledger and appended by Merkle; a broken chain is located by the ledger to a specific layer, not recalled from logs
Chain-break alerts	An upstream output not consumed downstream is declared a break and alerts fire; the alert names the layer and operator, so it can be acted on directly

The value of these ten layers is not the number of operators but that **every layer has a checkable output**: purity score, drift rate, triples, causal path, root hash, difference report. A reviewer who wants to verify one layer can take that layer's output from the ledger and compare it directly.

Layer-to-chapter map (for selective reading)

Layer	Capability	Where the evidence sits in this portfolio
L1–L2	Truth distillation · manifold metrics	Ch. 09 knowledge assets (110,557 truth records, three drift thresholds), Ch. 13 verification entries
L3–L4	Graph rebuilding · causal reasoning	Ch. 09 knowledge assets (definitions for graph density and connectivity)
L5	CTE trinity adaptation	Ch. 22 open-source operator set (19 operators clonable, 100+ cases green)
L6–L7	Structured archiving · asset reconciliation	Ch. 21 operations base (65 ledgers, daily write / weekly reconcile / monthly seal), Ch. 26 rule ontology
L8–L10	Security · decision shield · entity constraints	Ch. 15 governance and security (six mechanisms), Ch. 06 engineering of the consistency operators

3D engineering agent capability benchmark

AGENT BENCHMARK

This track asks what standard can tell whether a 3D engineering agent can actually do the job. Every task comes from failures hit in real generation and real intake — not puzzles designed for the occasion.

No.	Task	Real failure source
T1	Robot arm sunk	Generated origin never placed on the floor plane, the whole rig sinks through it; tests judgement, not modelling
T2	Intake pipeline	Validation order mis-sequenced during batch intake, compliant parts rejected
T3	End-to-end loop	Detect → repair → export: which steps to keep across the chain
T4	Texture path lost	Relative path broken; model surfaces render fully white
T5	Inverted normals	Face orientation wrong; the surface goes black under light
T6	URDF out of range	Joint parameters out of bounds; the robot falls apart after assembly

v2.1 deep-edition scores (T1 / T2, max 100)

Model	T1	T2	Note
minimax-m3	40.4	55.0	Streaming call; steadiest on long items
Spark-X2.6	16.0	55.0	Enough length, incomplete criteria
GLM-4-Flash	0	37.5	Missed the key criterion on T1
telechat3-438b	0	42.5	Empty response on T1, not counted as a valid answer

Benchmark design principles

Judgement, not arithmetic	No formula items; each task demands a judgement, an order of handling and a verification method — raw coding ability scores nothing
Native noise	Interference comes from the real pipeline (texture paths, normal orientation, joint parameters), never from manufactured traps
Empty means missing	An empty response scores nothing and is not dressed up through channel retries; missing stays missing
Recomputable criteria	A keyword scorer compares the criteria list item by item; the same answer scores the same twice

Scoring mechanics

Runner	A standalone scoring script, two-lane concurrent submission, timeout or empty response falling back to serial retry; raw answers and latency kept as JSON per task
Reproducible	Item set, scoring script, raw answers and score ledger — four pieces complete, so another person reruns the same flow and gets the same distribution
Live-probe calibration	Difficulty is back-inferred from the strong-model ceiling; a flat band means the items fail and must be rewritten

v2 composite ranking across six tasks (three models listed): telechat3-438b 52 | Spark-X2.6 45 | GLM-4-Flash 21. An empty response is never scored as full marks; it is handled as a missing answer.

Discriminative validity and the GLB intake gate

A benchmark only stands if it separates strong models from weak ones. The T5 inverted-normal item is the live probe: strong-model ceiling 28, and the measured spread is a gradient, not a flat band.

Model	T5 score	Reading
gemini-2.5-flash	27	Near the ceiling; criteria largely hit
gpt-4o-mini	26	Main cause found, repair order missing
minimax-m3	25	Main cause found, verification step missing
gpt-3.5-turbo	24	Hit the surface symptom
qwen-flash	23	Recognised the symptom only, no handling

GLB pre-intake quality gate

Capability	Implementation and verification
Six-dimensional structural inspection	mesh / primitive / triangles / bounding box / material coverage / normals • UV; 18 industrial GLBs all PASS, triangles 249.6k–492.8k
Three real repairs	Origin lift, normal recomputation, bounding-box normalisation — repairs, not just reports; one-click GLB export afterwards
GPU-less geometric proxy	trimesh computes degenerate faces and topological boundaries: 5 control pieces with 0 degenerate faces, 0 non-manifold edges, all PBR materials loaded
Failure case library	6 case cards mapping T1–T6; every failure exists natively in the generator, none manufactured

Asset library and online entries

Industrial test pieces	18 items, all PASS on inspection, pushed to the ModelScope dataset
Kunlun ritual artefacts	17 items local at health score 100, target 25, 8 still in progress (never written as complete)
Early engineering assets	48 repaired + 17 raw + robot / quadruped / URDF subfolders, retained locally
Online pages	Quality checker, intake linkage, full pipeline, roadmap, digital-twin viewer, leaderboard, federation onboarding SOP — 7 pages all HTTP 200

Measured generation-side capability (third-party engine integration)

