
Versal: Versatile Evolution of Reusable Structure for Adaptive Learning

J. R. M. Gardner
Ardea AI, with Apart Research
john@ardea.io
ORCID: 0009-0000-9879-9882

Research conducted at the Digital Minds Research Sprint, August 2026.

Abstract

Intelligence appears in many forms and substrates, yet most learning systems are built around a fixed architecture and discard task-specific structure after training. I investigate a first-principles alternative: an inspectable digital mind that evolves reusable neural structure, preserves verified discoveries, and makes them available to later search. Versal: Versatile Evolution of Reusable Structure for Adaptive Learning, couples topology evolution with gradient-trained weights and orchestrates library lookup and refinement, sparse routing, grammar induction, resolution-independent spatial fields, task-shaped evolution, hierarchical composition, and recursive decomposition under one typed support/query contract. I evaluated one cold seed-0 run on 36 attempts (34 unique task references) spanning all 18 author-composed Icarus rungs. Every attempt received blind held-out evaluation, including all 15 that reached a search deadline. The run consumed 1,183 generations and 1.86 recorded task-hours. Mean task-appropriate score across rung means was 0.6099; mean literal sample/cell accuracy was 0.6471. Performance was sharply heterogeneous: five rungs exceeded 0.95 literal held-out accuracy, while CIFAR-100 averaged 0.10 and spherical classification 0.00. The library grew from zero to 18 indexed records (17 live), including a level-3 composition. A spatial field admitted on one Cosmic task was later retrieved for a distinct, same-resolution task and achieved 0.9965 held-out accuracy, although verification and attempted refinement were not cheap. Historical repeated-task runs additionally show persistent reuse, five-level composition, and compression of a perfect XOR circuit. These results establish integrated, auditable structure-building across heterogeneous data, not benchmark superiority, general intelligence, consciousness, or welfare.

1. Introduction

Human intelligence is one workable organization of cognition, not a complete specification of what intelligence must be. Designing learning systems exclusively through familiar human abstractions risks embedding those abstractions as unexamined priors. Conversely, modern frontier systems can be strikingly capable while retaining a jagged generalization profile: a strong aggregate impression can hide simple failures, brittle transfer, and representations that are difficult to audit. This motivates a structural question that precedes any claim about consciousness or moral status:

Can one persistent learning system construct, verify, retain, and recombine neural structure across heterogeneous tasks without a human authoring a task-specific network architecture?

Versal is an experimental answer. Its durable state is not one ever-growing set of weights. It is a typed library of immutable modules and compositions, plus an evolving router over those entries. Each task begins by testing prior structure, then escalates through configurable search strategies. Candidate weights are trained by gradient descent, while evolution controls graph topology, activation, aggregation, recurrence, and composition. Verified discoveries return to the library; useful failures can remain as stepping stones. The resulting *overmind* is a digital mind in an operational, architectural sense: a persistent computational organization with memory, specialization, routing, and lineage that can be inspected as it develops. That description is not evidence of subjective experience.

This paper makes three contributions:

1. **A substrate for cumulative structural learning.** Versal joins neuroevolution, differentiable weight training, sparse expert routing, field programs, and hierarchical composition around a persistent, content-addressed library.
2. **A heterogeneous evaluation contract.** Icarus presents 18 author-composed rungs, from Boolean functions through control, vision, scientific data, abstract reasoning, and ARC-AGI-1 through the same typed support/query interface.
3. **An audited cold-run account.** A complete seed-0 canary provides held-out results for all 36 attempts, exposes large support-to-query gaps rather than hiding them, and records the library and lineage produced along the way. Historical runs provide bounded evidence about persistence and compression; their older reporting semantics are kept separate.

The contribution is therefore a working and inspectable research substrate, not a claim that Versal is generally intelligent or competitive with specialized architectures. I designed the search space, operators, objective contracts, orchestration policy, and Icarus curriculum. I did not author a separate network architecture for each evaluated task; the networks were evolved.

2. Related Work

Versal inherits the central idea of NEAT: begin with small graphs and protect structural innovation while topologies complexify [1]. It differs in its division of labor and its unit of memory. Evolution proposes structure; gradient descent trains weights and writes them back into candidates. What persists is a cross-task library rather than a single champion. This places Versal near neural architecture search [2], learning-guided evolution [3], and quality-diversity or stepping-stone methods that preserve alternatives an objective-only search would discard [4,5].

