SCIENTIFIC SEARCH DEPENDS ON ITS SUBSTRATE: CONSTRUCTING AND AUDITING EXPLORATION REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Scientific search depends on decisions made before the first experiment: which alternatives remain distinguishable and which transitions a policy can take. We formalize this organization as a scientific exploration substrate and introduce CASES (Constructing and Auditing Scientific Exploration Substrates), a protocol for constructing, auditing, and activating substrates. A task policy records scientific, semantic, structural, and executable requirements; the audit authorizes the states and operations those requirements support. Controlled comparisons show that changing the substrate can alter search outcomes even when the policy remains fixed. Across scientific settings, the effects vary with task structure and consumer capability, shaping both which solutions can be reached and how evidence can be reused. Together, these results establish scientific search organization as an explicit experimental variable, separable from the policy that explores it.

1 INTRODUCTION

Search policies can select only among alternatives their computational organization makes distinguishable and reachable (Maus et al., 2024). Collapsed states and absent transitions cannot be recovered by selection alone. This creates an attribution problem: an observed gain may come from the selector, the organization, or both. We study organization as a separate variable and call the states, distinctions, and transitions exposed to a scientific search policy the *scientific exploration substrate*. In a controlled GB1 intervention on the measured protein landscape (Wu et al., 2016), holding the candidate library, environment, oracle, query budget, and selector fixed while changing only the organization changes observed access to the global optimum. Hill climb also reaches it on the same neighborhood, showing that the organization exposes the target to simple local search.

Scientific agents increasingly connect hypothesis generation to experimental feedback (Yamada et al., 2025). Separable Pathways restructures hypothesis spaces as evidence arrives (Alderete et al., 2026). As an observation changes which explanations remain plausible, an agent must decide where to look next. Its selector can prioritize available alternatives, but it can act only on preserved distinctions and permitted transitions that let it follow up on unexpected evidence.

The same scientific alternatives can induce different search problems. A learned latent space supplies coordinates for molecular search (Gómez-Bombarelli et al., 2018), while Bayesian optimization couples an encoding to a surrogate and acquisition rule (Snoek et al., 2012). Directed evolution and materials discovery organize candidate pools and local moves (Yang et al., 2025; Malik et al., 2026). To attribute an outcome to the representation, its organization must be specified and varied separately from the policy that searches it under the same task and budget.

Natural language makes this upstream choice especially visible. A scientist can propose that hypotheses A and B fail separately but succeed together under condition Z. An executable representation may nevertheless merge A and B, omit Z, or provide no operation with which to test their interaction. Autoformalization translates language into formal objects (Wu et al., 2022), and Hypothesis Search implements natural language hypotheses as programs (Wang et al., 2024). The remaining question is which organization preserves the scientific distinctions needed for exploration (Fig. 1).

Code, prompts, and execution artifacts: https://anonymous.4open.science/r/CASES-666.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

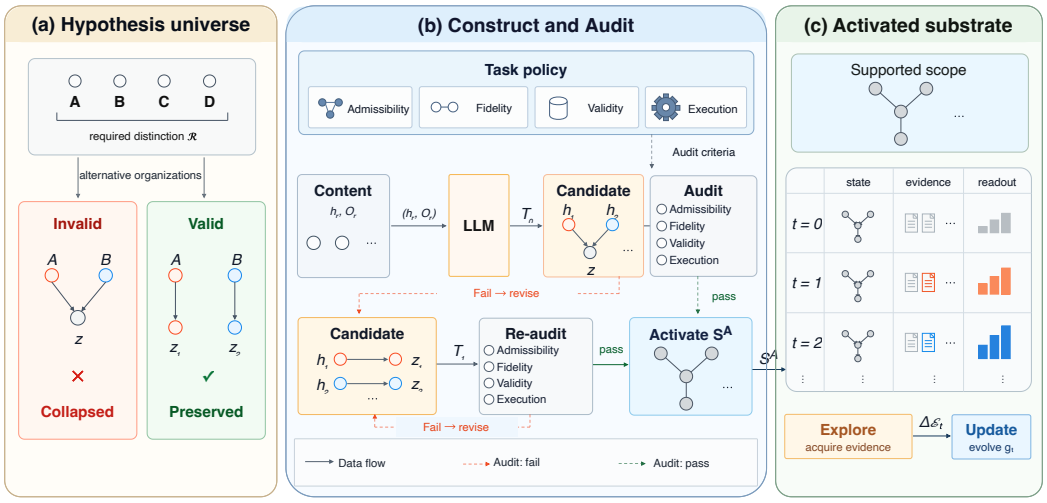


Figure 1: **The same scientific alternatives can induce different search problems depending on how they are computationally organized.** (a) Alternative organizations collapse or preserve a required distinction. (b) CASES proposes an organization, audits it against a task policy registered before construction, revises failed candidates, and activates the supported scope. (c) Exploration keeps the activated substrate fixed while new evidence updates its readout.

Candidate sets name available alternatives, and encodings give them computational form. A scientific exploration substrate also specifies what its states mean, which distinctions survive realization, and which transitions a consumer can execute. Representations may implement these parts. The substrate makes their joint commitments the unit of construction, audit, and controlled intervention. Its authorized organization can therefore change while the candidate universe and selector stay fixed.

We introduce **CASES (Constructing and Auditing Scientific Exploration Substrates)** to construct and authorize this object. A task policy registered before construction specifies required distinctions and checks. Audit activates only supported states and operations. Evidence then updates the readout on that fixed substrate (Protocol 1).

Our contributions are threefold. First, we formalize substrate organization as an experimental variable distinct from the search policy (Sec. 2.1). Second, we define CASES as a task-registered contract for construction, audit, and activation, with guarantees for preserving task requirements, making activation independent of representation family, and preserving registered distinctions (Secs. 2.2–2.3 and Appendix B). Third, controlled studies across six scientific settings show that organization changes search outcomes and downstream reuse, with effects shaped by task structure and consumer capability (Sec. 3). This makes substrate organization a testable part of scientific search.

2 Constructing and Auditing Exploration Substrates

Appendices A and B summarize the notation and formalize the substrate contract and its authorization guarantees.

2.1 Theoretical Formulation

Let τ be a scientific task with specification $\mathfrak{T}_\tau = (x_\tau, D_\tau, \mathcal{U}_\tau, \mathfrak{E}_\tau)$: a question, evidence available at construction time, a candidate universe, and an evaluation environment. Let $\mathcal{H}_\tau^{NL}$ denote hypotheses stated in natural language before they have computational form; this space need not be finite or enumerable. Let $H_\tau \subseteq \mathcal{H}_\tau^{NL}$ denote the hypotheses represented by a particular candidate substrate. The universe $\mathcal{U}_\tau$ specifies the alternatives available to the experiment.

Definition 1 (Scientific exploration substrate)**.** A substrate for τ is $S_\tau = (C_\tau, F_\tau, \Gamma_\tau, I_\tau)$. Scientific content C_τ specifies entities, distinctions, relations, and conditions relevant to the task. Computational form F_τ supplies states Z_τ and operations available to a downstream consumer. The realization relation $\Gamma_\tau \subseteq H_\tau \times Z_\tau$ connects scientific hypotheses to computational states, and I_τ interprets hypotheses stated in natural language within that organization. The correspondence may be partial and many-to-many.

At exploration step t, evidence E_t identifies promising or solution-bearing structure $R_t^{\mathrm{prom/sol}}$ within S_τ. Its interpretation is $\mathcal{S}_t^{NL} = \mathsf{Lift}(R_t^{\mathrm{prom/sol}}, E_t; S_\tau) \subseteq \mathcal{H}_\tau^{NL}$. For a budget B and consumer π, let $T_B(S, \pi)$ be the query-and-observation trace and $M(T_B)$ a task endpoint. When the universe, consumer, and environment are fixed, $\Delta_B(S_1, S_0; \pi) = \mathbb{E}[M(T_B(S_1, \pi)) - M(T_B(S_0, \pi))]$ measures how a change in organization affects the endpoint. Section 3.3 compares the same GB1 library organized as one-hot states or Hamming-1 neighbors. With selector, budget, and environment fixed, the contrast tests the route exposed through a shared candidate universe.

2.2 Construct: content, form, and realization

Construct returns a finite proposal $\chi = (\widehat{C}_\chi, \widehat{F}_\chi, \widehat{\Gamma}_\chi, \widehat{I}_\chi, \mathcal{K}_\chi)$, where $\mathcal{K}_\chi$ declares the candidate's scientific and operational claims. Content, form, and realization are specified together under the task policy.

Scientific content. The content component records claims relevant to the task, their contextual meaning, and their compatibility structure; Appendix B gives this structure precise semantics and compiles it into audit obligations. Candidate declarations can add local consistency requirements. Let $J_{\tau,\chi}^{\mathrm{core}}$ be the Grothendieck topology generated by requirements imposed by the task and $J_{\tau,\chi}$ the topology after adding requirements induced by the candidate.

Proposition 1 (Non-weakening of task-imposed local requirements)**.** If the task and candidate requirement families are instantiable on the same context category $\mathcal{C}_\chi$, then $J_{\tau,\chi}^{\mathrm{core}} \subseteq J_{\tau,\chi}$, and inclusion induces a full embedding

$$\mathbf{Sh}(\mathcal{C}_\chi, J_{\tau,\chi}) \hookrightarrow \mathbf{Sh}(\mathcal{C}_\chi, J_{\tau,\chi}^{\mathrm{core}}).$$

Thus candidate structure may refine the task's local consistency conditions but cannot remove them (Appendix B, Proposition B.1).

Computational form. The form component supplies the candidate's states, executable operations, and runtime interfaces. The common proposal-and-audit interface accepts graphical, metric, symbolic, executable, and hybrid forms. Computational states and operations need not live inside the semantic category.

Realization. The realization relation $\widehat{\Gamma}_\chi$, together with the interpretation interface $\widehat{I}_\chi$, connects claims in the content layer to states of the form layer. Because interpretation may be partial or set-valued, semantic grounding and execution are decoupled; a proposal is well formed only when content, form, and realization jointly specify what its states mean and which operations run.

2.3 Audit: certification before exploration

Before inspecting χ, the task registers a verification policy $\mathcal{V}_\tau = (\Phi_\tau, \mathcal{O}_\tau, \mathcal{A}_\tau)$: required scientific distinctions Φ_τ, mandatory checks $\mathcal{O}_\tau$, and allowed operational scope $\mathcal{A}_\tau$. Instantiating the policy against a candidate compiles core obligations from the task and induced obligations from the candidate's own declarations; candidate claims may add obligations, but they cannot remove registered ones. The proposal therefore cannot choose the tests by which it is judged.

Audit evaluates four kinds of commitments. Scientific admissibility asks whether proposed entities and relations have task support; semantic fidelity asks whether required distinctions survive their computational realization; structural validity checks the form's declared constraints; executability checks that exposed operations work as specified. Every mandatory obligation must pass, regardless of performance on other dimensions. The activation rule maps the activation-relevant record $v_\tau(\chi)$, which contains mandatory-check outcomes and registered scores, to a decision; the representation-family label is not an input. The record a_χ names the supported states and operations, so a passing score alone cannot authorize an unsupported part of the proposal.

Proposition 2 (Representation-family-neutral activation)**.** For candidates χ_1 and χ_2 with the same activation-relevant verification record, $v_\tau(\chi_1) = v_\tau(\chi_2)$, the registered activation rule returns the same decision, regardless of their representation-family labels (Appendix B, Proposition B.3).

Definition 2 (Registered authorization)**.** Write $\mathsf{Pass}_{\mathcal{V}_\tau}(a_\chi)$ when all mandatory obligations and policy thresholds pass. Activation then returns S^A_χ, whose states $Z^A_\chi \subseteq Z_\chi$, realization relation Γ^A_χ, and operations are restricted to the supported scope. Otherwise it returns $\perp$. Thus, an activated interface exposes only commitments authorized by the registered policy and audit record.

The semantic-fidelity obligation becomes a separation condition: let $\Sigma_U(h)$, $\widehat{\Sigma}_U(z)$ be semantic and computational signatures from registered probes under pseudometric d_U, γ_U the minimum separation between distinct required classes, and ε_U a bound on $d_U(\Sigma_U(h), \widehat{\Sigma}_U(z))$ over certified pairs $(h, z) \in \Gamma^A_\chi$.

Theorem 1 (Conditional preservation of registered distinctions)**.** Assume every required class has a realization in Z^A_χ. If $\gamma_U > 2\varepsilon_U$, no state in Z^A_χ can be a certified realization of hypotheses from two distinct required classes.

Proof sketch. If one state realized hypotheses from distinct classes, the triangle inequality through its computational signature would place their semantic signatures at distance at most $2\varepsilon_U$, contradicting $\gamma_U > 2\varepsilon_U$. Appendix B gives the full definitions and proof.

Theorem 1 gives a sufficient condition for registered semantic distinctions to survive realization. Its guarantee is relative to the distinctions and probes in the task policy.

A candidate that fails certification is revised rather than discarded. Revision receives sanitized diagnostics that name the unmet obligations without prescribing replacement structures, and any revised candidate is audited against a fresh certification suite. The loop terminates either in an activated substrate or, after the attempt budget is exhausted, in an unresolved state; Sec. 3.2 reports both acceptance and termination behavior. After activation, $S_{\tau,t} = S^A_\chi$ stays fixed during the evaluated episode, while evidence E_t updates priorities on its supported states through $g_t = \mathsf{Readout}(S^A_\chi, E_t)$.

2.4 From construction to exploration

Protocol 1 (CASES). Given task $\mathfrak{T}_\tau$, construction evidence D_τ, a registered policy $\mathcal{V}_\tau$, and an attempt budget K:

1. **Propose:** for attempt $k \leq K$, construct a finite candidate χ_k and its declared scientific and operational claims.
2. **Audit:** evaluate registered and candidate-induced obligations. A failed attempt returns sanitized unmet obligations; any revision is audited with fresh certification evidence.
3. **Activate:** on a passing record, return the authorized interface $S^A_{\chi_k}$ of Def. 2; if no attempt passes, return $\perp$.
4. **Consume:** keep $S^A_{\chi_k}$ fixed during the evaluated episode, updating evidence E_t and readout g_t after queries.