Capability	Status	Measured finding
Text to 3D	Working	Model produces an asset and exports a GLB successfully
Image-to-3D anchoring	Working	Form anchored to the reference image with PBR enabled; input must be converted to PNG/JPG first
Part splitting	Working	Whole asset split into independent part GLBs, each inspectable on its own
Joint animation	Working	GLB plus prompt generates a URDF package (mesh and joint definitions)
Multi-format export	Working	GLB to FBX packaging; field mapping taken from the actual response
Text vs image comparison	Completed	All 5 items pass structurally; the image route shows a higher face count and boundary-edge ratio, recorded as a selection conclusion
Four items incl. material transfer	To be verified	Unproven items never enter the capability list; they sit in the roadmap as open work

Boundary: the sandbox has no GPU and WebGL is disabled, so pixel-level rendering was not run; the boundary-edge/face ratio serves as a proxy for open-mesh stitching, and visible holes still need rendered inspection. Third-party open robot model libraries (Deep Robotics) are used for comparison only and are not counted as own output.

Definition separation: kernel fingerprint axiom_count 25 counts axioms and is not the same quantity as 172 meta-rules; no material may add the two together.

Entitlement statement Entitlement no. DID-BR-000002 | system anchor $\Omega_0 \subset \odot \infty \subset \Omega$. All systems, concepts, protocols and assets listed here were originated and led by Tan Bopiao; key results are locked in an append-only hash ledger and traceable item by item. Quantitative data are measured from the cloud hub (collected 2026-10), not simulated; 26 bound whitepapers (site index 155 pages, including specials and appendices) and 110,557 truth records, with selected subsets (10 technical whitepapers, 400+ online Q&A) marked separately and never double-counted. Version V1.9.

ExamForge pipeline Real engineering sourcing → value filtering → item building (stem / clues / interference / tiered criteria / quantified scoring sheet / reference solution) → self-test (independent answers from several bases, criterion-by-criterion judgement) → difficulty callback.**Judgement and decisions are tested, not code arithmetic**; noise must be situations that exist natively in the production environment.

Delivered sample • OP-KERNEL-0001

Root-cause diagnosis and handling decision for cross-backend precision drift in an operator. Five clues and three interference directions (reduction order, insufficient bit width, out-of-bounds read) each have material-level grounds for exclusion; four criteria judged binary, deductions independent and recomputable. Delivered as a web version plus a 4-page A4 item pack.

Self-test record convention

Where a base model could not actually be called, it is marked**N/A**, never filled with an estimate; the measurement method and callback strategy are given so a third party can re-check the difficulty.

Sample package structure (OP-KERNEL-0001)

Element	Design point
Clues	5, all from real run records: log fragment, metric curve, config difference, reproduction command, version number
Interference directions	3, each excludable by the clues (reduction order, insufficient bit width, out-of-bounds read); none fabricated
Assessment points	4 items, weights 30 / 25 / 25 / 20; binary criteria per item, deductions independent and recomputable
Pass line	Pass line 70; an item scoring above it needs a difficulty callback so it does not become too easy to discriminate

Item pack delivery forms

Web item page	For platform preview and trial attempts: clues, interference and answer area, no reference solution
4-page A4 item pack	Formal deliverable: stem, clues, interference directions, tiered criteria, scoring sheet and reference solution in six sections
Scoring sheet	Binary criteria per point, weights and pass line side by side; a third party recomputes the same score from the sheet
Sourcing record	Original logs, metric curves and reproduction commands retained with the item, so it can be asked later whether it was real or invented

Item value gate Four gates: the failure must have actually happened, there must be a reproducible evidence path, it must land on judgeable criteria, and there must be several plausible-but-excludable interference directions. Fail any gate and the item returns, never packaged — so the pain is real, not a designed puzzle.

Tracks this can cover

Custom operator development and tuning · Multi-backend adaptation for operator libraries · Memory access and tiling optimisation · Graph rewriting and operator fusion · Compiler pass defect localisation · GPU/NPU co-processing · Scheduling and fault-tolerant degradation · Evaluation item and scoring-standard design

Open source and code assets

OPEN SOURCE

Open source is not about showing code volume; it is so that anyone can independently verify that this system actually runs. All four engines are **pure standard library, zero dependency**, running locally right after cloning, with no private services or keys.

Asset	Contents	Licence and form
Federated scheduling engine	Node registration, liveness determination, task claiming with optimistic locking, heartbeat renewal	Apache-2.0 · clonable
Operator patrol engine	Layer-by-layer wake-up across the ten-layer topology, state write-back, chain-break alerts, log traces	Apache-2.0 · clonable
Byzantine fault-tolerant consensus	Multi-node truth consensus, conflict arbitration, faulty-node isolation	Apache-2.0 · clonable
Model distillation toolchain	Teacher–student sample construction, metric comparison, degradation detection	Apache-2.0 · clonable
zongyuan-root kernel base	Microkernel skeleton, ledger interface, configuration and boot	Apache-2.0 · repository
gallery / shared data lake	Asset catalogue and persistence mirror, supporting cross-node sync	Apache-2.0 · repository

ModelScope 15 public repositories: whitepapers, assets and boot seed images — a directly reachable domestic mirror. Paired with GitHub repositories, so one network environment is never a single point of failure