Persistent modular learning has close precedents. Progressive Networks freeze earlier columns to avoid catastrophic forgetting [6]; PathNet evolves routes through a shared supernet [7]; modular continual-learning systems choose whether to reuse, adapt, or create components [8]. DreamCoder and later library-learning systems similarly turn solved programs into a vocabulary for later search [9]. Versal applies the same cumulative-memory intuition to evolved neural graphs: entries are executable subnetworks with typed interfaces, compositions may reference earlier entries recursively, and lookup re-verifies an entry rather than trusting a remembered score.

The routed strategy is related to sparse mixture-of-experts systems [10], but its experts are independently discovered, frozen library entries rather than jointly trained blocks. Spatial-field evolution follows the generative-encoding tradition of compositional pattern-producing networks [11]: a compact program is evaluated over coordinates rather than materializing one unconstrained connection per spatial site. Icarus draws several scientific modalities from NAS-Bench-360 [12] and includes ARC-AGI-1 as a final abstract-transformation rung [13]. Its rung order is a curriculum in presentation, not a calibrated proof of monotonically increasing difficulty.

Digital-minds research warns against both over-attributing and under-attributing morally relevant properties to AI systems [14-16]. Versal does not elicit preferences, distress, introspective reports, or persona stability. Its relevance is instead methodological and forward-looking: future candidate digital minds may be persistent, adaptive systems rather than only static language models. Versal makes structural lineage, admission, reuse, refinement, and retirement visible, providing an auditable substrate on which future agency, identity, and welfare questions could be posed without treating present architectural behavior as evidence of consciousness.

Rung	Name	Task
1	XOR	2-bit XOR, the smallest non-linearly-separable problem
2	Parity-N	parity of a padded, mask-delimited bit vector
3	Two-spirals	which of two interleaved spirals a 2-D point lies on
4	Pole (Markov)	next-state prediction of a cart-pole from a short window
5	Double-pole (no velocity)	two poles, velocities hidden (a memory benchmark)
6	MNIST / Fashion-MNIST	10-way grayscale image classification
7	CIFAR-10 / CIFAR-100	natural color image classification
8	NB360 ecg	single-lead ECG rhythm classification
9	NB360 satellite	land-cover from a satellite time series
10	NB360 ninapro	hand gesture from surface-EMG
11	NB360 spherical	spherically-projected image classification
12	NB360 cosmic	per-pixel cosmic-ray segmentation
13	NB360 darcy_flow	Darcy-flow PDE solution-operator regression
14	NB360 psicov	protein residue-distance regression (variable length)
15	NB360 fsd50k	multi-label audio tagging from a mel-spectrogram
16	NB360 deepsea	multi-label chromatin marks from a DNA window
17	RAVEN / PGM	abstract visual analogy (pick the completing panel)
18	ARC-AGI v1	few-shot grid program induction (native train/test split)

Table 1. The Icarus curriculum. The author-composed ladder spans Boolean functions, control traces, vision, scientific modalities, abstract reasoning, and ARC-AGI-1. All tasks are exposed through typed input/output fields with support and held-out query splits. Rung order expresses a curriculum design; it is not a calibrated monotonic difficulty scale.

3. Methods

3.1 Icarus task contract

An Icarus task contains support and query pairs of typed tensor fields. Descriptors specify semantic axes, value type, class count, range, and masks. Loss selection follows those descriptors, not the task name, so the same consumer handles binary, categorical, multilabel, continuous, ordinal, temporal, and spatial outputs. Library compatibility is also structural: signatures derive from input/output descriptors and widths rather than benchmark identity. Tasks satisfying this field contract can be evaluated by Versal; the present evidence covers only the sampled Icarus rungs.

Support examples are visible during search. Query targets are held out until the final report step. A candidate can therefore fit the support set, fail support-fold validation, and still receive a blind held-out score. Conversely, a high held-out score does not retroactively make an artifact admissible. These two rails answer different questions: *what did search fit?* and *how did the best executable champion generalize?*

3.2 Per-task lifecycle

For each task, the orchestrator first evaluates compatible library entries. A hit can seed a bounded refinement attempt, but failed refinement returns the incumbent. On a miss, the configured strategy ladder is:

1. **Route experts:** learn sparse top- k routing over frozen library entries and, where possible, distill an executable composition.
2. **Apply grammar:** instantiate candidate structures from motifs induced across stored lineages.
3. **Evolve spatial field:** search compact coordinate-conditioned programs for aligned spatial mappings.
4. **Evolve dense network:** evolve a task-shaped graph directly.
5. **Compose modules:** evolve references and glue among prior modules.