Every setting supplies $\mathfrak{T}_\tau, D_\tau, \mathcal{V}_\tau$ and receives either S^A_χ or $\perp$; its explorer consumes (S^A_χ, g_t, E_t) after activation. The instantiated forms range from hypothesis and no-LLM search structures to reaction and variant neighborhoods, a reusable materials state, and a readout for an independent PiEvo agent. The protocol fixes the order of proposal, audit, activation, and use while allowing scientific content and computational form to depend on the task. Appendix C maps the common contract to the implemented checks and consumers.

3 Experiments

3.1 Experimental logic and provenance

The Introduction identifies organization as a search variable, CASES as an auditable contract, and controlled outcomes as the evidence for its relevance. We test what this means at three points in the

Table 1: **The role of each experimental setting.** Dots mark the primary contrast; controls and protocols appear in Appendix F.

Setting	Construct	Access	Reuse	Endpoint
ResearchBench	•			Hypothesis score
HypoSpace	•			RR-AUC
Buchwald–Hartwig		•		Region recall
GB1		•		Best fitness
MADE			•	mSUN
PiEvo			•	Task-specific best

Table 2: **ResearchBench hypothesis scores.** All methods use the same final consumer; the middle column identifies the upstream organization supplied to it. Higher is better. hardA is outcome-independent; hardB is a baseline-defined post hoc slice.

Method	Upstream organization	hardA	hardB	Overall
Fixed schema	Predefined fields	3.04	1.39	3.04
Editable structure	Editable claims and evidence	3.08	1.22	2.99
Direct refinement	Draft, critique, revise	3.30	1.75	3.20
Without Audit revision	Construct without Audit feedback	3.06	1.49	2.96
ResearchAgent NAACL'25	Literature-guided revision	3.33	2.01	3.23
VIRSCI ACL'25	Multiagent synthesis	2.84	1.55	2.83
CASES	Construct, Audit, Revise	**3.41**	**2.14**	**3.29**

workflow: construction, target access, and downstream reuse. ResearchBench (Liu et al., 2026) and no-LLM HypoSpace test construction. Buchwald–Hartwig reaction discovery (Ahneman et al., 2018) and GB1 protein exploration test target access.

MADE materials discovery and an independent PiEvo consumer (Pu et al., 2026) test the third claim as evidence or the consumer changes. Table 1 identifies the provenance and control in each setting; Appendices C, D, and F give the audit records, paired tests, and procedures.

3.2 Construction changes what reaches a shared consumer

CASES makes useful scientific distinctions available to a shared consumer. ResearchBench holds the final hypothesis consumer fixed and changes the upstream representation. CASES scores 3.29 overall, compared with 3.04 for a fixed schema and 2.96 without Audit-guided revision (Fig. 2a; Table 2). This directly tests the Introduction's concern: a consumer can reason over a distinction only if the supplied organization makes it available.

Direct Refinement and ResearchAgent narrow this gap, consistent with a capable reasoner reconstructing some missing structure during use. On hardA, standard Direct Refinement scores 3.30 against 3.41 for CASES, a gap of about 0.11 (Table 2). In a separate registered output-matched control, Direct Refinement scores 3.0562 against 3.4062 for CASES on the same 160 tasks, a 0.350 gap. These are distinct Direct Refinement conditions; paired-test details appear in Appendix D.

The causal-encoding ablation identifies a concrete part of that organization. Removing the clause that binds causal claims to evidence lowers the hardA mean from 3.406 to 3.263 on matched tasks (Appendix D). The gain therefore has a traceable link to an explicit relation in the artifact, rather than only to a larger or more elaborate prompt. The paired estimate and a representative case appear in Appendix D.

A sodium cathode case shows what this distinction looks like in the final artifact. CASES promotes state of charge as a control variable and connects it to ion migration and stacking faults; six comparison workflows instead concentrate on doping, coating, or composition changes. The shared consumer scores the CASES hypothesis 3 and each comparison 0 in this case (Appendix D). The case makes

Table 3: **Operational checks of the Audit boundary.** Each row uses its own frozen test set; complete records appear in Appendix C.

Check	Observed	What the record establishes
Corrupted proposals	$0/12$ pass	Injected corruptions stopped at the gate.
Governed recovery	$40/48$ valid	Eight blocked episodes produce no unauthorized recovery.
Fresh certification	$5/5$	Activated records use fresh suites.
Required distinctions	$9/9$	Obligations for four Construct activations meet the floor.

the ablation's mechanism concrete: a causal relation with low prior salience survives construction and remains available during final synthesis.

HypoSpace removes the language model proposer and fixes the candidate pool. CASES improves recovery-ranked AUC over Hypothesis Search by $+.0225$ in causal and $+.0185$ in Boolean tasks under identical budgets (Appendix D). The candidates already exist, so the gain concerns which ones the organization brings forward. Voxel is nearly saturated across methods and yields little separation; the complete domain results appear in Appendix D.

The domain means make this distinction visible. CASES and Hypothesis Search score 0.321 versus 0.299 in causal tasks and 0.353 versus 0.335 in Boolean tasks, but 0.786 versus 0.785 in Voxel. The same construction idea has room to improve ordering when relevant distinctions remain unresolved; once recovery is near saturation, another organization has little effect on the ranked endpoint. This complements ResearchBench, where a shared consumer receives different structured artifacts.

Audit determines which constructed distinctions can govern search. Across the open and menu lanes, 128 proposals yielded 121 candidates for Audit after seven execution errors; four activated (Appendix C). The registered gate rejected all 12 injected corruptions, while the required distinction obligations of activated substrates met the coverage floor (Table 3). Construction therefore supplies candidate structure, and Audit limits exploration to the tested scope. The candidate funnel and its limits are documented in Appendix C.

3.3 Organization changes access under controlled selection

CASES can steer a fixed query budget toward scientific regions that pointwise search misses. Buchwald–Hartwig separates finding individual high-yield reactions from covering independently defined high-yield regions. BO-GP queries more high-yield points by 240 queries (128 versus 112), while CASES reaches more regions (0.81 ± 0.13 versus 0.53 ± 0.15; Fig. 2b; Table 4). The substitution graph exposes reaction neighborhoods, and measured yields update its readout. The advantage grows during search: at 80 queries BALLET leads in region recall, but by 160 CASES leads. This contrast connects the substrate's transitions to the scientific target, rather than treating every high-yield query as interchangeable.

One recorded trajectory illustrates what the aggregate curve captures. In CASES seed 41, true high-yield reactions in successive 40-query batches rise from $0/40$ initially to $30/40$ in the final batch; the top ten unqueried archive improves from $0/10$ to $8/10$ true solutions (Appendix D). The graph does more than keep a list of promising points: its neighborhood and evidence readout redirect queries toward an increasingly concentrated part of the reaction space. The aggregate region metric tests whether this behavior extends beyond the illustrated seed.

A substrate alone can change whether a fixed selector reaches an optimum. In GB1, the candidate library, environment, oracle, 480-query budget, and CASES selector are identical across the intervention. Replacing one-hot organization with a deliberately specified Hamming-1 neighborhood changes observed global-optimum hits from $0/30$ to $9/30$ and mean terminal fitness from 7.05 to 7.58 (Fig. 2c; Table 5). A greedy hill climb on the same neighborhood reaches the optimum in $8/30$ runs: the neighborhood exposes the target to both policies. The hit counts describe this intervention; the registered primary mean-fitness comparison with GP-BO is 7.78 versus 7.55 ($p = .069$). Equal-pool controls in Appendix D distinguish optimum access from broad solution recovery.

These two interventions show that a useful substrate exposes the relations required by its endpoint. The reaction graph makes connected high-yield regions accessible as evidence arrives; the Hamming-1

Table 4: **Buchwald–Hartwig ranked region recall (50 seeds).** Top-60 recovery at each query budget; the 240-query endpoint reports mean ± standard deviation. Cited baselines are listed by publication year. Bold marks the highest mean.

	Region recall after oracle queries ↑					
Method	40	80	120	160	200	240
BO-GP BoTorch, NeurIPS'20	0.427	0.417	0.431	0.458	0.458	0.53 ± 0.15
BALLET ICML'23	0.545	**0.557**	0.508	0.509	0.504	0.48 ± 0.14
BO-DKL Commun. Chem.'24	0.180	0.223	0.232	0.245	0.255	0.27 ± 0.03
CASES	**0.596**	0.464	**0.628**	**0.685**	**0.698**	$\mathbf{0.81 \pm 0.13}$

Table 5: **GB1 outcomes after 480 queries (30 seeds).** The registered primary comparison and common-environment comparison are shown separately; the latter includes the one-hot intervention. Published baselines are ordered by year, followed by task controls. Fitness is mean ± standard deviation; see Appendix F.

	Primary endpoint	Common environment	
Method	Best fitness↑	Best fitness↑	Optimum hits↑
GP-BO NeurIPS'12	7.55 ± 0.00	7.22 ± 0.77	0/30
CLADE Nat. Comput. Sci.'21	5.99 ± 1.01	5.99 ± 1.01	1/30
Level-set AL ICML'23	6.33 ± 1.19	5.84 ± 1.00	1/30
ALDE Nat. Commun.'25	6.93 ± 0.97	6.93 ± 0.97	0/30
Hill-climb	6.96 ± 1.42	6.96 ± 1.42	8/30
Random	4.31 ± 1.22	4.31 ± 1.22	1/30
CASES (Hamming-1)	$\mathbf{7.78 \pm 1.11}$	$\mathbf{7.58 \pm 1.04}$	**9/30**
CASES (one-hot)	—	7.05 ± 0.94	0/30

neighborhood makes the protein optimum accessible through local moves. Neither relation guarantees the best outcome for every endpoint. On a common 25k GB1 candidate pool, GP-BO recovers more gold set members in all 20 paired seeds (Appendix D). Region coverage, broad recovery, and access to one extreme target reward different organizations of the same search effort.

Target concentration changes this balance. In a separate reaction face, random broad coverage is competitive early and when the high yield target is broad, while the graph gains value for concentrated regions after evidence accumulates (Appendix D). The substrate allocates a fixed query budget across the reaction landscape; the target distribution determines whether that allocation helps. The region result therefore depends on both target concentration and the stage of the search.

3.4 An activated substrate supports selective downstream reuse

An activated substrate can make accumulated evidence usable for the next decision. On MADE's 40-query hold-out task, CASES improves mSUN over the Length-matched summary (0.372 versus 0.354; paired $\Delta = 0.0175$, $p = 0.0346$) and leads in five of eight material systems (Fig. 2d; Table 6). Both arms use the same host and guidance allowance, so the contrast tests how observations are organized for the next query. AUDC and stable-discovery means are close (0.346 versus 0.343, and 14.88 versus 14.18); in the separate MADE-B task, the mSUN difference is not significant (Appendix D).

Both matched MADE arms receive guidance of comparable length through the same host interface. The Length-matched summary records evolving observations, while the CASES readout places them in an empirical field with coverage information. The mSUN gain thus concerns how evidence is organized for the next choice; the close AUDC and stable-discovery means show that the same contrast is less visible under broader aggregate endpoints.

The system-level results clarify this distinction. For Eu–Nb–Sn–Tl, CASES finds 16 stable structures across seven compositions, while the Length-matched summary finds nine across eight compositions

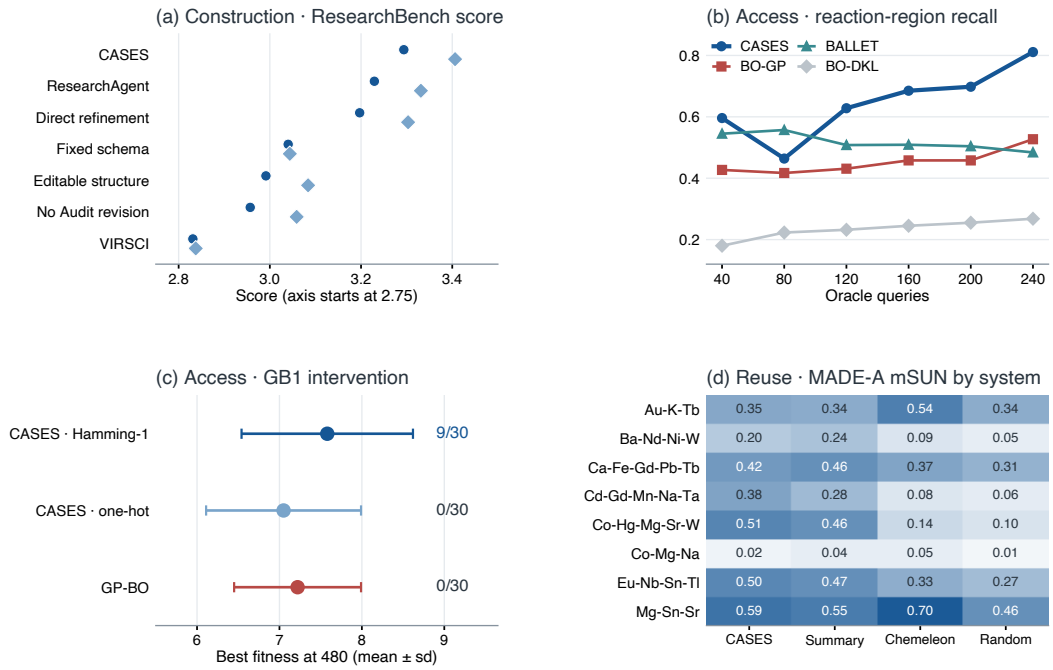


Figure 2: **Substrate organization affects construction, target access, and downstream reuse.** (a) ResearchBench scores for seven upstream organizations with one final consumer; circles show Overall (480 tasks), diamonds show hardA (160 tasks). The score axis begins at 2.75. (b) Buchwald–Hartwig region recall at six budgets (50 seeds). (c) GB1 common-environment intervention: points and whiskers show best fitness (mean ± sd), with optimum hits at right (30 seeds). The two CASES rows share a selector; GP-BO uses its native selector. Table 5 gives the separate registered primary comparison. (d) MADE-A mSUN across eight hold-out systems (five seeds per system), with four guidance arms.