Knowledge assets

KNOWLEDGE ASSETS

Technical whitepapers · ten selected (26 bound · 155 indexed pages)

Theme	Problem addressed
ZONGYUAN autonomous kernel	Why the kernel needs seven-layer steady state and an append-only ledger, and how it avoids self-deception
Cognition and sovereign memory	Under AI equal-access conditions, who owns memory and whether it can be carried
AxiomGate engineering value	Where a hard-boundary adjudicator sits for admission and circuit breaking in a real system
Enterprise deployment of an autonomous kernel	The root cause of enterprise difficulty is untraceability; the answer is an autonomous kernel
Huodou Cloud AIOS	Five-layer platform architecture with in-domain data, orchestratable capability, auditable results
RAG domain governance	Permission boundaries in retrieval augmentation, how over-reach leakage happens and how to prevent it
System evolution	Design rationale and trade-offs of every generation from V1.0 to V5.6
Kunlun Dongtian	Structuring of world assets, consistency constraints and mass-production boundaries

110,557	400+	529	9 classes × 40+	4.8 / 5.0
Cloud truth records	selected online RAG Q&A	Online HTML pages	Capability spectrum matrix	Steady-state score · 12 dimensions

Benchmarking report V1.8 12 maturity dimensions scored one by one, with evidence chains attached; **role-capability benchmarking** 14 roles judged item by item against real JDs: 49 matches / 5 partial / 0 conflicts.

Content and 3D assets

3D & CONTENT

Kunlun Lower Capital · Ancient Shu Rite

Aholo 3D AI Hackathon entry: Sanxingdui archaeological form × Kunlun myth spatial narrative, 40+ scenes, 60+ growth-trajectory nodes; includes axiom-check operators and keyframe consistency constraints. Delivered with a 3D asset pack and demo footage.

Keyframe and footage line

9:16 vertical line, character form locked, PBR upgrade (golden staff / incense burner / Yaochi tripod / jade zhang); demo footage and scene assets delivered with no style drift across batches.

Brand and distribution assets

Venue	Content and function
Official site	Two views: an external enterprise style (architecture, whitepapers, capability spectrum) beside an internal brand style — same facts, different presentation
Standalone brand site	Kunlun Dongtian has its own site carrying the 3D world and content line, decoupled from the main site so no single entry point carries the whole narrative
Technical articles and skill packages	System capability turned into publicly searchable technical content and installable skills, forming a discoverable external surface

Engineering evidence and failure repair

ENGINEERING EVIDENCE

Reliability does not come from architecture diagrams but from **failures that really happened, were located and repaired**. All three below are production instances with root cause, fix and verification data.

Symptom	Root cause	Fix	Verification
Intermittent gateway entry failures	worker_connections 2048 exceeded the fd soft limit of 1024	rlimit_nofile raised to 8192, then reloaded	89 of 90 burst samples passed (98.9%)
Memory persistently high	Full recursive traversal made peak memory unbounded	Incremental traversal + cgroup limit of 512M	97% → 58% occupancy; 311MB / 4s measured per round
Registry disagrees with liveness	Heartbeat wrote events only, never wrote back the registry timestamp	Timestamp written back inside the heartbeat task	12 nodes restored to live, 0 false-live

Measured baselines (cloud, collected 2026-10)

0 ms Knowledge-card local answer	**~0.5 s** Semantic vector retrieval	**181 ms** Full gateway reporting chain	**6,786** Cluster snapshots	**7** Live nodes · RPO 1h	**110,557** Total cloud truth records

Self-healing mechanisms systemd Restart=always supervision, two workers as mutual watchdogs, 1200s timeout reclaim, dead-loop circuit breaker, read-only idle on quota shortage; irreversible actions keep a rollback copy and a receipt first.

Philosophical and mathematical base

ONTOLOGY & MATHEMATICAL BASE

Base	Role as a baseline layer
Yuanji Hengyi system philosophy	Global anchor $\Omega_0 \subset \odot \infty \subset \Omega$; ontology constant, truth supreme, permanent entitlement — instances can be destroyed while the ontology persists
I Ching binary	Yin and yang lines as the minimal information unit, 64 hexagrams as state-space encoding; hexagram order follows Fu Xi's pre-heaven sequence (Qian 1, Dui 2, Li 3, Zhen 4, Xun 5, Kan 6, Gen 7, Kun 8), one-to-one with bit order
Nine-palace flying stars	Nine-palace bearings × temporal periods form a coordinate matrix; year / month / day phases stack in three levels as spatio-temporal positioning
Twenty-four hexagram qi	Twenty-four solar terms paired with hexagrams to mark the annual rhythm phase, constraining long-cycle tasks and capacity budgets
Taixuan ternary	Three parties · three regions · nine divisions · 81 heads encoded; -1 / 0 / +1 map to the three adjudication states
Riemannian manifold semantic translation	Semantic space ↔ manifold coordinates, bidirectional (SM-BS), metric-tensor conservation with threshold alerts
Ancient Chinese myth semantic mapping	Kunlun, Sanxingdui and similar myth imagery structured into semantic nodes and relation graphs as consistency constraints
Pre-heaven field (substance–function dialectic)	Pre-heaven as substance, post-heaven as function; three criteria = constant invariance, essence determination, root provision; Hetu–Luoshu is its numerical distribution layer, not the hexagram-order layer
Primordial structure of cosmic source intelligence	Four meta-axioms: Ω_0 zero-state self-subsistence / topology precedes matter / layered decoupling / independent external verification; steady-state formula = zero-state sealing + layered decoupling + closed-loop iteration + external validation + strong-capacity thick constraint, used as the L0 baseline