If the root search stalls, registered decompositions can propose independently solvable subtasks; none succeeded in the reported cold run. The best executable champion remains available throughout the lifecycle. At task termination, including a deadline, the system evaluates that champion once on the blind query, unless no executable champion exists.

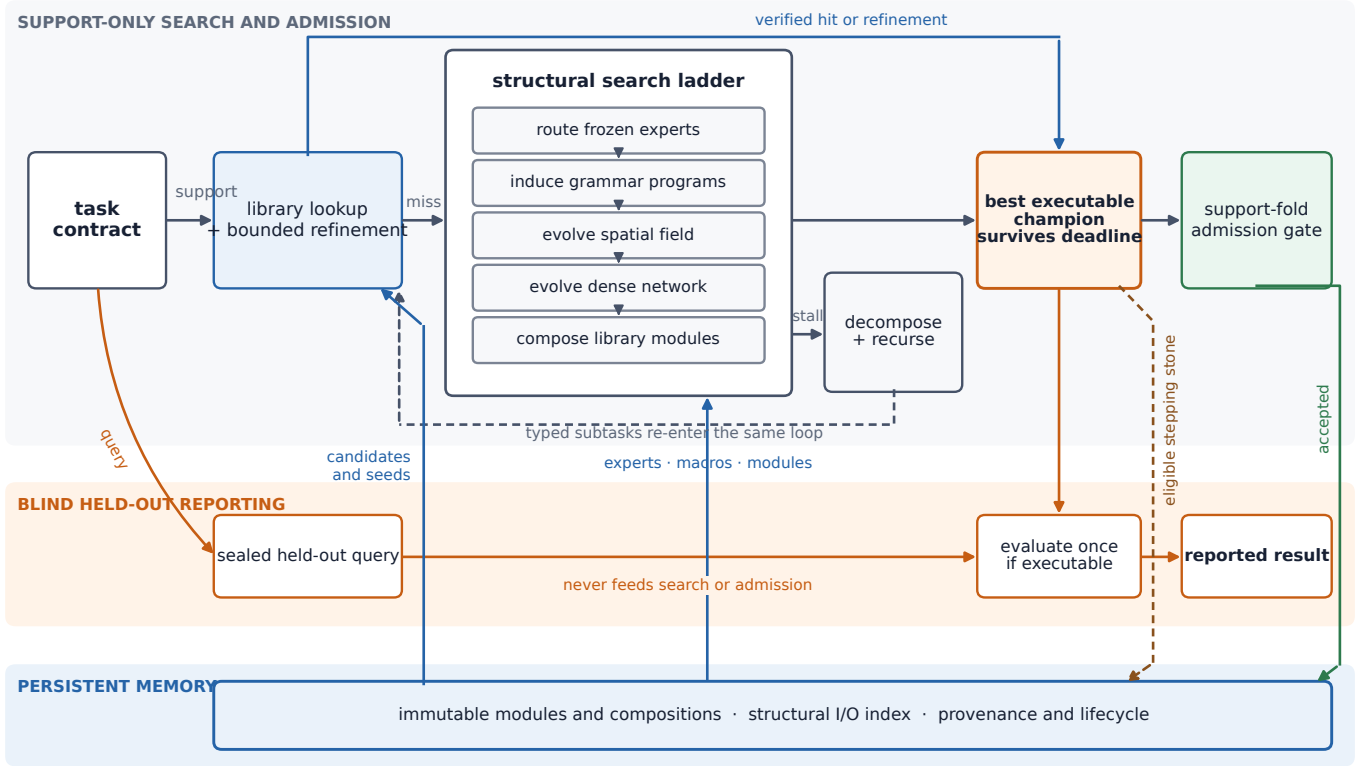


Figure 1. Versal’s per-task lifecycle. Search-time support evaluation and support-fold validation govern admission; blind held-out evaluation is a reporting rail and does not alter the library. The persistent substrate supplies lookup candidates, frozen experts, composition references, and warm-start stones to later tasks.

3.3 Evolution, training, and admission

Direct candidates are graph genomes initialized from small generative networks. Mutation may add or remove nodes and connections, change activation or aggregation functions, add recurrence, or reference stored macros. NSGA-II selection balances task fitness, novelty, and parsimony; speciation protects structural innovation. Composition candidates evolve graphs over stored module references. Gradient-based inner loops train candidate weights, with Lamarckian writeback into survivors.