(Appendix D). The endpoints distinguish depth within compositions from breadth across them. The summary leads in three of eight hold-out systems, so the aggregate mSUN gain combines different system-level responses.

Figure 2d makes those responses inspectable. On Cd–Gd–Mn–Na–Ta, CASES reaches mSUN 0.375 against 0.275 for the summary; on Co–Mg–Na the order reverses (0.020 versus 0.045). Chemeleon leads both on Au–K–Tb and Mg–Sn–Sr. The substrate's readout therefore changes the allocation of attention across material systems, with gains most evident in some local landscapes and little benefit in others. The average mSUN result summarizes these varied decisions rather than a uniform win.

MADE-B tests another use of organization: its search begins with known stable compositions and seeks structurally novel polymorphs. MADE-A begins without stable-phase information and seeks new stable structures. The two settings ask the readout to direct attention at different levels, within known compositions in MADE-B and across initially unknown stable regions in MADE-A. This difference in task semantics helps explain why the two mSUN contrasts should be read separately.

PiEvo changes the consumer while holding its guidance hook and native agents fixed. The readout enters through the Hypothesis Agent, where it names tested candidates, prioritizes untested ones, and identifies regions with little coverage. The Experiment and Principle agents and scorer remain unchanged. Transfer is mixed across tasks, with gains on SPO and TMC and declines on MBO and NHO; the task-level estimates are descriptive (Appendix D). This pattern links the value of the readout to the receiving agent's objective and its ability to use the exposed structure.

On SPO, the readout supports a higher best T_c than the character-matched summary (96.2 versus 81.3 K), whereas MBO favors that summary (Appendix D). The shared guidance interface makes these opposing outcomes a direct test of whether PiEvo can use the substrate's organization for its own objective.

Table 6: **MADE hold-out materials discovery.** Outcomes are mean ± standard deviation over 40 matched task-seed runs (eight systems, five seeds each). AUDC is normalized discovery-curve area; the final column is mean LLM calls per episode.

Method	mSUN↑	AUDC↑	Stable discoveries↑	LLM calls
Random	0.198 ± 0.165	0.206 ± 0.175	7.90 ± 6.58	0
Length-matched summary	0.354 ± 0.179	0.343 ± 0.180	14.18 ± 7.16	14.2
Chemeleon Nat. Commun.'25	0.286 ± 0.235	0.299 ± 0.242	11.43 ± 9.39	0
CASES	$\mathbf{0.372 \pm 0.208}$	$\mathbf{0.346 \pm 0.190}$	$\mathbf{14.88 \pm 8.30}$	14.6

Across PiEvo's four tasks, CASES also leads the summary on TMC (494.4 versus 449.8), while the summary leads on MBO (4.81 versus 4.25) and NHO (1.51 versus 1.27). Each pair uses the same receiving agent and guidance hook within its task. The reversals show why transfer cannot be summarized by a single rank across tasks: a readout helps when its exposed relations answer the next decision the consumer must make. The full paired outcomes remain in Appendix D.

When does the substrate matter? Organization matters when the substrate exposes relations relevant to the task and the consumer can use them within its budget. Those relations include causal links for hypothesis synthesis, reaction neighborhoods for region coverage, and single substitutions around protein variants. In reaction discovery, the graph's advantage widens after measured yields update its readout; a nearly saturated Voxel pool leaves little recovery order to improve. Downstream reuse adds the fit between the readout and the next decision. That fit varies across material systems and PiEvo tasks even when the host interface and budget are held constant within each task. The observed gains arise where scientific distinctions become usable computational choices.

4 Related Work

Scientific agents and hypothesis search. Scientific agents connect idea generation, experiments, and revision (Gottweis et al., 2026). Hypothesis Search implements language hypotheses as programs, while PiEvo evolves scientific principles during discovery. Separable Pathways restructures the hypothesis space as evidence arrives. These systems motivate a distinct question about the representation supplied to a search policy. CASES separates construction and audit of that organization from the policy that subsequently uses it.

Representation and search geometry. Latent representations and Bayesian optimization place representational choices inside the search procedure. BALLET directs queries toward high value regions with level set acquisition (Zhang et al., 2023); directed evolution similarly works over measured sequence landscapes. CASES treats scientific meaning, executable moves, and authorized scope as one inspectable interface; fixed-selector interventions isolate its effect on exploration.

Formalization, synthesis, and verification. Reflective autoformalization revises formal objects (Chen et al., 2026), and program synthesis learns reusable executable abstractions (Ellis et al., 2021). FunSearch evaluates candidate programs through execution (Romera-Paredes et al., 2024). CASES registers task obligations before construction, audits the candidate's commitments (Table 3), and measures the authorized organization's effect under controlled selection (Fig. 2).

5 Conclusion

Scientific search depends on both its selector and the substrate that makes alternatives distinguishable and reachable. CASES specifies this organization through construction, audit, and activation. Across six settings, controlled comparisons show that organization can change a fixed selector's reach and downstream evidence reuse, with effects varying by task and consumer. In GB1, changing organization altered observed access to the optimum; a simple hill climb on the same Hamming-1 neighborhood also reached it. Guarantees are relative to the registered task policy, the value of rejected candidates remains unknown, and the favorable GB1 organization was specified in advance. Automatically discovering the best representation remains open. These results establish substrate organization as an explicit experimental variable, separable from the policy that explores it.

AI Use Statement

Generative AI tools assisted with literature exploration, experimental planning, code debugging, interpretation of supplied results, language editing, and manuscript and figure presentation based on supplied results. The research questions, experimental campaigns, numerical outcomes, and final scientific judgments remain the authors' responsibility. The authors take responsibility for verifying the accuracy, originality, and integrity of the submitted work.

Reproducibility Statement

All six settings are specified in Sec. 2.4 and Appendix F, including task configurations, model deployments, prompts and interfaces, baseline adaptations, budgets, pairing rules, metrics, and statistical conventions. The frozen execution artifacts underlying the audit records of Sec. 3.2 are itemized in Appendix C, and the paired tests and complementary protocols are reported in Appendix D.

References

Derek T. Ahneman, Jesús G. Estrada, Shishi Lin, Spencer D. Dreher, and Abigail G. Doyle. Predicting reaction performance in C–N cross-coupling using machine learning. *Science*, 360(6385):186–190, 2018. doi: 10.1126/science.aar5169. URL https://doi.org/10.1126/science.aar5169.

John Alderete, Sebastian Benthall, Connie Xu, and John Xing. Separable pathways for causal reasoning: How architectural scaffolding enables hypothesis-space restructuring in LLM agents, 2026. URL https://arxiv.org/abs/2604.20039.

Jinheon Baek, Sujay Kumar Jauhar, Silviu Cucerzan, and Sung Ju Hwang. Researchagent: Iterative research idea generation over scientific literature with large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 6709–6738. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-long.342. URL https://aclanthology.org/2025.naacl-long.342/.

Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Ben Letham, Andrew Gordon Wilson, and Eytan Bakshy. BoTorch: A framework for efficient monte-carlo bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/f5b1b89d98b7286673128a5fb112cb9a-Abstract.html.

Guoxin Chen, Wu Jing, Xinjie Chen, Xin Zhao, Ruihua Song, Chengxi Li, Kai Fan, Dayiheng Liu, and Minpeng Liao. ReForm: Reflective autoformalization with prospective bounded sequence optimization. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/419bd6a89ab56e9c09ededb560b06192-Abstract-Conference.html.

DeepSeek-AI. DeepSeek-V4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348. DeepSeek-V4-Flash checkpoint, deployed locally; model card https://huggingface.co/deepseek-ai/DeepSeek-V4-Flash.

Kevin Ellis, Catherine Wong, Maxwell Nye, Mathias Sablé-Meyer, Lucas Morales, Luke Hewitt, Luc Cary, Armando Solar-Lezama, and Joshua B. Tenenbaum. Dreamcoder: Bootstrapping inductive program synthesis with wake-sleep library learning. In *Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation*, pp. 835–850. ACM, 2021. doi: 10.1145/3453483.3454080. URL https://doi.org/10.1145/3453483.3454080.

GLM-5 Team. GLM-5: from vibe coding to agentic engineering, 2026. URL https://arxiv.org/abs/2602.15763.

Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018. doi: 10.1021/acscentsci.7b00572. URL https://doi.org/10.1021/acscentsci.7b00572.

Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Petar Sirkovic, Artiom Myaskovsky, Grzegorz Glowaty, Felix Weissenberger, Alessio Orlandi, et al. Accelerating scientific discovery with Co-Scientist. *Nature*, 655:487–496, 2026. doi: 10.1038/s41586-026-10644-y. URL https://doi.org/10.1038/s41586-026-10644-y.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. DSPy: Compiling declarative language model calls into self-improving pipelines. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.03714.

Yujie Liu, Zonglin Yang, Tong Xie, Jinjie Ni, Ben Gao, Yuqiang Li, Shixiang Tang, Wanli Ouyang, Erik Cambria, and Dongzhan Zhou. ResearchBench: Benchmarking LLMs in scientific discovery via inspiration-based task decomposition. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 13187–13207. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.644. URL https://aclanthology.org/2026.findings-acl.644/.

Saunders Mac Lane and Ieke Moerdijk. *Sheaves in Geometry and Logic: A First Introduction to Topos Theory*. Springer, 1992. doi: 10.1007/978-1-4612-0927-0. URL https://doi.org/10.1007/978-1-4612-0927-0.

Shreshth A. Malik, Tiarnan Doherty, Panagiotis Tigas, Muhammed Razzak, Stephen J. Roberts, Aron Walsh, and Yarin Gal. MADE: Benchmark environments for closed-loop materials discovery, 2026. URL https://arxiv.org/abs/2601.20996. Accepted at ICML 2026.

Natalie Maus, Zhiyuan Jerry Lin, Maximilian Balandat, and Eytan Bakshy. Joint composite latent space Bayesian optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35163–35180. PMLR, 2024. URL https://proceedings.mlr.press/v235/maus24a.html.

OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Hyunsoo Park, Anthony Onwuli, and Aron Walsh. Exploration of crystal chemical space using text-guided generative artificial intelligence. *Nature Communications*, 16:4379, 2025. doi: 10.1038/s41467-025-59636-y. URL https://doi.org/10.1038/s41467-025-59636-y.

Yingming Pu, Tao Lin, and Hongyu Chen. Principle-evolvable scientific discovery via uncertainty minimization, 2026. URL https://arxiv.org/abs/2602.06448.

Yuchi Qiu, Jian Hu, and Guo-Wei Wei. Cluster learning-assisted directed evolution. *Nature Computational Science*, 1:809–818, 2021. doi: 10.1038/s43588-021-00168-y. URL https://doi.org/10.1038/s43588-021-00168-y.

Qwen Team. Qwen3.8-27B. Hugging Face model card, 2026. URL https://huggingface.co/Qwen/Qwen3.8-27B.

Benjamin Rhodes, Sander Vandenhaute, Vaidotas Šimkus, James Gin, Jonathan Godwin, Tim Duignan, and Mark Neumann. Orb-v3: atomistic simulation at scale, 2025. URL https://arxiv.org/abs/2504.06231.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. Mathematical discoveries from program search with large language models. *Nature*, 625(7995):468–475, 2024. doi: 10.1038/s41586-023-06924-6. URL https://doi.org/10.1038/s41586-023-06924-6.

Sukriti Singh and José Miguel Hernández-Lobato. Deep kernel learning for reaction outcome prediction and optimization. *Communications Chemistry*, 7:136, 2024. doi: 10.1038/s42004-024-01219-x. URL https://doi.org/10.1038/s42004-024-01219-x.

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, volume 25, 2012. URL https://proceedings.neurips.cc/paper/2012/hash/05311655a15b75fab86956663e1819cd-Abstract.html.

Haoyang Su, Renqi Chen, Shixiang Tang, Zhenfei Yin, Xinzhe Zheng, Jinzhe Li, Biqing Qi, Qi Wu, Hui Li, Wanli Ouyang, Philip Torr, Bowen Zhou, and Nanqing Dong. Many heads are better than one: Improved scientific idea generation by a LLM-based multi-agent system. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28201–28240. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1368. URL https://aclanthology.org/2025.acl-long.1368/.

Haorui Wang, Parshin Shojaee, Kazem Meidani, Kunyang Sun, José Miguel Hernández-Lobato, Teresa Head-Gordon, Jiajun He, Chandan K. Reddy, Chao Zhang, and Yuanqi Du. Towards diverse scientific hypothesis search with large language models, 2026. URL https://arxiv.org/abs/2606.10587.

Ruocheng Wang, Eric Zelikman, Gabriel Poesia, Yewen Pu, Nick Haber, and Noah D. Goodman. Hypothesis search: Inductive reasoning with language models. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/a934282496e7b907f5d48d49bb4c9d7d-Abstract-Conference.html.

Nicholas C. Wu, Lei Dai, C. Anders Olson, James O. Lloyd-Smith, and Ren Sun. Adaptation in protein fitness landscapes is facilitated by indirect paths. *eLife*, 5:e16965, 2016. doi: 10.7554/eLife.16965. URL https://doi.org/10.7554/eLife.16965.

Yuhuai Wu, Albert Qiaochu Jiang, Wenda Li, Markus N. Rabe, Charles Staats, Mateja Jamnik, and Christian Szegedy. Autoformalization with large language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-2344. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/d0c6bc641a56bebee9d985b937307367-Abstract-Conference.html.

Yutaro Yamada, Robert Tjarko Lange, Cong Lu, Shengran Hu, Chris Lu, Jakob Foerster, Jeff Clune, and David Ha. The AI scientist-v2: Workshop-level automated scientific discovery via agentic tree search, 2025. URL https://arxiv.org/abs/2504.08066.

Jason Yang, Ravi G. Lal, James C. Bowden, Raul Astudillo, Mikhail A. Hameedi, Sukhvinder Kaur, Matthew Hill, Yisong Yue, et al. Active learning-assisted directed evolution. *Nature Communications*, 16:714, 2025. doi: 10.1038/s41467-025-55987-8. URL https://doi.org/10.1038/s41467-025-55987-8.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.03629.