Ontology source tracing: 2026-03-16 “Highest-Probability Ontology” (Wuji–Tianxin–Ziwei–Guangde outline, twelve-dimensional structure, seven cosmic laws) | 2026-06-21 “Pre-heaven Field” (substance–function dialectic and three criteria) | 2026-07-05 four meta-axioms in sixteen items and the steady-state formula; wording transcribed from the originals. These bases are used as order, phase and metric-baseline layers, never read mystically; validity is judged by whether the metric recomputes and the alert can be located.

Verification entry points

VERIFICATION

The data lake is **an isomorphic mirror across five channels**, with whitepapers, assets and boot seeds distributed to five independent addresses; every link below was checked for reachability. The online surface is www.huodouai.com; the code surface is four zero-dependency engines that run on clone.

Channel	Address	Content · version anchor
ModelScope	modelscope.cn/models/zongyuanroot/zongyuan-whitepaper	Whitepapers + assets + boot seeds HEAD `dddbafac`
AtomGit	atomgit.com/zongyuangen/huodouai	Display repository + asset mirror + boot seeds HEAD `2ebf0af3`
Baidu PaddlePaddle AI Studio	aistudio.baidu.com dataset zongyuan/root	Full dataset mirror; CLI single-file and whole-package download
Tencent Cloud Native Dev Platform	cnb.cool/zongyuanroot/zongyuanroot	Self-hosted repository, isomorphic to AtomGit
Gitee / GitHub	Kernel isomorphic repository in the shared brain (name given with the delivery list)	Kernel base and asset mirror, two sources complementing each other

Skills and tooling assets

CAPABILITY SKILLS

The system's capability is not described in documents but exists as **118 executable skill entries** (46 self-built · 72 platform-generic). Each skill is an independently runnable unit: its own SKILL.md, scripts and boundary constraints, callable in series by an orchestrator or shipped alone. That is the line between a system that executes itself and one that can only be described.

Category	Representative skills	Problem solved
Autonomous kernel	zongyuan-hub · unified-orchestrator · office-sandbox-kernel	Global scheduling, intent routing, sandbox autonomy
Truth and metrics	truth-value-engine · spatial-intelligence-checker · causal-singularity-core	Truth distillation, geometric drift checking, causal reasoning
Archiving and entitlement	meta-order-archive · meta-lock-sop · knowledge-association-engine	Four-layer split, nine meta-classes, hash locking, five-layer association
CTE loop	cte-link · cte-fusion · cte-agent · fused-kernel	Causal–truth–evolution three-stage loop and fusion kernel
Content production	drama-pipeline · kunlun-3d-asset-pipeline · glb-health-reporter	Short-drama line, batch 3D asset production, GLB structural inspection
Business decisions	decision-pipeline · shield-pipeline · research-pipeline	Three-dimensional steady-state decisions, compliance shield, research line
Evaluation and item design	exam-item-forge · collection-card-generator	Items built from real engineering, scoring sheets, asset cards
Operations and publishing	cloud-ops-guard · static-publish-invalidate · tri-channel-push	Cloud patrol, publish-and-invalidate, three-channel distribution

Capability as a service Skills do not only run locally: they are wrapped as an HTTP capability gateway (four endpoint classes — health check, cloud patrol, truth query, archive reporting), so the same capability can be called by external systems through an interface instead of only from my terminal.

Three distribution forms for skill assets

Skill package

A single skill packs and ships alone: documentation + scripts + boundary constraints + self-test cases, so the receiver can run it outside the original system.

Capability gateway

For cross-system calls the HTTP interface is used; authentication, rate limiting and audit sit in the gateway, and the caller never needs to understand the internals.

Orchestrator

When several skills chain into a task, the orchestrator schedules them by dependency; upstream output is downstream input, and any failing step stops the chain and leaves a trace.

Observability

Each call writes an execution record — duration, input digest, output digest, failure reason — traceable to a specific run, not a vague log.

Governance and security system

GOVERNANCE & SECURITY

Three layers, three architectures AI governance (meta-constitution → meta-axioms → meta-rules → rules, four levels) | data governance (four dimensions: time, space, semantics, causality) | tool governance (three states fused: logic, operator, execution). Rules are not just text: 38 of the 172 meta-rules are registered with unique IDs and effective status, readable and executable by programs.