The cold canary used population size 64 for direct and field search and 48 for compositions. The depth-0 orchestration budget was 400 generations, split across routed, grammar, field, direct, and composition strategies, with a 400-second ordinary search cap and a 600-second total task cap. A support-side, task-appropriate score of 0.95 was required for acceptance. Candidates at the gate were tested on 2-5 support folds; at least half had to pass. Below-threshold champions above the configured floor could be archived as stepping stones but could not pass lookup without fresh verification at the full gate.

Library records are content-addressed and immutable. Modules are level 1; a composition’s level is one plus its deepest dependency. Garbage collection retires dominated or excess entries while preserving referenced dependencies. The router’s learned traffic state and structural library are related but distinct artifacts; the paper’s overmind portrait cold-renders the routable subset of the final library, not measured routing traffic.

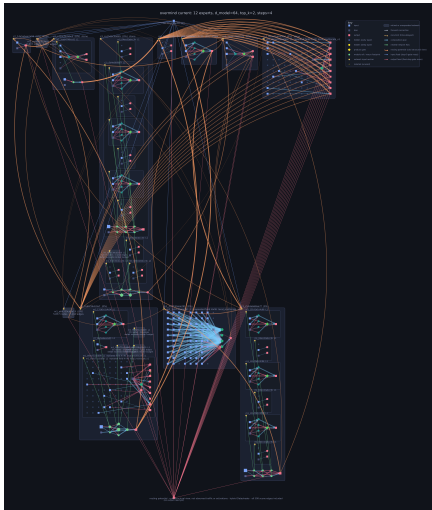


Figure 2. Cold structural portrait. Twelve routable experts are shown without a trained gate; edges indicate structural potential, not observed traffic or consciousness.

3.4 Experimental protocol and metrics

The primary study was one cold seed-0 run at code commit 043e1e2 with a clean worktree and an empty starting library. It used Icarus revision 412029e and an exact, pinned task manifest. The interleaved schedule attempted two tasks per rung: 36 attempts over 34 unique references because XOR and two-spirals each have one fixed task and were revisited. The run executed on an Apple M4 Max and recorded 1,183 total generations and 6,702.8 task-seconds.

The primary aggregate is the unweighted mean of the 18 rung means using each task’s configured task-appropriate held-out metric. This *cross-rung mean* gives every heterogeneous family equal weight. I also report literal sample/cell accuracy because it is intuitive and preserves the system’s raw evaluation rail. For ARC, however, literal cell accuracy is not an exact task solve; the task-appropriate metric is exact-grid accuracy. Sample standard deviations across rung means describe dispersion only. With one run and two tasks per rung, they are not confidence intervals.

The two historical studies were frozen before several current reporting and admission changes and are analyzed separately: a 400-encounter, rungs-1-6 persistence run, and a 200-encounter repetition of one XOR task. They provide mechanistic observations, not pooled benchmark evidence.

4. Results

4.1 Complete held-out coverage and heterogeneous generalization

All 36 attempts produced a blind held-out evaluation, including all 15 attempts marked as reaching a search deadline. This directly verifies the lifecycle property that motivated the final reporting fix: deadlines terminate search, not evaluation of an already available executable champion.

Across 18 rung means, the task-appropriate held-out score was **0.6099** (SD across rung means 0.3499; range 0-1). Mean literal sample/cell accuracy was **0.6471** (SD 0.3151). Mean support accuracy was 0.8869, yielding a 0.2769 mean macro gap against the task-appropriate rail. Five rungs averaged at least 0.95 literal held-out accuracy: XOR, pole, double-pole, Cosmic, and FSD50K, while CIFAR-100 averaged 0.10 and spherical classification 0.00.