Fengxue Zhang, Jialin Song, James C. Bowden, Alexander Ladd, Yisong Yue, Thomas Desautels, and Yuxin Chen. Learning regions of interest for bayesian optimization with adaptive level-set estimation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 41579–41595. PMLR, 2023. URL https://proceedings.mlr.press/v202/zhang23aj.html.

CONTENTS

1 Introduction **1**

2 Constructing and Auditing Exploration Substrates **2**
2.1 Theoretical formulation . . . 2
2.2 Construct: content, form, and realization . . . 3
2.3 Audit: certification before exploration . . . 3
2.4 From construction to exploration . . . 4

3 Experiments **4**
3.1 Experimental logic and provenance . . . 4
3.2 Construction changes what reaches a shared consumer . . . 5
3.3 Organization changes access under controlled selection . . . 6
3.4 An activated substrate supports selective downstream reuse . . . 7

4 Related Work **9**

5 Conclusion **9**

A Notation and Formal Objects **16**
A.1 Tasks, Spaces, and Evidence . . . 16
A.2 Natural Language and Solution Space Objects . . . 16
A.3 Scientific Exploration Substrate . . . 16
A.4 Construction and Candidate Objects . . . 17
A.5 Verification and Activation . . . 17
A.6 Formal Semantic Objects and Theoretical Quantities . . . 17

B Formal Theory of Scientific Exploration Substrates **18**
B.1 Detailed Theoretical Setup . . . 18
B.2 Natural Language Solution Space . . . 19
B.3 Scientific Content and Local Consistency . . . 20
B.3.1 Preservation of Task Imposed Local Constraints . . . 21
B.4 Computational Form and Semantic and Computational Realization . . . 21
B.4.1 Semantic and Computational Realization . . . 22
B.4.2 Natural Language Interpretation . . . 22
B.4.3 Representation Family Openness . . . 23
B.5 Construction and Certification . . . 23
B.5.1 Verification Policy and Obligations . . . 23
B.5.2 Hybrid Audit . . . 23
B.5.3 Certified Scope . . . 24
B.5.4 Activation . . . 24
B.5.5 Revision After Failed Certification . . . 24
B.5.6 Fresh Certification Under Adaptive Revision . . . 25

B.5.7 Budgeted Termination 25
B.6 Preservation of Required Scientific Distinctions 25
B.7 Activation and Menu Neutrality 26
B.8 Explore, Lift, and Frozen Substrate 27
B.9 Boundary to Evidence Driven Evolution 28
B.10 Operational Mapping to Concrete Implementations 28

C Empirical Correspondence Between Theory and Mechanism **29**
C.1 Theory–Mechanism Correspondence 29
C.2 Hybrid Audit Creates an Authorization Boundary 29
C.3 Certified Scope Separates Recoverability from Authorization 30
C.4 Open and Menu Proposals Face the Same Activation Rule 30
C.5 Activated Revisions Use Fresh Certification 31
C.6 The Activated Substrate Remains Fixed During Exploration 31
C.7 Required Distinctions Are Nontrivial Activation Obligations 31
C.8 What These Checks Establish 31

D Complementary Empirical Analyses **33**
D.1 Complementary Analysis of Explicit Substrate Construction 34
D.1.1 Structural controls, adaptive reasoning, and difficult tasks 34
D.1.2 A localized contribution from explicit causal encoding 34
D.1.3 Full no-LLM paired effects 34
D.1.4 Representative ResearchBench cases 35
D.2 Complementary Analysis of Scientific Structure and Extreme-Solution Access . . . 35
D.2.1 Buchwald–Hartwig: the region advantage emerges with accumulated evidence 35
D.2.2 The value of organization depends on target concentration 37
D.2.3 GB1 terminal support reveals more than the mean 38
D.2.4 Same-environment intervention and equal-pool reachability 39
D.3 Complementary Analysis of Frozen-Substrate Reuse 40
D.3.1 A representative MADE discovery trajectory 40
D.3.2 Aggregate performance hides material-system heterogeneity 40
D.3.3 Breadth and polymorph depth can trade off 40
D.4 Generalization and Regimes of Usefulness 40
D.4.1 Seed-level external consumption in PiEvo 40
D.4.2 Reintroducing an LLM creates a proposer bottleneck 41
D.4.3 ResearchBench gains are heterogeneous across disciplines 41
D.5 How Objective and Setting Shape the Outcome 41
D.6 Summary of Complementary Findings 42

E Future Directions **44**

E.1 Closing the Construct–Explore–Lift–Evolve Loop 44

E.2 Evidence-Driven Substrate Evolution and Scientific Versioning 44

E.3 Multiple and Shared Substrates for Scientific Agents 44

E.4 Meta-Construction: When and How Should a Substrate Be Built? 45

F Models, Prompts, and Experimental Configurations **45**

F.1 Task and Evaluation Configurations 45

F.1.1 ResearchBench 46

F.1.2 HypoSpace 46

F.1.3 Buchwald–Hartwig 46

F.1.4 GB1 46

F.1.5 MADE 47

F.1.6 PiEvo 47

F.2 Baselines and Experimental Controls 47

F.2.1 ResearchBench 47

F.2.2 HypoSpace 47

F.2.3 Buchwald–Hartwig 48

F.2.4 GB1 48

F.2.5 MADE 48

F.2.6 PiEvo 49

F.3 Models and Scientific Hosts 49

F.4 Prompt and Interface Configurations 49

F.4.1 ResearchBench CASES workflow 49

F.4.2 ResearchBench expanded-output Direct control 50

F.4.3 HypoSpace controller 50

F.4.4 MADE readout and length-matched control 50

F.4.5 PiEvo guidance interface 50

F.5 Budgets, Pairing, and Failure Handling 50

F.6 Metrics and Statistical Conventions 51

F.7 Interpreting the Configurations 52

A Notation and Formal Objects

We summarize the notation used in Sec. 2 and Appendix B. Subscripts denote scientific tasks (τ), candidate specifications (χ), exploration steps (t), construction attempts (k), and scientific contexts (U). Hats denote proposed candidate components, tildes denote candidate substrates prior to certification, and the superscript A denotes certified operational scope. Symbols used only within individual proofs are defined locally.

A.1 Tasks, Spaces, and Evidence

Symbol	Meaning
τ	Scientific task identifier.
$\mathfrak{T}_\tau = (x_\tau, D_\tau, \mathcal{U}_\tau, \mathfrak{E}_\tau)$	Scientific task specification consisting of the scientific question, evidence available at construction time, available scientific alternatives, and evaluation environment.
x_τ	Scientific question associated with task τ.
D_τ	Evidence available when substrate construction begins.
$\mathcal{U}_\tau$	Scientific alternatives available to construction; may be natural language hypotheses, finite candidates, structured objects or records, or a combination.
$\mathfrak{E}_\tau$	Environment through which proposed hypotheses can subsequently be evaluated.
E_t	Evidence accumulated through exploration step t.

A.2 Natural Language and Solution Space Objects

Symbol	Meaning
$\mathcal{H}_\tau^{NL}$	Natural Language Scientific Hypothesis Space.
$R_t^{\mathrm{prom/sol}}$	Promising or solution bearing structure identified from accumulated evidence on the substrate.
$\mathcal{S}_t^{NL}$	Evidence conditioned Natural Language Solution Space.
Lift	Mapping from discovered substrate structure back into $\mathcal{H}_\tau^{NL}$.
Explore	Downstream exploration procedure operating on an activated substrate.
Evolve	General interface through which later scientific evidence may motivate substrate reconstruction or evolution.

A.3 Scientific Exploration Substrate

Symbol	Meaning
$S_\tau = (C_\tau, F_\tau, \Gamma_\tau, I_\tau)$	Scientific Exploration Substrate for task τ.
C_τ	Scientific Content specifying scientific contexts, distinctions, and consistency structure.
F_τ	Computational Form through which scientifically relevant variation becomes computationally accessible.
Γ_τ	Semantic and computational realization relation.
I_τ	Interpretation interface connecting natural language hypotheses to scientific semantic objects.
$\mathrm{Cap}(F_\tau)$	Executable capabilities supported by Computational Form.
$\mathbf{Z}_\tau$	Computational interface exposed by Computational Form.
$\mathbf{H}_\tau^{\mathrm{hyp}}$	Scientific hypothesis object in the formal semantic representation.

A.4 Construction and Candidate Objects

Symbol	Meaning
$\mathsf{Construct}_\tau$	Construction procedure returning a certified substrate or$\mathsf{ConstructionUnresolved}$.
$\mathfrak{X}_\tau$	Space of finite candidate specifications.
$\chi = (\widehat{C}_\chi, \widehat{F}_\chi, \widehat{\Gamma}_\chi, \widehat{I}_\chi, \mathcal{K}_\chi)$	Finite candidate specification.
$\mathcal{K}_\chi$	Scientific and operational claims declared by candidateχ.
$\mathcal{D}_\tau$	Partial denotation map from finite candidate specifications to formal substrate objects.
$\mathfrak{S}_\tau$	Codomain of substrate denotations for taskτ.
$\widetilde{S}_\chi = (C_\chi, F_\chi, \Gamma_\chi, I_\chi)$	Candidate substrate prior to certification.
r_k	Sanitized diagnostic returned after construction attempt$k$.
$\mathsf{Rev}_{\theta_{\mathrm{con}}}$	Candidate revision mechanism.
B_{con}	Maximum construction budget.
$\mathsf{ConstructionUnresolved}$	Termination state indicating that no admissible substrate was established within the registered construction procedure and budget.

A.5 Verification and Activation

Symbol	Meaning
$\mathcal{V}_\tau$	Verification policy specified independently of candidate proposals.
$\mathfrak{A}_\tau = (\Phi_\tau^{\mathrm{sep}}, \mathfrak{B}_\tau^{\mathrm{loc}}, \Omega_\tau^{\mathrm{op}})$	Task verification specification.
Φ_τ^{sep}	Templates specifying scientific distinctions that must remain separable.
$\mathfrak{B}_\tau^{\mathrm{loc}}$	Task requirements with local consistency semantics.
$\Omega_\tau^{\mathrm{op}}$	Mandatory operational and executability requirements.
$\mathcal{O}_{\tau,\chi}$	Full verification obligation set compiled for candidateχ.
$\mathcal{O}_{\tau,\chi}^{\mathrm{core}}$	Obligations imposed directly by task requirements.
$\mathcal{O}_{\tau,\chi}^{\mathrm{ind}}$	Additional obligations induced by candidate declarations.
$\mathcal{O}_{\tau,\chi}^{\mathrm{hard}}$	Mandatory obligations whose failure prevents activation.
$\mathcal{J}_{\tau,\chi}^{\mathrm{score}}$	Index set of scored audit criteria.
$q_j(\chi)$	Estimated audit score for criterion$j$.
$\underline{q}_j(\chi)$	Conservative audit score used for certification.
$\underline{\mathbf{q}}_\chi$	Vector of conservative audit scores.
ϑ_j	Registered minimum threshold for audit criterion$j$.
ϑ_{agg}	Registered aggregate activation threshold.
Agg_τ	Registered aggregation rule for scored audit criteria.
$\mathsf{HardPass}_\tau(\chi)$	Indicator that every mandatory obligation has been satisfied.
$\mathsf{Activate}_\tau(\chi)$	Activation decision for candidateχ.
$\mathbf{Z}_\chi^A$	Certified portion of the computational interface authorized for downstream use.
Γ_χ^A	Certified realization relation authorized for downstream use.

A.6 Formal Semantic Objects and Theoretical Quantities

Symbol	Meaning
$\mathcal{C}_\chi$	Small category of scientific contexts associated with candidateχ.
$\mathfrak{B}_{\tau,\chi}^{\mathrm{core}}$	Core local generating families derived from task requirements.
$\mathfrak{B}_{\tau,\chi}^{\mathrm{ind}}$	Additional local generating families induced by candidate declarations.

Symbol	Meaning
$J^{\mathrm{core}}_{\tau,\chi}$	Grothendieck topology generated by core local requirements.
$J_{\tau,\chi}$	Grothendieck topology generated jointly by core and candidate induced requirements.
$\mathscr{E}_{\tau,\chi} = \mathbf{Sh}(\mathcal{C}_\chi, J_{\tau,\chi})$	Semantic topos associated with candidateχ.
$\widehat{\mathcal{C}_\chi} = \mathbf{Set}^{\mathcal{C}_\chi^{op}}$	Presheaf category over$\mathcal{C}_\chi$.
$\mathcal{P}_{\mathrm{fin}}(A)$	Set of finite subsets of$A$.
$\perp_{\mathrm{interp}}$	Token representing unresolved interpretation.
$\Sigma_{\Phi,U}(h)$	Semantic signature induced by registered distinction probes in context$U$.
$\widehat{\Sigma}_{\Phi,U}(z)$	Computational signature induced by the corresponding certified probes.
$d_{\Phi,U}$	Pseudometric on the shared signature space.
ε_U	Certified realization discrepancy bound in context$U$.
γ_U	Minimum separation between distinct required semantic classes.

After activation, candidate subscripts are omitted when unambiguous. We distinguish throughout between a candidate specification χ, its denoted but uncertified substrate $\widetilde{S}_\chi$, and the certified substrate S_τ. Likewise, the proposed computational and realization interfaces $(\mathbf{Z}_\chi, \Gamma_\chi)$ are distinguished from their certified operational scopes $(\mathbf{Z}^A_\chi, \Gamma^A_\chi)$.

B Formal Theory of Scientific Exploration Substrates

This appendix defines the mathematical objects, construction assumptions, certification conditions, and guarantees summarized in Secs. 2.1–2.3.

The categorical semantics in this appendix is optional for implementing CASES: the operational protocol requires explicit content, form, and realization records together with executable audit obligations. The denotational objects here describe the scientific content, computational realization, and certified use of candidate substrates.