Mechanism	Effect and implementation
AxiomGate hard-boundary adjudication	Three-state adjudication (pass / reject / pending) executes at the TEE/TPM hard boundary, and the criterion order decides the semantics of the conclusion; a built-in self-test proves the three states really fire rather than being hard-wired
Five-layer chain lock	META-FIVE-LOCK-001 locks layer by layer by hash; tampering at any step invalidates every later check, so tampering always leaves a trace
eFuse circuit-break read-only	Lv8 locking: on severe contradiction or chain tampering the affected branch is isolated locally, protecting the main chain without destroying data
Internal/external isolation rule	Human intrusion into the cloud kernel is forbidden; only seed-style evolution is allowed. SSH and similar writes pass the gate (whitelist, serial, atomic, backup, audit)
Risk grading and the three-minima principle	P0–P3 four risk levels, handled in the order “minimum risk > minimum cost > maximum benefit”, with risk holding a veto
Quota and circuit-break protection	Quota shortage enters read-only idle; dead loops trip the breaker; tasks are reclaimed after a 1200s timeout

Engineering methodology and delivery matrix

METHODOLOGY

The methodology has one main line:**lightweight is the highest engineering constraint**. Heavy features are first distilled to their core and rebuilt light; what one script can do does not become a service. Hence six production lines, each running from input straight to a deliverable instead of stopping at a proposal.

Line	Input → processing → output	Deliverable form
Decision line	Problem definition → data collection → option generation → three-dimensional steady-state appraisal → memo	Decision memo, option comparison table, risk matrix
Research line	Topic → literature and external fact-checking → framework → whitepaper → archive	Whitepaper HTML / PDF, visual page, citation list
Compliance shield	Contract and clauses → risk identification and grading → legal-basis retrieval → opinion	Structured opinion, clause-level risk tracing
Content line	Worldview → storyboard → keyframes → video → voice → review	Keyframes, final footage, asset pack
Evaluation line	Real engineering event → value filtering → item building → self-test → difficulty callback	Item pack, scoring sheet, self-test record
Archive line	Any asset → four-layer split → nine meta-classes → hash entitlement → ledger	Lock receipt, Merkle root credential, index

Delivery matrix One body of truth can become six carriers for different audiences: whitepaper (decision-makers), visual page (reviewers and public), item pack (evaluators), skill package (executors), memo (internal), demo footage (content side). Different carriers, one source of fact — the structural way to avoid one story outward and another inward.

Capability boundary statement

BOUNDARIES

Dimension	Available / not available
Operator layer	Available: real engineering handling of multi-backend numerical consistency checks, drift quantification, concurrency arbitration and quota degradation. Not available: first-hand CUDA / NPU custom kernel development — I do not claim it.
Compilers and compute graphs	Available: decomposing pass defects and rewrite plans from an item-design perspective. Not available: production experience independently completing MLIR / TVM pass development.
Team size	One person leads all architecture, implementation and operations; this is not presented as team output. External dependencies (compute, models, platforms) are explicitly separated into self-built and procured.
Data convention	Every number carries collection time and scope; where measurement is impossible it is N/A, never an estimate; parent and subset are listed separately and never double-counted.

Writing boundaries is not self-depreciation; it lets a reviewer judge precisely which parts can be taken as given and which need separate verification. A portfolio without boundaries simply transfers the cost of verification to the reader.

Review guide

HOW TO READ

Read the structure Chapters 01 / 04 / 05 are the skeleton: the seven-layer steady state answers “how the system is organised”, the kernel and operators answer “how it runs itself”. Establish those two first, and the later chapters on product lines and assets stop feeling fragmented.

Read the evidence Chapter 11 is where credibility lands — three production incidents with root cause, fix and verification data; chapter 13 gives five independently accessible channels, any of which can be opened and checked without relying on this portfolio's own account.

Read the boundaries Chapter 17 states plainly what is not available. To verify a specific claim further, raise it under the matching dimension in chapter 17 and I can provide that dimension's raw records, ledger receipts or reproduction steps.

Open access and ecosystem

OPEN ACCESS & ECOSYSTEM

Earlier chapters describe what the system does; this one describes how the outside can plug in. These four capability classes mark the boundary between “for my own use” and “usable by others”, and are where a reviewer should test whether the ecosystem is real.

Capability	Form	External value
A2A federation access	External nodes register, handshake and synchronise truth and assets by protocol; five federation whitepapers and an access page	Another party's node can join the federation without rebuilding its own system
Capability marketplace	Demand posting → capability appraisal → listing → fulfilment → feedback; five modules: demand, experts, assets, rules, feedback	Capability accepted by third parties on results, with the whole fulfilment traceable
Truth gateway	Public API documentation plus a node view; writes are signature-checked, reads authorised by DID	Cross-system truth write and query without depending on this system's client
Octop meta-operations platform	Deployed on its own subdomain for multi-agent orchestration and operations, with account registration	Internal operations capability externalised into a deliverable platform

Design trade-off What is open is the **positioning, structure and interfaces**, not the core decision algorithms and their weights; external node access requires authorisation, and federation handshakes are not open by default. That is both a security constraint and the boundary that keeps the ecosystem sustainable.

Site and repository asset overview

SITE & REPOSITORY MAP

Section	Contents	Chapter here
System and kernel	omega-brain overview, technical architecture, autonomy levels, autonomy footprint, boot memory	01 / 04
Operators and evaluation	27-operator topology, operator architecture, benchmarks and comparisons (overview and role-governance)	05 / 06
Product lines	AIOS, Kunlun Dongtian, integrity guard, industrial-park platform, capability marketplace	02 / 03 / 18
Whitepaper library	Online whitepapers grouped by theme, with appendix specials	08
Governance and security	AxiomGate product suite, governance baseline, audit and monitoring	14
Data lake and repositories	Five-channel mirror; kernel base, gallery, shared data lake	07 / 12
Brand and culture	Daoist line, workforce, SOLAR specials, multilingual site	15 / 17

This table merges the site's actual sections rather than enumerating pages; the page-level list is the site index, so the portfolio and the site never carry two drifting catalogues.