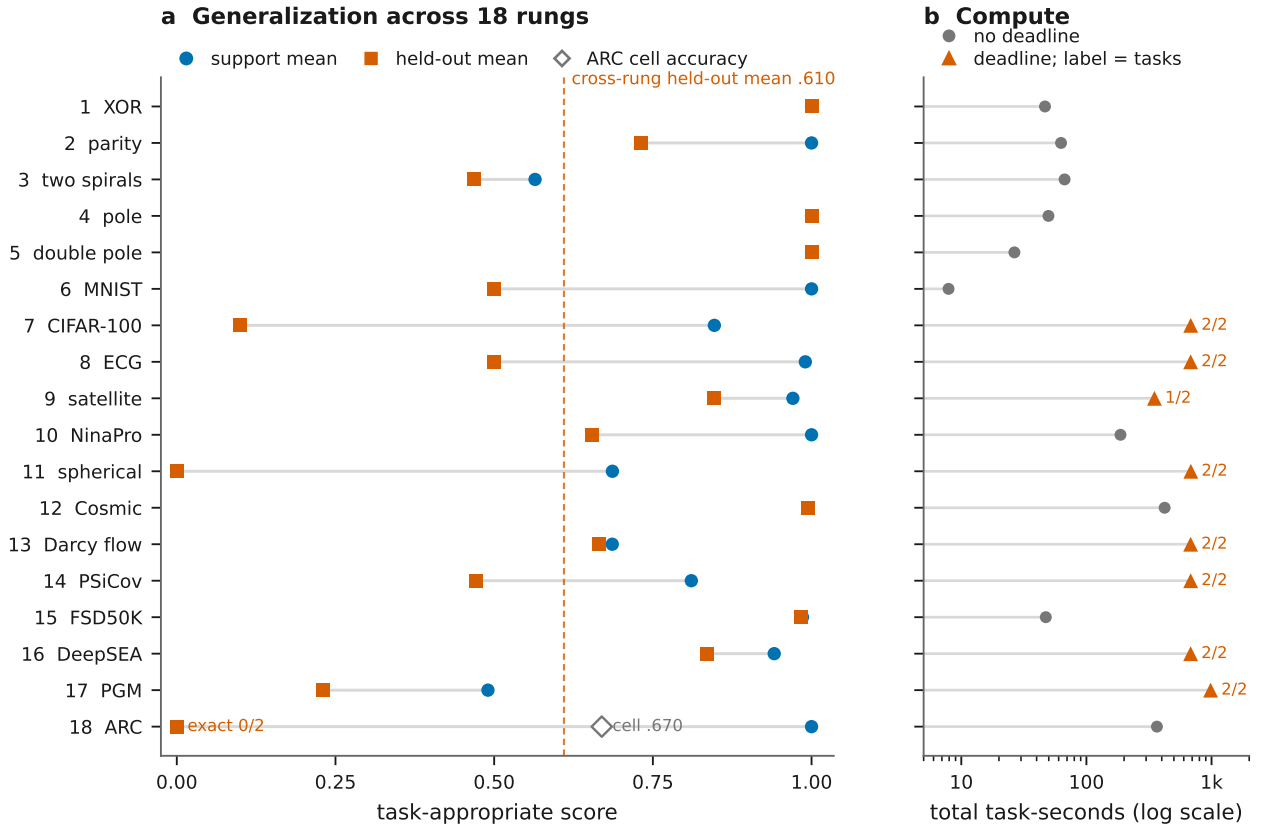


Figure 3. Cold-canary performance across all 18 rungs. Points are rung means over two scheduled attempts (one unique fixed task revisited for XOR and two-spirals). Connected literal support and task-appropriate blind held-out values make generalization gaps explicit. The headline 0.6099 cross-rung mean uses configured task-appropriate query metrics. Literal ARC cell accuracy averaged 0.6695, but exact held-out accuracy was 0/2; these are not ARC solves. Valid zeroes are plotted as zero, not missing.

The largest literal support-to-query gaps were CIFAR-100 (0.7469), spherical (0.6863), MNIST (0.5000), and ECG (0.4902). Across individual attempts, query accuracy was below support on 24/36, equal on 8/36, and above it on 4/36. These failures matter more than the headline mean: the system often found structures that fit small support sets without representing the rule needed by the query distribution. Seven high-query attempts nevertheless ended with a failed outcome because outcome encodes acceptance and support-fold status rather than post-hoc query success. Conversely, both accepted MNIST compositions had 0.500 held-out accuracy. “Accepted” is therefore not a synonym for “generalized.”