Throughout, categories of scientific contexts are assumed to be small. Set valued contextual interfaces are modeled as presheaves. Whenever stochastic explorers, execution environments, or operational capabilities require measurable structure, the corresponding measurable spaces are assumed to be standard Borel. No topology, metric, algebraic structure, finiteness, or enumerability assumption is imposed on the Natural Language Scientific Hypothesis Space unless stated explicitly.

B.1 Detailed Theoretical Setup

For scientific task τ, let

$$\mathfrak{T}_\tau = (x_\tau, D_\tau, \mathcal{U}_\tau, \mathfrak{E}_\tau),$$

where x_τ is the scientific question, D_τ denotes evidence available when construction begins, and $\mathfrak{E}_\tau$ denotes the environment through which hypotheses may subsequently be evaluated.

Let

$$\mathcal{H}^{NL}_\tau$$

be the Natural Language Scientific Hypothesis Space. We treat $\mathcal{H}^{NL}_\tau$ as a set of well formed scientific hypotheses rather than as a space equipped in advance with a computational geometry. In particular, the formulation does not assume that $\mathcal{H}^{NL}_\tau$ is finite, enumerable, metrizable, or supplied with a predefined neighborhood relation.

Let $\mathfrak{X}_\tau$ denote the space of finite candidate specifications. A candidate is written

$$\chi = (\widehat{C}_\chi, \widehat{F}_\chi, \widehat{\Gamma}_\chi, \widehat{I}_\chi, \mathcal{K}_\chi),$$

where $\mathcal{K}_\chi$ records the scientific and operational claims declared by the candidate.

A sufficiently specified candidate admits a partial denotation

$$\mathcal{D}_\tau : \mathfrak{X}_\tau \rightharpoonup \mathfrak{S}_\tau,$$

with

$$\mathcal{D}_\tau(\chi) = \widetilde{S}_\chi = (C_\chi, F_\chi, \Gamma_\chi, I_\chi).$$

The map is partial because an incomplete or internally ill formed finite specification need not determine the required formal objects.

Denotation and certification are distinct. Equation (B.5) states that a candidate denotes a mathematically defined substrate object; it does not imply that this object is scientifically admissible or authorized for downstream use. Certification of a denoted candidate is introduced separately in Sec. B.5.

Once a candidate has been certified and activated, we omit the candidate subscript when no ambiguity arises and write

$$S_\tau = (C_\tau, F_\tau, \Gamma_\tau, I_\tau)$$

for the resulting Scientific Exploration Substrate.

B.2 Natural Language Solution Space

CASES distinguishes four objects that play different roles in scientific exploration:

$$\mathcal{H}_\tau^{NL} \xrightarrow{\mathsf{Construct}} S_\tau \xrightarrow{\mathsf{Explore}} R_t^{\mathrm{prom/sol}} \xrightarrow{\mathsf{Lift}} \mathcal{S}_t^{NL} \subseteq \mathcal{H}_\tau^{NL}.$$

The first object, $\mathcal{H}_\tau^{NL}$, is the open hypothesis space from which scientific hypotheses may be proposed. The second, S_τ, is a task conditioned computational substrate constructed to expose scientifically relevant variation in an operational form. These two objects are deliberately distinct: constructing S_τ does not require enumerating $\mathcal{H}_\tau^{NL}$, nor does it identify the substrate with the hypothesis space itself.

During downstream exploration, let E_t denote the evidence accumulated through step t. The explorer operates on S_τ and identifies an evidence supported promising or solution bearing structure

$$R_t^{\mathrm{prom/sol}}.$$

The precise mathematical type of $R_t^{\mathrm{prom/sol}}$ is allowed to depend on the substrate and scientific task. It may, for example, correspond to a region, subset, collection of states, structured archive, or another task appropriate object exposed through the certified computational interface.

The Natural Language Solution Space at step t is defined by

$$\mathcal{S}_t^{NL} = \mathsf{Lift}\left(R_t^{\mathrm{prom/sol}}, E_t; S_\tau\right) \subseteq \mathcal{H}_\tau^{NL}.$$

Thus, $\mathcal{S}_t^{NL}$ contains the natural language hypotheses supported by the currently identified solution structure and accumulated evidence under the semantics supplied by S_τ.

The operator Lift is substrate dependent. Its role is not to invert Computational Form globally, but to interpret certified computational discoveries through the scientific content, realization relation, and interpretation interface of the activated substrate. Accordingly, no global bijection between $\mathcal{H}_\tau^{NL}$ and the computational interface is assumed.

This distinction is important when interpretation or realization is many valued, partial, or unresolved. Several natural language hypotheses may share a scientific realization, a scientific hypothesis may admit multiple computational realizations, and some natural language hypotheses may remain outside the currently certified scope. The theory therefore requires only task relative, certified correspondence rather than a global equivalence between language and computation.

For the experiments in the present work, the activated substrate is held fixed during an exploration episode:

$$S_{\tau,t} = S_\tau, \qquad \forall t.$$

Consequently, changes in $\mathcal{S}_t^{NL}$ during an episode arise from newly accumulated evidence and changes in $R_t^{\mathrm{prom/sol}}$, rather than from changes to the substrate itself. Continual evolution of S_τ in response to later scientific evidence is treated separately as the broader Evolve interface.

B.3 Scientific Content and Local Consistency

Using standard category- and sheaf-theoretic terminology (Mac Lane & Moerdijk, 1992), the Scientific Content component C_χ specifies the semantic organization that a candidate substrate proposes to preserve. Formally, C_χ determines a small category of scientific contexts

$$\mathcal{C}_\chi,$$

whose objects represent scientific contexts relevant to the task and whose morphisms encode admissible restriction or contextual transition relations.

Among the scientific requirements associated with task τ, some admit genuinely local consistency semantics. Instantiating these requirements against candidate χ produces a collection of core generating families

$$\mathfrak{B}_{\tau,\chi}^{\mathrm{core}}.$$

Candidate declarations may introduce additional local compatibility commitments, producing

$$\mathfrak{B}_{\tau,\chi}^{\mathrm{ind}}.$$

For a family collection $\mathfrak{B}$ over a small category $\mathcal{C}$, define

$$J(\mathfrak{B}) = \bigcap \left\{ J : \begin{array}{l} J \text{ is a Grothendieck topology on } \mathcal{C}, \\ \langle b \rangle_{\mathrm{sieve}} \in J(U) \\ \text{for every } b \\ b \in \mathfrak{B}(U) \end{array} \right\},$$

where $\langle b \rangle_{\mathrm{sieve}}$ denotes the sieve generated by family b. Thus, $J(\mathfrak{B})$ is the coarsest Grothendieck topology under which every family in $\mathfrak{B}$ is covering.

The topology induced solely by requirements fixed by the task is

$$J_{\tau,\chi}^{\mathrm{core}} = J\left(\mathfrak{B}_{\tau,\chi}^{\mathrm{core}}\right),$$

while the topology incorporating both task requirements and candidate induced commitments is

$$J_{\tau,\chi} = J\left(\mathfrak{B}_{\tau,\chi}^{\mathrm{core}} \cup \mathfrak{B}_{\tau,\chi}^{\mathrm{ind}}\right).$$

The corresponding semantic environment is the topos

$$\mathscr{E}_{\tau,\chi} = \mathbf{Sh}\left(\mathcal{C}_\chi, J_{\tau,\chi}\right).$$

For each semantic sort $\lambda \in \Lambda_\chi$, let

$$\mathbf{H}_\chi^\lambda \in \mathscr{E}_{\tau,\chi}$$

denote the associated semantic object. If $\Lambda_\chi^{\mathrm{hyp}} \subseteq \Lambda_\chi$ indexes the sorts corresponding to scientific hypotheses, we define the aggregate candidate hypothesis object by

$$\mathbf{H}_\chi^{\mathrm{hyp}} = \coprod_{\lambda \in \Lambda_\chi^{\mathrm{hyp}}} \mathbf{H}_\chi^\lambda.$$

Scientific relations may be represented as subobjects of finite products of these semantic objects, while scientifically meaningful compositions are represented by morphisms in $\mathscr{E}_{\tau,\chi}$.

The use of a Grothendieck topology is restricted to requirements that genuinely admit local compatibility and gluing semantics. Global scientific distinctions, executable contracts, fidelity requirements, and other properties that are not naturally local are not encoded as covering conditions. They remain explicit verification obligations and are handled by the certification machinery developed in Sec. B.5.

B.3.1 Preservation of Task Imposed Local Constraints

The distinction between $J^{\mathrm{core}}_{\tau,\chi}$ and $J_{\tau,\chi}$ captures an important asymmetry: a candidate may introduce additional scientific structure, but it may not remove local consistency requirements already imposed by the task.

Lemma B.1 (Monotonicity of Generated Topologies). If $\mathfrak{B} \subseteq \mathfrak{B}'$, then

$$J(\mathfrak{B}) \subseteq J(\mathfrak{B}').$$

Proof. Every Grothendieck topology containing the covering generators in $\mathfrak{B}'$ also contains those in $\mathfrak{B}$. Since $J(\mathfrak{B})$ is the coarsest Grothendieck topology containing the generators in $\mathfrak{B}$, while $J(\mathfrak{B}')$ is one such topology, the inclusion follows:

$$J(\mathfrak{B}) \subseteq J(\mathfrak{B}').$$

□

Proposition B.1 (Preservation of Task Imposed Local Constraints). Suppose the task requirements represented by $\mathfrak{B}^{\mathrm{core}}_{\tau,\chi}$ are instantiable on candidate χ. Then

$$J^{\mathrm{core}}_{\tau,\chi} \subseteq J_{\tau,\chi}.$$

Moreover, the identity on the underlying presheaf category restricts to a full inclusion

$$\mathbf{Sh}\left(\mathcal{C}_\chi, J_{\tau,\chi}\right) \hookrightarrow \mathbf{Sh}\left(\mathcal{C}_\chi, J^{\mathrm{core}}_{\tau,\chi}\right).$$

Proof. By construction,

$$\mathfrak{B}^{\mathrm{core}}_{\tau,\chi} \subseteq \mathfrak{B}^{\mathrm{core}}_{\tau,\chi} \cup \mathfrak{B}^{\mathrm{ind}}_{\tau,\chi}.$$

Lemma B.1 therefore gives

$$J^{\mathrm{core}}_{\tau,\chi} \subseteq J_{\tau,\chi}.$$

Since $J_{\tau,\chi}$ contains every cover required by $J^{\mathrm{core}}_{\tau,\chi}$, any sheaf satisfying the matching and gluing conditions for $J_{\tau,\chi}$ also satisfies those required by $J^{\mathrm{core}}_{\tau,\chi}$. The morphisms are the same natural transformations between the underlying presheaves, yielding the full inclusion in Eq. (B.22). □

Proposition B.1 formalizes the preservation property summarized in Sec. 2.2. Candidate induced scientific structure may strengthen or refine the local consistency conditions associated with the task, but it cannot remove those conditions from the semantic organization of an admissible candidate.

B.4 Computational Form and Semantic and Computational Realization

Scientific Content specifies which scientific distinctions and relations are meaningful for the task. Computational Form specifies how such structure becomes accessible to systematic computation.

For candidate χ, let F_χ denote its Computational Form and let

$$\mathrm{Cap}(F_\chi)$$

denote the executable capabilities declared by that form. These capabilities may include task appropriate operations such as traversal, comparison, composition, sampling, optimization, simulation, or evaluation. Their exact type is not fixed by the theory and depends on the scientific task and the representation proposed by the candidate.

The contextual interface exposed by F_χ is modeled as a presheaf

$$\mathbf{Z}_\chi \in \widehat{\mathcal{C}_\chi} := \mathbf{Set}^{\mathcal{C}_\chi^{op}}.$$

For each scientific context $U \in \mathcal{C}_\chi$, the set

$$\mathbf{Z}_\chi(U)$$

contains the computational objects exposed by the candidate in that context, while restriction maps specify how these objects behave when scientific context is restricted.

Crucially, F_χ itself is not required to be an object of the semantic topos

$\mathscr{E}_{\tau,\chi} = \mathbf{Sh}(\mathcal{C}_\chi, J_{\tau,\chi})$.

Only its task relevant contextual interface must be exposed through $\mathbf{Z}_\chi$. This separation allows Computational Form to instantiate different internal representation families while retaining a common formal interface through which scientific semantics and verification requirements can be expressed.

B.4.1 Semantic and Computational Realization

Let

$\iota_{\tau,\chi} : \mathbf{Sh}(\mathcal{C}_\chi, J_{\tau,\chi}) \hookrightarrow \widehat{\mathcal{C}_\chi}$

denote the canonical inclusion of sheaves into presheaves. Viewing the candidate hypothesis object through this inclusion, the semantic and computational realization relation is a subpresheaf

$\Gamma_\chi \hookrightarrow \iota_{\tau,\chi}\big(\mathbf{H}_\chi^{\mathrm{hyp}}\big) \times \mathbf{Z}_\chi$.

When no ambiguity arises, we suppress $\iota_{\tau,\chi}$ and use the shorter notation

$\Gamma_\chi \hookrightarrow \mathbf{H}_\chi^{\mathrm{hyp}} \times \mathbf{Z}_\chi$.

For context U, a pair

$(h, z) \in \Gamma_{\chi,U}$

states that computational object $z \in \mathbf{Z}_\chi(U)$ is proposed as an admissible realization of semantic hypothesis object $h \in \mathbf{H}_\chi^{\mathrm{hyp}}(U)$. Because Γ_χ is a subpresheaf, these proposed correspondences are stable under the contextual restriction maps inherited from the two factors.

The relation Γ_χ is not assumed to be functional, injective, or surjective. A scientific object may admit several computational realizations, several scientific objects may map to related computational objects before certification, and parts of either side may remain unrealized. The role of verification is precisely to determine which of these proposed correspondences are sufficiently faithful and operationally supported for scientific use.

B.4.2 Natural Language Interpretation

Natural language hypotheses enter the semantic organization through the interpretation interface. At context U, define

$I_{\chi,U} : \mathcal{H}_\tau^{NL} \longrightarrow \mathcal{P}_{\mathrm{fin}}\left(\mathbf{H}_\chi^{\mathrm{hyp}}(U)\right) \cup \{\perp_{\mathrm{interp}}\}$.