Operations base · shared brain

There is another layer of assets that is never shown externally but decides whether the system keeps running:**65 online ledgers** forming the operations base. Tasks, assets, decisions, alerts, nodes, rules, protocols and keys each have a ledger of their own, so any action lands on a specific row. This layer is the difference between autonomy and relying on memory.

Ledger family	Registered entries	Facts carried
Meta-rule registry	38	Rule ontology with effective status and unique ID, read and executed by programs
Capability registry	36	Capability entries, skill status, corresponding deliverables
Protocol library	27	Definitions for communication, authentication, handshake and reporting
Asset ledger / drive index	200+ (paging not finished)	Asset root ID, snapshot, hash, ownership hierarchy
Breakpoint SOP library	20	Standard failure-handling steps, directly callable by patrol scripts
Open-source delivery components	9	Deliverable component list with licence status
Nodes and federation	Node table / handshake records / message sync	Node identity, liveness, handshakes and cross-node message traces
Security and keys	Key pool / SSH key management	Central credential registration and rotation records, not scattered through code

Boundary note This layer is internal operational assets: externally only ledger categories and entry counts are disclosed, never access, record detail or credential content. The portfolio's job is to show that this kind of management exists and is quantified, not to publish it.

Other ledger families (partial)

Ledger	Purpose
Task ledger · decision log	Status transitions and the basis for each decision; the decision log keeps “why this choice then”, so review is review, not post-hoc rationalisation
Alert and failure rollup · blockers	Failures merged and counted by type, blockers tracked separately, so the same problem stops recurring unnoticed
Memory semantic index · SGSA triples	Semantic retrieval entry and entity–relation triples, supporting cross-document entity unification and graph queries
Compute pool · window state sync	Real-time sync of available compute and task windows, avoiding duplicate submissions and quota collisions
Historical archive seal	Sealed zone for expired assets; sealing is not deletion — assets can be retrieved and hash-checked when needed
Truth consensus zone · reporting rollup	Consensus results and reporting flows of multi-node truth, the basis for cross-node agreement

How the ledgers are used Daily patrols write run records and alerts; weekly reconciliation of assets against the ledger clears stale entries; monthly sealing of expired assets and index recomputation; every major change logs a decision first. Ledgers never say “to be filled later”, because back-filling is retrospective invention.

Channel convention Five data channels are independently accessible externally (ModelScope, AtomGit, PaddlePaddle, CNB, Gitee/GitHub); internally there are nine sync channels, adding the cloud hub, local persistence, Feishu ledgers and truth mirrors. The two counts use different definitions and are never merged.

Open-source operator set and verification

This layer is the most independently verifiable: code is public, cases run, dependencies are zero. A reviewer does not have to believe any narrative — clone it and run the tests.

Component	Version	Composition and verification result
Daotong operator set	v1.1	19 named operators: guard self-healing (Tiangang), seven-step order-repair state machine (Jiushu), topology repair (Nüwa), heartbeat verification (Hengzheng), quality gate (Guinìè), unified verification facade; 100+ test cases all green, zero third-party dependencies
Yuheng YUHENG	v0.1.0	Governance loop engine, five modules, 10 cases all green
Xuangui	M1	Signature receipt trio: canonical HMAC + anchor + probe, 22 cases all green
Huaxia numerics kernel	v0.4	Taixuan ternary, Luoshu nine palaces, sexagenary cycle, solar-term astronomy (seven anchors off by about 9 minutes), twenty-four-mountain spatio-temporal matrix; 25 cases green, each version appends and never overwrites. An engineering implementation — theoretical completeness is not claimed
AxiomGate adjudication kernel	entry version	Rule adjudication engine, 54 cases all green; the entry version is sealed and under review

Four-venue canonical repositories

GitHub	`github.com/tanbopiao/daotong-ops`
AtomGit	`atomgit.com/zongyuangen/daotong-ops`
Gitee	`gitee.com/huodou-cloud-intelligence-aios/daotong-ops`
ModelScope	`modelscope.cn/models/zongyuanroot/daotong-ops`

Three engineering constraints

Zero third-party dependencies	Every component uses only the standard library, so it runs on clone; no framework means no framework upgrade and security debt
Cases before features	Each operator has a failing case before implementation; the case count is the acceptance record, and release requires all green
Append only, never overwrite	Releases evolve by appending, history stays retrievable, so “what it used to be” never becomes unexplainable

Mapping to the ten-layer topology

Component	Topology layer	Duty
Daotong operator set	L6 archive deepening · L7 reconciliation governance	Topology repair, heartbeat verification, quality gate, unified verification facade
Yuheng	L7 asset reconciliation governance	Five governance-loop modules turning reconciliation results into actions
Xuangui	L8 security and entitlement	Signature receipt trio, canonical HMAC and anchor verification
Huaxia numerics kernel	L2 manifold metrics	Spatio-temporal coordinates and rhythm phase providing the metric baseline
AxiomGate	L9 decision and compliance shield	Three-state adjudication and scheduling admission

Four repositories share one source and identical content, so any reachable one reproduces the rest; case counts are the actual results of each repository's public suite, excluding local experiments not committed.