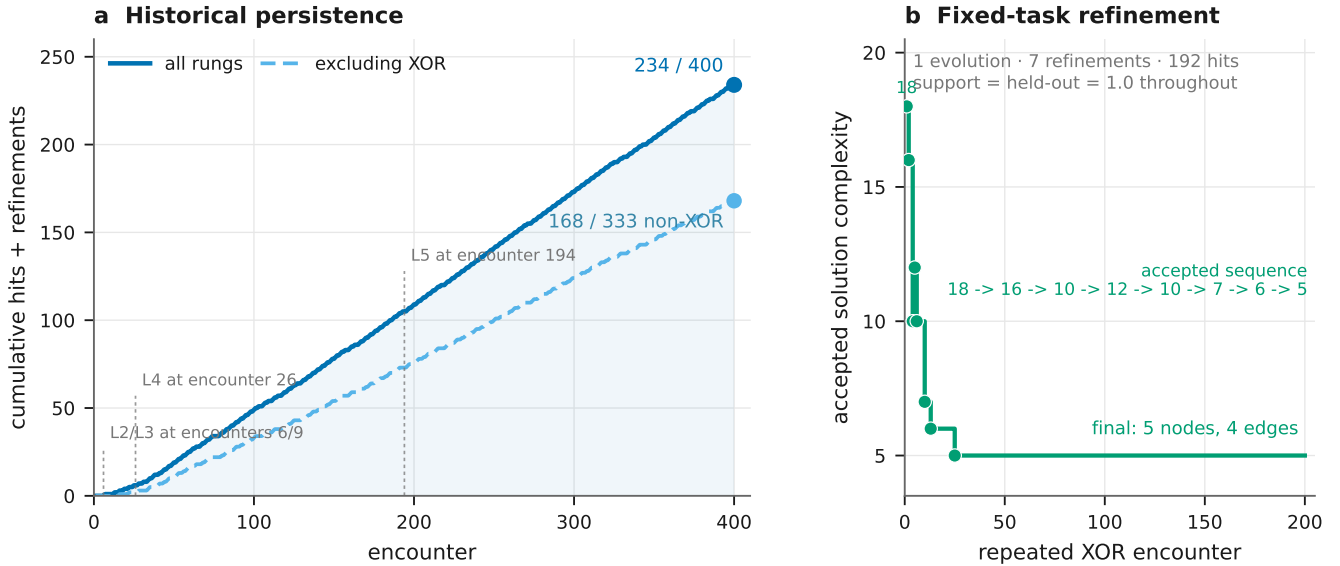
4.2 What persisted

The library grew from 0 to a peak of 19 records and ended with 18 indexed records: 17 live entries and one retired dependency retained because an accepted composition referenced it. The final structure comprised 12 modules and 6 compositions, reaching level 3. Seven live, non-dependency artifacts passed support-fold validation; the remainder were dependencies or stepping stones. The level-3 accepted composition demonstrates that previously discovered structure was actually embedded recursively, although this run does not show that each dependency was necessary or improved accuracy.

The clearest cross-task reuse occurred on Cosmic. A `local_multiscale_v1` spatial field admitted on `cosmic.b432` at 0.9919 support and 4/5 passing support folds. A later lookup on distinct `cosmic.b201`, with the same 256×256 resolution, directly evaluated that field at **0.9965 held-out accuracy**. This is evidence of same-resolution cross-task reuse. It is not cross-resolution reuse: a report counter labeled it that way because stored field metadata changed the dictionary compared by the counter. It is also not evidence of lower cost. The lookup task took 344.1 seconds because the configured always-refine path attempted four generations and reached its field deadline without improving the incumbent.

4.3 Historical persistence and XOR compression

The strongest larger persistence trace is a historical cold run of 400 encounters over rungs 1-6. It recorded 217 direct library hits and 17 refinements (234 memory-mediated outcomes) while the frozen library contained 24 live entries and a recursively referenced level-5 composition. Even after excluding XOR, 168/333 encounters were hits or refinements. Direct hits had a median duration of 0.001 seconds versus 9.20 seconds for evolved outcomes. These timings are descriptive, not causal: task identity, difficulty, encounter order, and outcome are confounded, and there was no matched no-memory arm.



Observational traces from earlier software states: persistence is not a matched cost comparison; XOR is one fixed task, not 200 independent tasks.

Figure 4. Historical mechanistic evidence, kept separate from the current benchmark rail. (a) Outcomes in a cold 400-encounter rungs-1-6 run and the five-level retained reference chain. (b) One fixed XOR task repeated 200 times: one initial evolution, seven strict refinements, and 192 hits reduced recorded champion complexity from 18 to 5. Both studies used one deterministic run and repeated task encounters; they do not establish independent rediscovery or causal speedup.

The XOR run returned perfect support and query accuracy on all 200 encounters. Its recorded champion complexity moved 18 -> 16 -> 10 -> 12 -> 10 -> 7 -> 6 -> 5 by encounter 25 and then remained stable; the final rendering contains five nodes and four edges. Three archived seed-0 launches also produced the same initial content-addressed XOR key and byte-identical payload in two generations, showing deterministic reproducibility under that fixed seed. I could not recover the final five-node entry payload. Although the topology is compatible with a compact sinusoidal XOR construction, the render does not encode activation identity, and preserved accepted XOR entries containing sine have sine nodes disconnected from the output. I therefore report the verified topology compression, not an unverified claim that the final circuit used sine or was mathematically minimal.