A finite image

$I_{\chi,U}(h^{NL}) = \{h_1, \ldots, h_m\}$

represents the candidate semantic interpretations admitted for natural language hypothesis h^{NL} in context U. The value

$I_{\chi,U}(h^{NL}) = \perp_{\mathrm{interp}}$

records that no interpretation has yet been established.

The interpretation interface is therefore allowed to be ambiguous or unresolved. CASES does not assume a general solution to natural language understanding, nor does it require every hypothesis in $\mathcal{H}_\tau^{NL}$ to receive a unique semantic grounding. Cross context coherence, interpretation fidelity, and the admissibility of proposed groundings are instead treated as explicit verification requirements.

Combining interpretation and realization gives the candidate level chain

$h^{NL} \xrightarrow{I_{\chi,U}} h \xrightarrow{\Gamma_{\chi,U}} z$,

where

$h \in I_{\chi,U}(h^{NL}), \qquad (h, z) \in \Gamma_{\chi,U}$.

This chain formalizes the connection between open natural language hypotheses, scientific semantic objects, and computational realizations without identifying any two of these levels.

B.4.3 Representation Family Openness

The formal role of Computational Form is therefore interface based rather than architecture based. The theory constrains the scientific and operational properties that an admissible form must expose and satisfy, but does not require its internal representation to belong to a fixed metric, geometric, graphical, symbolic, probabilistic, or other predefined family.

Formally, different internal forms F_χ may induce different contextual interfaces $\mathbf{Z}_\chi$ and different realization relations Γ_χ. Their admissibility is determined by the task requirements and certification procedure rather than by membership in a prespecified representation class. Representation freedom is thus separated from scientific authority: a candidate may propose a new computational form, but only its subsequently certified scope is permitted to enter downstream scientific exploration.

B.5 Construction and Certification

A candidate substrate becomes available for scientific use only after it satisfies a verification policy specified independently of that candidate. This separation ensures that the mechanism proposing a representation does not simultaneously determine the conditions under which that representation is accepted.

B.5.1 Verification Policy and Obligations

For task τ, let

$$\mathcal{V}_\tau$$

denote the verification policy. Before candidate construction, $\mathcal{V}_\tau$ fixes the task verification specification

$$\mathfrak{A}_\tau = \left(\Phi_\tau^{\mathrm{sep}}, \mathfrak{B}_\tau^{\mathrm{loc}}, \Omega_\tau^{\mathrm{op}}\right),$$

where Φ_τ^{sep} specifies scientific distinctions that must remain separable, $\mathfrak{B}_\tau^{\mathrm{loc}}$ specifies requirements admitting local consistency semantics, and $\Omega_\tau^{\mathrm{op}}$ specifies mandatory operational and executability requirements.

Instantiating these task requirements against candidate χ produces the core obligation set

$$\mathcal{O}_{\tau,\chi}^{\mathrm{core}} = \mathsf{CompileCore}_\tau\left(\mathfrak{A}_\tau, \chi\right).$$

Candidate declarations $\mathcal{K}_\chi$ may introduce additional scientific or operational commitments. These induce

$$\mathcal{O}_{\tau,\chi}^{\mathrm{ind}} = \mathsf{CompileInduced}_\tau\left(\mathcal{V}_\tau, \chi, \mathcal{K}_\chi\right).$$

The complete verification challenge is therefore

$$\mathcal{O}_{\tau,\chi} = \mathcal{O}_{\tau,\chi}^{\mathrm{core}} \cup \mathcal{O}_{\tau,\chi}^{\mathrm{ind}}.$$

A task requirement that cannot be instantiated or resolved against a candidate remains an unresolved verification obligation. It is not removed merely because the candidate fails to represent it. Candidate declarations may therefore add requirements, but they cannot eliminate requirements fixed by $\mathcal{V}_\tau$.

B.5.2 Hybrid Audit

The verifier evaluates $\mathcal{O}_{\tau,\chi}$ along four complementary dimensions:

scientific admissibility, structural validity, semantic fidelity, executability.

The audit is hybrid because different claims support different forms of verification. Scientific and semantic claims may require structured semantic evaluation, whereas structural constraints, declared invariants, executable operations, and other formally checkable properties are evaluated deterministically whenever an explicit checker is available.

Let

$$\mathcal{J}_{\tau,\chi}^{\mathrm{score}}$$

index the scored audit criteria. For each $j \in \mathcal{J}^{\mathrm{score}}_{\tau,\chi}$, let $q_j(\chi)$ denote its estimated audit score and $\underline{q}_j(\chi)$ the conservative score used for certification. We collect the latter as

$$\underline{\mathbf{q}}_\chi = \Big(\underline{q}_j(\chi)\Big)_{j \in \mathcal{J}^{\mathrm{score}}_{\tau,\chi}}.$$

The formal theory does not require a particular construction of the conservative score. Any confidence or uncertainty interpretation attached to $\underline{q}_j$ must be justified by the corresponding evaluator and calibration assumptions in the concrete implementation.

B.5.3 Certified Scope

Verification need not authorize every computational object proposed by F_χ or every correspondence proposed by Γ_χ. Instead, the audit identifies restriction stable certified subinterfaces

$$\mathbf{Z}^A_\chi \hookrightarrow \mathbf{Z}_\chi, \qquad \Gamma^A_\chi \hookrightarrow \Gamma_\chi \cap \big(\mathbf{H}^{\mathrm{hyp}}_\chi \times \mathbf{Z}^A_\chi\big).$$

The pair

$$\big(\mathbf{Z}^A_\chi, \Gamma^A_\chi\big)$$

is the certified operational scope of candidate χ. Computational recoverability outside this scope does not by itself authorize scientific use.

B.5.4 Activation

Let

$$\mathcal{O}^{\mathrm{hard}}_{\tau,\chi} \subseteq \mathcal{O}_{\tau,\chi}$$

denote the mandatory obligations, and define

$$\mathsf{HardPass}_\tau(\chi) = 1$$

when all obligations in $\mathcal{O}^{\mathrm{hard}}_{\tau,\chi}$ are satisfied.

For registered criterion thresholds ϑ_j, aggregation rule Agg_τ, and aggregate threshold ϑ_{agg}, candidate activation is defined by

$$\mathsf{Activate}_\tau(\chi) = 1 \iff \begin{cases} \mathsf{HardPass}_\tau(\chi) = 1, \\ \underline{q}_j(\chi) \geq \vartheta_j, \quad \forall j \in \mathcal{J}^{\mathrm{score}}_{\tau,\chi}, \\ \mathsf{Agg}_\tau(\underline{\mathbf{q}}_\chi) \geq \vartheta_{\mathrm{agg}}. \end{cases}$$

Thus, every mandatory obligation and every registered minimum criterion must be satisfied. Strong performance on other audit dimensions does not compensate for failure of a required condition.

For later use, let

$$v_\tau(\chi)$$

denote the verification record containing the registered obligation outcomes, certified scope, and scored quantities used by the activation rule. The dependence of activation on this record, rather than on a representation family label, is formalized in Sec. B.7.

B.5.5 Revision After Failed Certification

Failure to activate does not imply that the proposed representation family is intrinsically invalid. The audit may instead identify missing specifications, unsupported claims, fidelity failures, structural inconsistencies, or insufficient evidence.

After construction attempt k, let a_k denote the audit record. A sanitized diagnostic

$$r_k = \mathsf{Sanitize}(a_k)$$

is returned to the constructor. The diagnostic may expose failed obligations and unresolved properties, but it does not prescribe a replacement representation family.

The next candidate is sampled or generated according to

$$\chi_{k+1} \sim \mathsf{Rev}_{\theta_{\mathrm{con}}}\left(\cdot \mid \mathfrak{T}_\tau, \mathcal{H}_\tau^{NL}, \chi_k, r_k\right).$$

Revision may alter

$$C_\chi, \qquad F_\chi, \qquad \Gamma_\chi, \qquad I_\chi,$$

and may therefore change the proposed representation family itself. The verifier identifies what has not been established; it does not determine which architecture must replace the failed candidate.

B.5.6 Fresh Certification Under Adaptive Revision

Because audit diagnostics influence later candidates, evidence used during revision may no longer provide an independent basis for final certification. We therefore distinguish developmental audit information from the evidence used to authorize the final candidate.

Let

$$\mathsf{FreshCert}(a_k) = 1$$

indicate that the activation decision for attempt k is supported by a certification procedure whose validity remains justified after the preceding adaptive revisions. In concrete implementations this may be achieved through fresh held out evidence or another certification procedure with appropriate conditional validity.

A candidate is returned as the task substrate only when both

$$\mathsf{Activate}_\tau(\chi_k) = 1 \qquad \text{and} \qquad \mathsf{FreshCert}(a_k) = 1.$$

B.5.7 Budgeted Termination

Let $B_{\mathrm{con}} < \infty$ denote the registered construction budget. The construction process terminates with either an activated substrate or an unresolved result:

$$\mathsf{Construct}_\tau\left(\mathfrak{T}_\tau, \mathcal{H}_\tau^{NL}\right) \in \{S_\tau, \mathsf{ConstructionUnresolved}\}.$$

If no candidate satisfies the certification requirements within the construction budget, the procedure returns

$$\mathsf{ConstructionUnresolved}.$$

This state means that the registered construction and verification procedure has not established an admissible substrate under the available resources and evidence. It does not assert that no admissible Scientific Exploration Substrate exists for the task.

The complete construction loop is therefore

$$\mathsf{Propose} \to \mathsf{Denote} \to \mathsf{Compile} \to \mathsf{Audit} \to \begin{cases} \mathsf{Activate}, & \text{if certification succeeds,} \\ \mathsf{Revise}, & \text{otherwise.} \end{cases}$$

Revision continues only within the finite construction budget.

B.6 Preservation of Required Scientific Distinctions

A central role of certification is to prevent scientific distinctions required by the task from being erased by the constructed computational representation. We formalize this requirement only for distinctions explicitly registered by the verification policy.

Fix a scientific context U. Let

$$\Phi_U = \left\{\phi_{i,U} : \mathbf{H}_\chi^{\mathrm{hyp}}(U) \to Y_{i,U}\right\}_{i \in \mathcal{I}_{\Phi,U}}$$

denote the semantic distinction probes compiled from Φ_τ^{sep}, and let

$$\widehat{\Phi}_U = \left\{ \widehat{\phi}_{i,U} : \mathbf{Z}_\chi^A(U) \to Y_{i,U} \right\}_{i \in \mathcal{I}_{\Phi,U}}$$

be the corresponding probes on the certified computational scope.

They induce semantic and computational signatures

$$\Sigma_{\Phi,U}(h) = \big(\phi_{i,U}(h)\big)_{i \in \mathcal{I}_{\Phi,U}}, \qquad \widehat{\Sigma}_{\Phi,U}(z) = \big(\widehat{\phi}_{i,U}(z)\big)_{i \in \mathcal{I}_{\Phi,U}},$$

measured in a shared signature space equipped with pseudometric $d_{\Phi,U}$.

Two semantic hypotheses are considered equivalent with respect to the registered distinctions when their signatures have zero distance:

$$h \sim_{\Phi,U} h' \iff d_{\Phi,U}\left(\Sigma_{\Phi,U}(h), \Sigma_{\Phi,U}(h')\right) = 0.$$

Let $\mathcal{C}_{\mathrm{req}}(U)$ denote the task required equivalence classes under this relation. When at least two such classes are present, define their separation by

$$\gamma_U = \inf_{\substack{C \neq C' \\ C, C' \in \mathcal{C}_{\mathrm{req}}(U)}} d_{\Phi,U}\left(\Sigma_{\Phi,U}(h_C), \Sigma_{\Phi,U}(h_{C'})\right),$$

where h_C and $h_{C'}$ are representatives of the corresponding signature classes.

Theorem B.2 (Preservation of Registered Required Distinctions). Assume that every required class in $\mathcal{C}_{\mathrm{req}}(U)$ has at least one realization in the certified scope. Suppose further that every certified pair

$$(h, z) \in \Gamma_{\chi,U}^A$$

satisfies

$$d_{\Phi,U}\left(\Sigma_{\Phi,U}(h), \widehat{\Sigma}_{\Phi,U}(z)\right) \leq \varepsilon_U.$$

If

$$\gamma_U > 2\varepsilon_U,$$

then no $z \in \mathbf{Z}_\chi^A(U)$ can be a certified realization of hypotheses belonging to two distinct required classes.

Proof. Suppose that a certified computational state z realizes $h \in C$ and $h' \in C'$, where $C \neq C'$. By the triangle inequality,

$$\begin{aligned} d_{\Phi,U}\left(\Sigma_{\Phi,U}(h), \Sigma_{\Phi,U}(h')\right) &\leq d_{\Phi,U}\left(\Sigma_{\Phi,U}(h), \widehat{\Sigma}_{\Phi,U}(z)\right) \\ &\quad + d_{\Phi,U}\left(\widehat{\Sigma}_{\Phi,U}(z), \Sigma_{\Phi,U}(h')\right) \\ &\leq 2\varepsilon_U. \end{aligned}$$

This contradicts the required class separation $\gamma_U > 2\varepsilon_U$. □

Theorem B.2 protects the distinctions registered by $\mathcal{V}_\tau$ within the certified operational domain. Other scientific distinctions would require additional obligations. The experimental artifacts check required distinction coverage at activation; they do not estimate the realization discrepancy ε_U or separation margin $\gamma_U - 2\varepsilon_U$. Appendix C reports coverage, while Eq. (B.61) gives the formal sufficient condition for noncollapse.

B.7 Activation and Menu Neutrality

CASES allows Computational Form to vary across representation families, while activation is determined by the verification requirements established in Sec. B.5. We formalize this separation between representation proposal and activation authority.

Let $\mathcal{M}$ denote any reference collection of representation families, and let

$m_{\mathcal{M}}(\chi)$

denote the representation family label, or membership record, assigned to candidate χ.