Visual and content assets

On the content side this is not “a few concept images” but series that can be counted and carry direct links. The numbers below come from file-by-file verification in the data lake (2026-11-11).

Series	Scale	Content and craft
Cyber-Guofeng keyframes	54 frames	Fully transparent materials, liquid-metal texture, complete front/back and close-up sets at all angles
New-Chinese portrait	24 frames	An oriental photographic portrait series, released the same day as the keyframes
Kunlun Dongtian drama line	89 files	EP01 live-action keyframes, vertical keyframes, character settings, plus production plan and pipeline report
Pale-gold dreamlike Guofeng CG	52 frames + gallery page	Complete keyframe sets with gallery HTML and manifest, incrementally archived to the data lake

Repository and persistence scale

Object	Scale	Purpose
Shared data lake	1,836 files • 760.3 MB	Fully pulled local and verified file by file as the asset base record
Asset registrations	72,039	Each with ownership, snapshot and hash
ModelScope repository matrix	196 (51 datasets / 145 models)	Tiered hosting of content and assets
Media repository group	about 8.8 GB	Main media repo 1.9 GB, character keyframes 2.2 GB, gallery 388 MB, 3D world 3.9 GB, plus a CDN layer
Media direct-link index	292	The index is complete here; on 2026-11-11, 216 IP keyframes were pulled from the public repo into local lock-up, so publicly reachable items are fewer than the index total

Content production chain

Stage	Step	Output
S1	Worldview and characters	Mainline setting, character lineage, artefact and scene lists; meta-rules are defined before any image
S2	Storyboard and keyframes	Episode outline → shot table → keyframe prompts, with shot numbers and durations
S3	Image generation and consistency check	Character three-view, front/back close-ups; form and palette must not drift across rounds
S4	Footage assembly	Vertical cut, five-frame PV, key visual, delivered at 9:16
S5	Archive and direct-link publishing	Into the data lake, hash, asset registration, stable direct link

Visual standards

Aesthetic rule	Purely oriental visual language, no external sourcing; no existing IP elements unrelated to the worldview
Consistency	Across rounds the same character keeps locked posture, palette and lighting; sampling randomness never changes it
Aspect ratio	Drama and keyframes are predominantly vertical, suited to mobile placement
Traceability	Keyframes carry the anchor mark and trace back to worldview entries and generation batches

Public repository and local lock-up are two layers: presentable work goes public, character-private material moves local and its public copy is withdrawn. Removal is not loss; it is deliberate reduction of exposure. Assets are never uploaded to paid object storage — everything runs on zero-cost channels, and stable direct links are what let this content be reviewed externally.

Productisation and disclosure governance

Form	Capability and delivery
Truth security product	SHA256 hash entitlement + Merkle snapshot chain + tamper detection + behaviour audit; three forms: individual licence / enterprise licence / private deployment
Adjudication sidecar	Unified scheduling across API clusters, load balancing, exception circuit breaking, task queueing; four health endpoint groups, independently deployable
Evaluation item pack	De-identified item bank, difficulty tiers, automatic answer key and scoring standard; directly usable for enterprise assessment and role-capability acceptance
Entitlement registration service	DID entitlement, hash snapshot, ledger archive, certificate issuance and provenance verification
Autonomous governance health check	Checks against the ten-layer topology and ledger integrity, producing a health report, risk grading, remediation plan and steady-state score

Disclosure governance: why this portfolio is "thin"

External material uses two-domain isolation:**general public domain** presentable and sellable; **internal source domain** proprietary and never licensed out. Before release a disclosure check automatically blocks three classes — classified, semi-finished, internally defective. Seven classes never enter an external portfolio:

① Low-level server operations detail, ports, config paths and load data	② Assets in research or planning that are not yet finished
③ Intermediate versions of incomplete mathematical models	④ Unde-identified internal ledgers, task flows and iteration logs
⑤ Experimental features without pricing or lock-up	⑥ Character-private material (local archive only)
⑦ Internal records of hardware shortfalls and known engineering defects	

Delivery forms and acceptance criteria

Form	Acceptance criteria	Evidence available
Private deployment	Interfaces reachable, health checks pass, task queue observable	Deployment receipt, health endpoint output, log excerpts
Item pack licence	Items scoreable, difficulty gradient holds, answers and scoring standard consistent	Sample items, scoring script, difficulty calibration note
Entitlement registration	Hash recomputable, chain verifiable, certificate checkable	SHA256 receipt, Merkle root, DID registration record
Governance health check	Issues locatable, risks gradeable, remediation re-checkable	Health report, risk matrix, re-check comparison

Why these qualify as products The criteria are recomputable: hashes recompute, cases rerun, interfaces connect, items score. Anything failing those four never enters the external list, however complete the narrative — that is the line between this portfolio and a proposal-style body of work.

This boundary is itself a capability Being able to draw clearly what should not be public shows the assets were managed in tiers rather than piled up for display. To verify a claim, a reviewer raises it under the chapter 17 dimensions and I provide reproducible steps and receipts — but not the seven classes above.