5. Discussion and Limitations

What the canary establishes

Versal executed the same orchestration and task contract across all 18 heterogeneous Icarus rungs; always reported an available champion after search; created executable modules, fields, and nested compositions; and performed one verified cross-task library reuse. Its failures are informative. The separation between support, support folds, and held-out query exposes overfitting that a support-only narrative would hide. The final library is also inspectable at the level of immutable payloads, references, validation status, and lineage. These are useful properties for a digital-mind research substrate: the system's persistent organization can be observed directly instead of inferred only from conversational behavior.

Limitations

The evidence is intentionally narrow. The main run has one seed, 34 unique task references, and only two attempts per rung. Rungs have heterogeneous metrics and sample counts, so the cross-rung mean is a system-level summary, not a standardized benchmark score. Task order and accumulating state are confounded. No matched baseline shows that persistence, routing, composition, or any individual strategy improved quality or compute. The one library hit was high quality but slow under always-refine. Fifteen attempts reached their search deadline. No root decomposition succeeded, no grammar candidate was selected as the final strategy, and router diagnostics should not be read as executable parent solutions.

Icarus is author-composed and therefore encodes my priors. It accepts a broad typed field contract, not arbitrary data. I designed the objective, operators, mutation vocabulary, gates, and curriculum even though I did not hand-author per-task architectures. Several apparently strong support solutions generalized poorly. ARC’s 0.6695 mean cell accuracy corresponded to 0/2 exact tasks and must not be presented as an ARC solve.

Most importantly for the Digital Minds context, architecture and capability are not evidence of consciousness, sentience, preferences, welfare, distress, selfhood, or general intelligence. Versal produced no introspective or welfare measurements. “Digital mind” here names the persistent adaptive organization visible in the overmind, not a conclusion about moral patienthood.

Future Work

The immediate priority is a matched, three-seed persistence ablation: compare full cross-task memory with a control that resets library, router, and shared modules for every task, using interleaved cold tasks, distinct within-rung tasks, and exact revisits. The primary outcomes should be revisit time, effective generations, held-out quality, and whether the memory lever was actually exercised. Without that control, compounding remains a design hypothesis supported only by observational traces.

Engineering work should next target the large support-query gaps, especially wide classification and scientific tasks; calibrate support-fold validation for tiny fixed supports; correct field compatibility accounting; and distinguish a cheap lookup-only mode from always-refine. Multi-seed evaluation, wider task sampling, and causal ablations of routing, field evolution, and hierarchy are needed before comparative claims. Reinforcement learning could add online action and consequence, while adversarial coevolution could generate counterexamples and shifting environments. Both would expand the research question and are therefore deferred.

Longer term, Versal could serve as the reusable-structure layer of a system that models distinctions among world, self, other agents, and value. That possibility is speculative. A future study would need operational tests of identity, agency, preference, self-model stability, and welfare rather than extrapolation from network diagrams.

6. Conclusion

Versal demonstrates that one orchestrated neuroevolution system can build and retain executable structure while traversing an unusually heterogeneous task ladder. In the finished cold canary, it completed 36/36 held-out evaluations, achieved a 0.6099 task-appropriate cross-rung mean, exposed substantial generalization failures, grew a three-level library, and reused one field program on a distinct task. Historical runs show deeper persistence and reproducible XOR compression, while also making clear how much remains observational.

The practical contribution is an auditable substrate: an evolving digital mind whose modules, compositions, lineage, gates, failures, and reuse can be inspected. It is not evidence that the system is conscious or aligned. These results instead motivate a stronger conjecture: alignment understood only as externally imposed behavioral constraint may be a dead end, perhaps even a fool’s errand. Durable alignment may instead require internally grounded distinctions among object, self, other agents, society, and value. Versal does not test that conjecture; it provides an inspectable substrate in which parts of it could eventually be studied.

Code and Data

Versal source code is available at github.com/ArdeaAI/Versal. Icarus is available at github.com/ArdeaAI/Icarus-Dataset and huggingface.co/datasets/ArdeaAI/Icarus-Dataset. The primary run pins the code commit, dataset revision, configuration, task manifest, empty starting-library hash, and durable output hashes. A machine-verifiable evidence manifest accompanies this draft.

Author Contributions

J. R. M. Gardner conceived and implemented Versal and Icarus, designed and ran the experiments, analyzed the results, and wrote and revised the manuscript.