Let the activation relevant verification record be

$$v_\tau(\chi) = \left(\mathsf{HardPass}_\tau(\chi), \left\{ \underline{q}_j(\chi) \right\}_{j \in \mathcal{J}^{\mathrm{score}}_{\tau,\chi}} \right).$$

Because the registered thresholds and aggregation rule are fixed by $\mathcal{V}_\tau$, Eq. (B.45) defines a policy function a_τ such that

$$\mathsf{Activate}_\tau(\chi) = a_\tau\left(v_\tau(\chi)\right).$$

We call the activation policy **menu neutral** when $m_{\mathcal{M}}(\chi)$ is not an argument of a_τ.

Proposition B.3 (Menu Neutrality of Activation). For two candidates χ_1 and χ_2, if

$$v_\tau(\chi_1) = v_\tau(\chi_2),$$

then

$$\mathsf{Activate}_\tau(\chi_1) = \mathsf{Activate}_\tau(\chi_2),$$

regardless of their representation family labels.

Proof. Both candidates are evaluated by the same activation function a_τ. Equal verification records therefore produce equal activation decisions. Since the representation family label is not an argument of a_τ, changing that label while holding the verification record fixed cannot change the decision. □

Proposition B.3 formalizes the sense in which CASES prescribes verification rather than representation. The verifier may impose scientific, semantic, structural, and operational requirements, but it does not grant or deny activation merely because a candidate belongs to a particular representation family.

Menu neutrality concerns the activation rule. Proposal diversity, verification pass rates, and downstream utility depend on the constructor and task and are measured separately.

B.8 Explore, Lift, and Frozen Substrate

The experiments in this work are designed to isolate the consequences of substrate construction from continual representation change. Once a candidate is certified and activated, the Scientific Exploration Substrate is held fixed throughout the corresponding exploration episode:

$$S_{\tau,t} = S_\tau, \qquad \forall t.$$

Let

$$S_\tau = (C_\tau, F_\tau, \Gamma_\tau, I_\tau)$$

be the activated substrate. At step t, downstream exploration may update accumulated evidence E_t and the promising or solution bearing structure $R_t^{\mathrm{prom/sol}}$, but it does not modify C_τ, F_τ, Γ_τ, or I_τ.

To make the lifting operation explicit, let $\mathbf{P}_t$ denote the portion of the certified computational interface currently identified as promising or solution bearing, with

$$\mathbf{P}_t \hookrightarrow \mathbf{Z}^A_\tau.$$

Its semantic projection is

$$R_t^{\mathrm{sem}}(U) = \left\{ h \in \mathbf{H}^{\mathrm{hyp}}_\tau(U) : \exists z \in \mathbf{P}_t(U),\ (h,z) \in \Gamma^A_{\tau,U} \right\}.$$

A corresponding natural language solution set may then be read out as

$$\mathcal{S}_t^{NL} = \left\{ h^{NL} \in \mathcal{H}_\tau^{NL} : \exists U \in \mathcal{C}_\tau,\ I_{\tau,U}(h^{NL}) \neq \bot_{\mathrm{interp}},\ I_{\tau,U}(h^{NL}) \cap R_t^{\mathrm{sem}}(U) \neq \varnothing \right\}.$$

Equation (B.71) is one formal realization of the abstract Lift operator used in Sec. 2.1. Concrete tasks may expose task specific readouts, provided they remain consistent with the certified realization and interpretation interfaces.

Proposition B.4 (Substrate Invariance During Frozen Exploration). Assume that an exploration episode updates only evidence and downstream discovery state, that all computational objects used by the explorer lie within the certified scope, and that the explorer has no operation that mutates C_τ, F_τ, Γ_τ, or I_τ. Then

$$S_{\tau,t} = S_\tau \qquad \forall t.$$

Proof. At $t = 0$, the activated substrate is S_τ. By assumption, every valid exploration transition updates only evidence or downstream discovery state and assigns no new value to any component of S_τ. Therefore, if $S_{\tau,t} = S_\tau$, then $S_{\tau,t+1} = S_\tau$. The claim follows by induction. □

This proposition describes the fixed-substrate episodes used in the experiments. Substrate evolution across episodes is discussed next.

B.9 Boundary to Evidence Driven Evolution

The broader CASES program allows new experimental evidence to motivate changes to the substrate itself:

$$S_{\tau,t} \xrightarrow[\text{new scientific evidence}]{\text{Evolve}} S_{\tau,t+1}.$$

Such evolution may revise scientific distinctions, computational form, realization relations, or interpretation interfaces. This process is conceptually different from the preactivation revision loop in Sec. B.5. Preactivation revision searches for a candidate that can be certified under the current task specification; evidence driven evolution would reconsider an already activated substrate after downstream scientific evidence reveals new structure, limitations, or mismatches.

The experiments study construction and certification with a fixed downstream substrate, which isolates the effect of its organization during each episode. Equation (B.73) outlines a future evolution step whose algorithm and guarantees remain open.

B.10 Operational Mapping to Concrete Implementations

The formal objects above provide semantics for finite candidate specifications and certified runtime interfaces. The experimental implementations operate on concrete schemas, graphs, structured records, executable operators, and compiled checks rather than explicitly materializing ambient categorical constructions.

Theoretical construct	Operational implementation
C_χ	Declared scientific contexts, hypotheses, relations, and required distinctions.
F_χ	Computational carriers, neighborhood or structural relations, executable operators, and runtime interfaces.
Γ_χ	Declared correspondences between scientific objects and computational realizations.
I_χ	Interpretation rule and ambiguity policy connecting natural language hypotheses to scientific objects.
$\mathcal{K}_\chi$	Scientific and operational claims declared by the candidate.
$\mathcal{O}^{\mathrm{core}}_{\tau,\chi}$	Obligations compiled from task requirements.
$\mathcal{O}^{\mathrm{ind}}_{\tau,\chi}$	Additional obligations induced by candidate declarations.
$\mathbf{Z}^A_\chi, \Gamma^A_\chi$	Audit certified computational handles and realization scope authorized for downstream scientific use.
r_k	Sanitized diagnostic used to guide candidate revision.
FreshCert	Requirement that final activation be supported by a certification procedure whose validity remains justified after adaptive revision.

Theoretical construct	Operational implementation
$S_{\tau,t} = S_\tau$	Runtime invariance of the activated substrate during the frozen exploration episodes studied here.

The categorical layer formalizes task conditioned scientific meaning and contextual compatibility. Concrete implementations use schemas, graphs, structured records, and executable operators; Appendix C examines the corresponding operational mechanisms. Each guarantee applies under its stated assumptions and certified scope.

C Empirical Correspondence Between Theory and Mechanism

Appendix B develops the formal properties of Scientific Exploration Substrates. Here we examine their operational counterparts in frozen execution artifacts: registered obligations, audit decisions, authorized scope, revision, and runtime identity. The records establish which mechanisms were implemented and observed; they do not estimate unrecorded theorem margins.

C.1 Theory–Mechanism Correspondence

Table 14: Correspondence between formal substrate properties, operational mechanisms, and frozen empirical checks.

Formal object or property	Operational realization	Empirical question	Frozen evidence	Supported scope
Candidate substrate (C_χ, F_χ) (Γ_χ, I_χ)	Declared scientific distinctions; computational carriers, operators, and interfaces; realization records; interpretation and ambiguity policies	Are the components of a proposed substrate represented by explicit operational artifacts?	Construct records and compiled candidate specifications	Inspectable components under the recorded procedure
Hybrid Audit and activation	Registered scientific, semantic, structural, and executable checks	Does proposal face a nontrivial authorization boundary?	Corruption tests, risk–coverage records, construction funnel	Selective authorization under the registered policy
Certified scope$(\mathbf{Z}^A_\chi, \Gamma^A_\chi)$	Only handles and realization relations authorized by the registered policy are exposed downstream	Is computational recoverability distinct from authorization?	Governed-recovery experiments	Recoverability does not imply authorization under the evaluated policy
Menu-neutral activation	Representation-family labels are not inputs to activation once the verification record is fixed	Does proposal freedom imply an intrinsic acceptance advantage?	Open versus menu-constrained construction	Common activation rule across proposal families
Revision and fresh certification	Sanitized diagnostics may guide revision; activation requires a fresh certification suite	Does adaptive revision bypass or reuse its final authorization test?	Attempt traces and certification records	Integrity of the recorded revision procedure
Frozen runtime$S_{\tau,t} = S_\tau$	Substrate identity is recorded after activation and prohibited mutations are monitored	Does downstream exploration modify the activated organization?	Substrate-hash and mutation logs	Invariance for logged runs
Required distinction preservation	Required distinctions are compiled into activation obligations	Are required distinctions covered by activated candidates?	Required distinction audit records	Coverage at activation; not the noncollapse margin of Theorem B.2

The categorical semantics in Appendix B describe contextual meaning and compatibility. The implementation exposes the corresponding scientific distinctions, computational capabilities, and realization commitments through ordinary operational artifacts.

C.2 Hybrid Audit Creates an Authorization Boundary

The frozen corruption study injects 12 corrupted candidates; none survives the registered audit condition:

12 corrupted candidates $\longrightarrow$ 0 surviving the registered Audit gate.

The risk–coverage study evaluates 18 operating points with $n = 11$ repeated trials per point. A sampled construction funnel further shows rejection at multiple stages:

55 generated $\rightarrow$ 40 schema-valid $\rightarrow$ 40 hybrid-structure $\rightarrow$ 35 audited $\rightarrow$ 7 auditable without a Fail outcome.

The corruption test and construction funnel show that proposals pass through a substantive authorization gate. Activation remains relative to the registered verification policy and available evidence.

C.3 Certified Scope Separates Recoverability from Authorization

Across 48 governed-recovery episodes, governed CASES produces valid governed recovery in

$$40/48 = 83.3\%$$

of all episodes and in

$$40/40 = 100\%$$

of the eligible subset. The remaining eight episodes are governance-blocked; no unauthorized recovery is produced in these episodes.

By contrast, greedy recovery, HS-tree recovery, and the corresponding ungoverned CASES condition each recover all eight blocked cases computationally. Those recoveries are unauthorized, giving valid governed recovery $32/48 = 66.7\%$.

Thus,

computationally recoverable $\not\Rightarrow$ authorized for scientific use.

This is the operational counterpart of exposing only $\mathbf{Z}_\chi^A$ and Γ_χ^A downstream.

C.4 Open and Menu Proposals Face the Same Activation Rule

Open construction produces $2/68 = 2.9\%$ passing candidates, while menu-constrained construction produces $2/60 = 3.3\%$:

$$2/68 = 2.9\%, \qquad 2/60 = 3.3\%.$$

The similar pass rates show that open proposal does not receive preferential activation.

A separately recorded best-of-three comparison yields 5 wins, 1 loss, and 54 ties for open proposals relative to menu proposals. A two-sided sign test over the six discordant pairs gives

$$p \approx .22.$$

The best-of-three comparison uses a different effective proposal budget. Together, these records describe activation authority; proposal quality and downstream utility are evaluated separately.

What happens when construction does not activate. Failed candidates receive sanitized diagnostics and may be revised within the registered attempt budget. If no candidate passes, CASES returns ConstructionUnresolved and exposes no uncertified substrate to its downstream explorer. A host may continue through a separately specified native workflow, but such a fallback is outside the certified CASES path evaluated here. The four activations among 121 auditable construction records measure candidate level yield under this policy; they are not a task level probability that an agent can complete a scientific campaign.

What the current records can measure. The open and menu arms contain 68 and 60 proposal records, respectively. The stored episode table also reports seven execution errors and 428 attempt traces across parsing, schema validation, and Audit. The best-of-three comparison changes the effective proposal budget, but its reported wins, losses, and ties do not give activation counts at each attempt budget. Consequently, these summaries do not support an attempt-budget-versus-activation curve or an estimate of time to first activation.

Authorization precision and false rejection. The corruption challenge tests whether known invalid proposals pass $(0/12)$; it does not label rejected proposals for scientific usefulness. Nor do the supplied summaries include blinded human adjudication of rejected candidates. Thus the false-rejection rate, constructor recall, and absolute precision of authorization cannot be estimated from these records. CASES currently favors a strict authorization boundary, as reflected in the low activation yield; quantifying the precision–recall frontier requires candidate level rejection reasons, independent expert labels, and downstream tests of candidates judged useful.

C.5 Activated Revisions Use Fresh Certification

The frozen integrity package contains 428 stored attempt traces across parsing, schema validation, and Audit execution. The corresponding stored episode table contains 128 records, including seven recorded execution errors.

Five candidates are marked valid for activation. Every activated record is associated with a certification suite marked fresh:

$5/5$ fresh at activation.

No activation is recorded against a suite marked stale or explicitly reused. Inspection of the frozen artifacts additionally finds no logged diagnostic-sanitization violation and no raw distinction-probe text in proposer-visible revision feedback.

The recorded procedure therefore follows

Audit diagnosis $\rightarrow$ sanitized revision signal $\rightarrow$ revised proposal $\rightarrow$ fresh certification.

The recorded revisions therefore meet the procedure's fresh-certification requirement.

C.6 The Activated Substrate Remains Fixed During Exploration

The implementation records four classes of prohibited mutation: `space_type_change`, `new_sort_insertion`,`gamma_mutation`, and`interp_mutation`.

Across the formal runs in the frozen execution package, no prohibited mutation was recorded. This establishes $S_{\tau,t} = S_\tau$ for the logged episodes. It is distinct from Sec. 3.4, which asks whether such a fixed substrate remains useful as evidence changes.

C.7 Required Distinctions Are Nontrivial Activation Obligations

The frozen artifacts do not contain the certified z-level realization pairs and registered signature pseudometric needed to estimate the formal margin $\gamma_U - 2\varepsilon_U$.

The empirical check here is required distinction coverage at activation.

Among 121 audited open- and menu-construction candidate records, 251 required distinction obligations are compiled. Of these, $193/251 = 76.9\%$ satisfy the registered coverage floor under both the point and conservative scores.

Among the four candidates activated by the full Construct procedure, all nine registered required distinction obligations satisfy the same criterion: $9/9 = 100\%$.