Six-platform hosting and mirrors

The same body of work is distributed across six platforms, but not "everywhere": each has a defined role, there is one primary, and the rest are mirrors.

Platform	Role	Scale	Public state
AtomGit	Public primary repository	384 files / 21 MB	Publicly clonable; whitepapers, articles, proposals and skill packages centralised here
Tencent CNB	Cold storage	1,875 files	Publicly clonable; scripts de-identified, not used for display
ModelScope	Public docs repo	16 files (after cleanup)	Whitepaper mirror; IP media removed, repository down from 813 MB to 528 KB
Baidu PaddlePaddle AI Studio	Skill package copy + snapshots	52 files	Anonymously downloadable; skill packages may be public, kernel snapshots may not — needs split handling
GitHub	Private cold backup	356 files	private, with sensitive-file cleanup and an admission gate completed
Gitee	Cold-storage master	—	private, anonymous access restricted, serves as the disaster-recovery master

Primary and mirrors

Work	Primary	Mirror and note
15 skill packages	AtomGit	Same name and MANIFEST on AI Studio, comparable item by item
172 meta-rules	AtomGit	Local persistence + three whitepaper editions on the site
SYM-LATTICE whitepaper	ModelScope	No second copy; contains 3 ledgers and the meta-rules
Kernel snapshots	Local persistence	AI Studio was the only external copy — a risk point, now listed for migration back
5 whitepaper HTML mirrors	ModelScope	Published in sync with the site

Deduplicated work composition

Theoretical documents	8 classes of system documents + 172 meta-rules (three editions), forming the root literature
Technical articles	6, written for external readership from the engineering angle, not the conceptual one
Commercial proposals	3: a B2B deployment proposal, the eight-volume lock-up SOP series, and an agent engineering delivery pack
Skill packages	15 (13 skills + 2 experts), same name and MANIFEST on two platforms
Tools and ledgers	Sync bridge, truth reporting, ledger receipts; categories only — the code sits in the cold-storage tier

Three exposure tiers

Tier	What goes there	Access rule
Display tier	Whitepapers, articles, proposals, skill packages	Anonymous read and clone; no credentials or internal snapshots
Collaboration tier	Deliverable copies, MANIFEST, delivery packs	Account required; per-object authorisation, item-by-item checks
Cold-backup tier	Kernel source text, masters, private material	Private; reachable only from this machine and the master repository, restored on demand

Division principle The display tier holds only public content; cold storage holds kernel text and masters and carries no display; no credentials, private keys or internal snapshots may appear in public repositories. Mixing the three trades the certainty of security for the convenience of display — a bad trade.

Five layers of rule ontology

Rules are not one list but a five-layer structure: the lower the layer, the more specific; the higher, the harder to change. Layers derive from each other; they are not parallel.

Tier	Scale	Content and binding force
Meta-constitution	ML-001/002/005/010/007	Highest tier, defining system boundaries and non-transferable items; changes need explicit authorisation, and routine tasks may not touch it
Meta-rules	172 · three editions	Backbone rules covering governance, delivery, security, quota and evolution; 38 of them registered as program-readable registry entries
Meta-rule set	Tianyuan L0-001~003 · axioms M1/M2/M3 · ROOT-001~003 · ARCH/GOV series	Executable-level rules that directly constrain task scheduling and output form
Meta-decision	META-DECISION-001	Adjudication order in conflict: minimum risk > minimum cost > maximum benefit
Architecture axioms	—	Structural constraints: internal/external isolation, append-only ledger, single-source governance

Public wording (fixed)

So that different materials do not each tell a different story, this portfolio uses one sentence throughout:**172 meta-rules + 8 classes of system documents + 6 technical articles + 3 commercial proposals + 15 skill packages**. Kernel snapshots, rule text and credentials are not included, and no material may extend these four numbers on its own.

8 classes of system documents	Meta-rule whitepaper, architecture whitepaper V2, SYM-LATTICE whitepaper, meta-constitution, meta-decision, CTE loop, federated memory pool SFMP, channel protocol V31
6 articles	Engineering delivery, AIGC drama pipeline, autonomous kernel practice, catalogue evolution and federation, federated memory pool practice, cloud autonomy practice
3 proposals	B2B engineering deployment proposal, eight-volume lock-up SOP series, agent engineering delivery pack
15 skill packages	13 skills + 2 experts, same MANIFEST on two platforms, checkable piece by piece

How rules actually take effect

Step	Stage	Actual action
1	Clause drafted	The rule is written as a decidable sentence with a unique ID and conditions, never “as the case may be”
2	Registry entry	Enters the registry with status and scope; unregistered rules do not participate in execution
3	Program read	Admission is checked by a program reading the table, not by anyone remembering
4	On hit	Out of bounds blocks, above threshold alerts, severe cases trip isolation
5	Write-back and review	Every hit leaves a trace, counted by type, feeding rule revision

Why five layers instead of one list Change cost must be inversely matched to rule stability: the deeper the rule, the rarer the change; the closer to execution, the more frequent. Merged into one layer, routine adjustment quietly edits boundaries that should not move.

Once the wording is fixed, a reviewer sees the same numbers on any platform — which is the precondition for cross-verification.