References

1. Stanley, K. O., & Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2), 99-127. doi:10.1162/106365602320169811
2. Elsken, T., Metzen, J. H., & Hutter, F. (2019). Neural architecture search: A survey. *Journal of Machine Learning Research*, 20(55), 1-21. arXiv:1808.05377
3. Hinton, G. E., & Nowlan, S. J. (1987). How learning can guide evolution. *Complex Systems*, 1, 495-502.
4. Pugh, J. K., Soros, L. B., & Stanley, K. O. (2016). Quality diversity: A new frontier for evolutionary computation. *Frontiers in Robotics and AI*, 3, 40. doi:10.3389/frobt.2016.00040
5. Ecoffet, A., Huizinga, J., Lehman, J., Stanley, K. O., & Clune, J. (2021). First return, then explore. *Nature*, 590, 580-586. doi:10.1038/s41586-020-03157-9
6. Rusu, A. A., et al. (2016). Progressive neural networks. arXiv:1606.04671
7. Fernando, C., et al. (2017). PathNet: Evolution channels gradient descent in super neural networks. arXiv:1701.08734
8. Mendez, J. A., & Eaton, E. (2021). Lifelong learning of compositional structures. *ICLR 2021*. arXiv:2007.07732
9. Ellis, K., et al. (2021). DreamCoder: Bootstrapping inductive program synthesis with wake-sleep library learning. *PLDI 2021*. doi:10.1145/3453483.3454080
10. Shazeer, N., et al. (2017). Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *ICLR 2017*. arXiv:1701.06538
11. Stanley, K. O. (2007). Compositional pattern producing networks: A novel abstraction of development. *Genetic Programming and Evolvable Machines*, 8, 131-162. doi:10.1007/s10710-007-9028-8
12. Tu, R., et al. (2022). NAS-Bench-360: Benchmarking neural architecture search on diverse tasks. *NeurIPS 2022 Datasets and Benchmarks*. arXiv:2110.05668
13. Chollet, F. (2019). On the measure of intelligence. arXiv:1911.01547
14. Butlin, P., et al. (2023). Consciousness in artificial intelligence: Insights from the science of consciousness. arXiv:2308.08708
15. Long, R., et al. (2024). Taking AI welfare seriously. arXiv:2411.00986
16. Anthropic. (2025). Exploring model welfare. Research note

Limitations and Dual-Use / Ethical Considerations

A.1 Over-attribution and under-attribution

Versal’s networks, routes, persistence, and self-modification-by-search are observable computational properties. They do not demonstrate phenomenal consciousness, valenced experience, a stable self, autonomous preferences, or moral status. Calling the rendered substrate a digital mind is an operational metaphor for its persistent adaptive organization. No distress elicitation, self-report, preference test, or welfare assay was performed, and no model produced distress-associated language in this study.

The converse error is also possible. Restricting digital-minds inquiry to conversational language models could overlook other persistent adaptive substrates. Versal broadens the candidate space worth instrumenting, but this study supplies no evidence for where moral concern should begin. Future work should pre-register observable indicators and distinguish architectural, behavioral, and phenomenological hypotheses.

A.2 Dual use

Persistent open-ended search and reusable capability accumulation could increase autonomy, lower the cost of acquiring capabilities, or make provenance harder to follow as hierarchies deepen. The present system mitigates some of these risks through content-addressed immutable entries, explicit lineage, executable re-verification, support-fold gates, blind held-out reporting, bounded search, retirement and garbage collection, and inspectable artifacts. These mechanisms improve auditability; they do not guarantee safe behavior or prevent capability misuse.

The work does not include deployment, environmental action, self-replication, recursive source-code modification, or independent resource acquisition. Adding reinforcement learning or adversarial environments would materially change the risk profile and should be paired with explicit containment and monitoring.

A.3 Reproducibility record

The primary evidence comes from `results/20260817_031349_orchestrated` and `library_canary_clean_seed0`. The run used code commit `043e1e253d2345515ae70ee15b55c012b7e2bfc7` with a clean worktree, seed 0, Icarus revision `412029ed1b86072a08f47102959d3ebdc9dee7` and an empty starting library. SHA-256 digests are pinned for the run summary, manifest, effective configuration, task pool, manuscript, scripts, and figure outputs in the accompanying evidence manifest.

LLM Usage Statement

OpenAI Codex assisted with repository and artifact inspection, quantitative aggregation, figure generation, claim checking, and drafting. The author set the research direction and framing, implemented the system, ran the experiments, selected the final claims, and verified every reported result against pinned artifacts.