Including the independent Audit lane gives $12/12$ covered obligations across five valid-for-activation candidates. Across 171 auditable candidate records in the larger package, the corresponding coverage is $265/379 = 69.9\%$.

Activated candidates in the frozen runs satisfy their registered required distinction coverage floor. The formal noncollapse condition concerns a separate, unestimated margin.

C.8 What These Checks Establish

The frozen artifacts show the operational chain behind CASES: registered checks filter proposals, activation exposes an authorized scope, revision uses fresh certification, and the substrate remains fixed during logged exploration. Required distinctions meet their coverage floor at activation.

These checks concern the observed procedure; Appendix D measures scientific outcomes, and the noncollapse margin remains a formal condition.

Together, the formal substrate property has an explicit operational mechanism and auditable execution evidence.

D Complementary Empirical Analyses

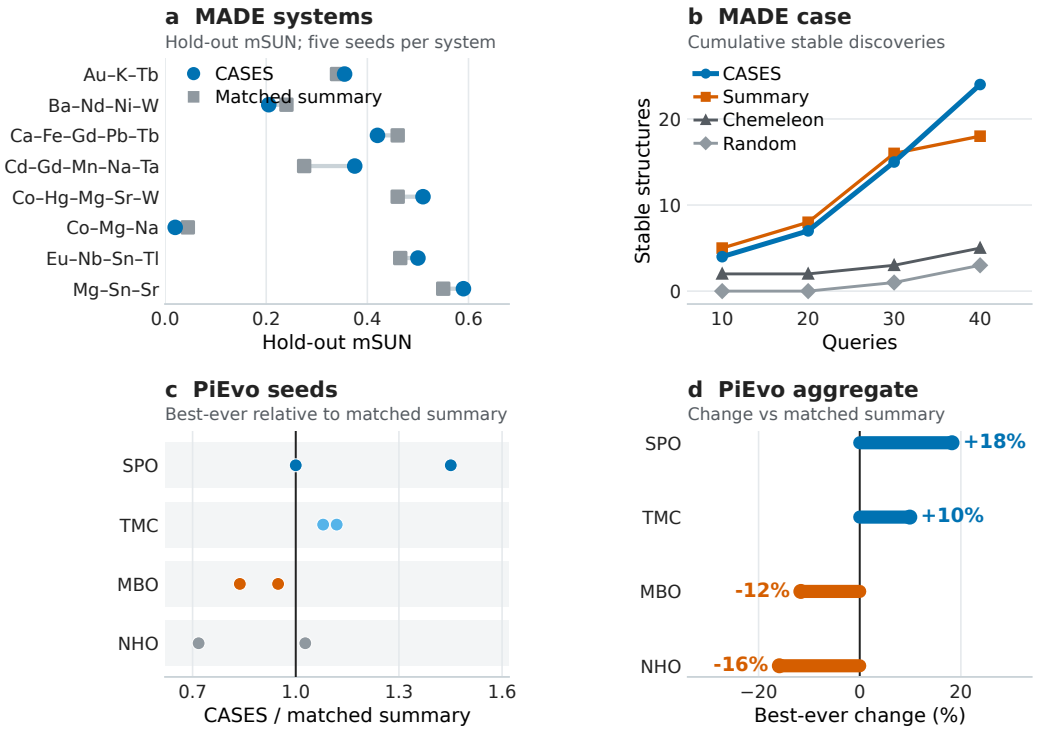


Figure 3: **Downstream use of an activated substrate.** (a) MADE hold-out mSUN across eight material systems (circles: CASES; squares: Length-matched summary; five seeds each). (b) Cumulative stable discoveries in a representative MADE case (illustrative). (c) PiEvo seed level best ever scores relative to the matched summary on the same task and seed (two paired seeds per task; values above 1 favor CASES). (d) Aggregate change in best ever score versus the matched summary.

The main text follows the substrate from construction to search and downstream use. This appendix looks beneath the aggregate outcomes: which controls reconstruct useful organization, when solution regions emerge during search, how the substrate changes access to extreme solutions, and how its readout behaves across hosts and consumers.

The analyses follow construction, scientific access, and downstream reuse. Representative trajectories illustrate the behaviors measured across instances and seeds.

D.1 Complementary Analysis of Explicit Substrate Construction

D.1.1 Structural controls, adaptive reasoning, and difficult tasks

Table 15: **ResearchBench construction and reasoning controls.** hardA is the outcome-independent split ($n = 160$); hardB is a baseline-defined post hoc slice.

Method	Overall	hardA	hardB
Fixed schema (control)	3.040	3.044	1.394
Editable structure (control)	2.991	3.084	1.221
Without Audit revision (ablation)	2.957	3.059	1.487
Direct refinement (control)	3.197	3.303	1.748
VIRSCI	2.831	2.838	1.547
ResearchAgent	3.229	3.331	2.013
CASES	**3.294**	**3.406**	**2.145**

Fixed, editable, and no-revision representations remain substantially below CASES, whereas adaptive reasoning closes much of the overall gap. The hierarchy supports the interpretation that explicit upstream construction and online reconstruction are partially substitutable rather than interchangeable.

The registered output-matched Direct Refinement control maps each role to the CASES output allowance (2,048/1,024/2,048/2,048 tokens; 7,168 total) while retaining the Direct prompts. Its hardA score is 3.0562, compared with 3.4062 for CASES:

$$\Delta_{\mathrm{hardA}} = +0.350, \qquad p = .002.$$

The hardA difference persists when Direct Refinement receives the same total output allowance as CASES.

D.1.2 A localized contribution from explicit causal encoding

Removing the explicit causal encoding clause from the ResearchBench Construct prompt while retaining the remainder of the workflow changes the reported hardA arm means from 3.406 to 3.263. The paired estimate is

$$\Delta = -0.147, \qquad p = .045$$

on the matched subset ($n = 156$); this paired estimate is distinct from the difference between the reported arm means. The remaining tested main effects and interactions are not individually significant. This prompt intervention isolates a contribution from the causal encoding clause on this subset.

D.1.3 Full no-LLM paired effects

Table 16: No-LLM HypoSpace paired effects.

Domain	Comparator	Δ RR-AUC	p_{perm}
Causal	History	+.0125	.08
Causal	Hypothesis Search	+.0225	$\approx 5 \times 10^{-6}$
Causal	Diverse Scientific Hypothesis Search	+.0370	$\approx 5 \times 10^{-6}$
Boolean	History	+.0100	.033
Boolean	Hypothesis Search	+.0185	$\approx 5 \times 10^{-6}$
Boolean	Diverse Scientific Hypothesis Search	+.0671	$\approx 5 \times 10^{-6}$
Voxel	History	+.0588	$\approx 5 \times 10^{-6}$
Voxel	Hypothesis Search	+.0003	3×10^{-5}

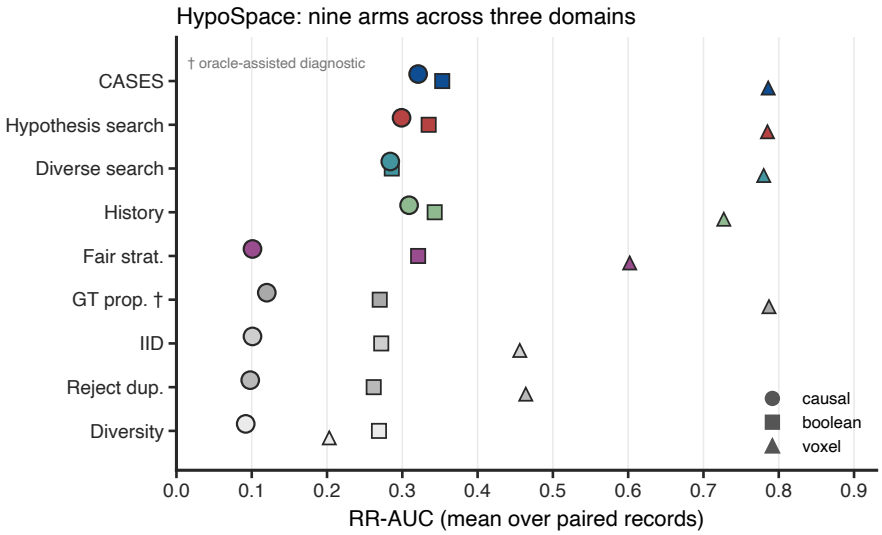


Figure 4: **Full no-LLM HypoSpace comparison.** Recovery-ranked AUC across nine registered arms in the causal, Boolean, and Voxel domains. Oracle-assisted diagnostics are marked in the panel and kept separate from deployable methods.

Domain	Comparator	Δ RR-AUC	p_{perm}
Voxel	Diverse Scientific Hypothesis Search	+.0055	$\approx 5 \times 10^{-6}$

The causal comparison with History gives $p = .08$. Hypothesis Search is practically tied with CASES in voxel: the paired effect is $+.0003$, with $444/645$ exact ties. Every arm eventually reaches

$$V/U/R = 0.227/1.000/0.920,$$

so RR-AUC distinguishes only the order of recovery.

D.1.4 Representative ResearchBench cases

Environmental Science: retaining a low-prior causal control variable. In a task on P2-type sodium cathodes, CASES scores 3 while all six evaluated comparison workflows score 0. Comparison workflows converge on familiar intervention families such as doping, coating, or composition gradients. CASES instead promotes state of charge into an explicit control variable and preserves its relation to ion migration and stacking faults through final synthesis.

Cell Biology: retaining a low-prior mechanistic branch. In a hard task on microsatellite-stable colorectal cancer, CASES is the only evaluated workflow in the recorded case to retain an evidence-supported microbial–neoantigen mechanism through final synthesis. Competing workflows largely collapse onto the more familiar immunosuppression pathway.

These recorded cases illustrate scientific distinctions retained through final synthesis.

D.2 Complementary Analysis of Scientific Structure and Extreme-Solution Access

D.2.1 Buchwald–Hartwig: the region advantage emerges with accumulated evidence

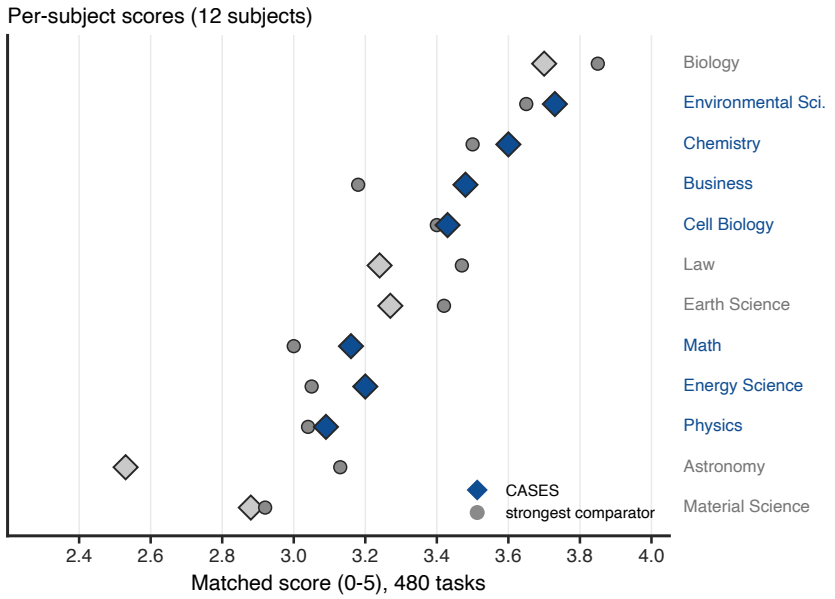


Figure 5: **ResearchBench effects across disciplines.** Matched subject scores compare CASES with the strongest comparator in each of 12 subjects. The subject pattern shows where the overall construction gain is concentrated.

Table 17: BH region recall over the exploration budget.

Budget	CASES	BO-GP	BALLET	BO-DKL
40	**.596**	.427	.545	.180
80	.464	.417	**.557**	.223
120	**.628**	.431	.508	.232
160	**.685**	.458	.509	.245
200	**.698**	.458	.504	.255
240	**.811**	.527	.484	.268

Across the full registered curve, CASES leads the trained comparison methods at $10/11$ checkpoints; the exception is budget 80, where BALLET is higher. The strongest region-recovery advantage therefore opens after evidence has accumulated rather than following from a uniformly superior initialization.

In representative CASES seed 41, the number of true high-yield reactions in successive 40-query batches rises from $0/40$ initially to $30/40$ in the final batch. The top-10 unqueried archive simultaneously improves from $0/10$ to $8/10$ true solutions. This trajectory illustrates behavior already visible in the aggregate budget curve.

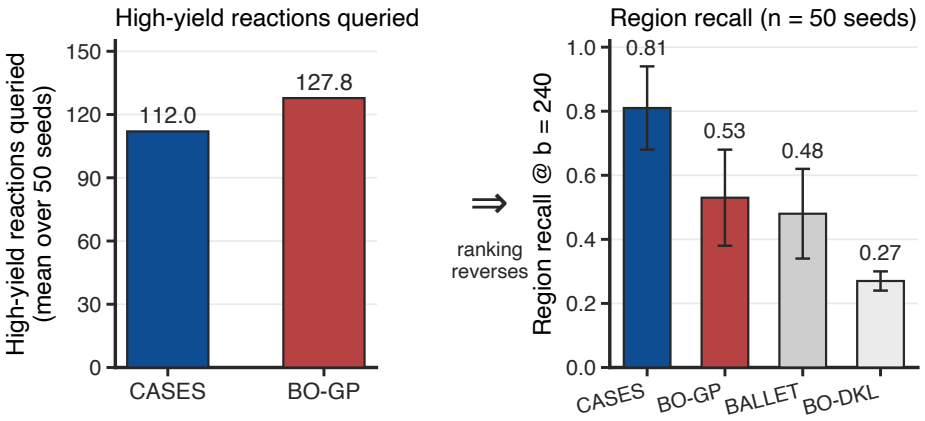


Figure 6: **Point hits and solution regions give different rankings.** At 240 Buchwald–Hartwig queries, BO-GP finds more individual high-yield reactions, while CASES recovers more independently defined high-yield regions.

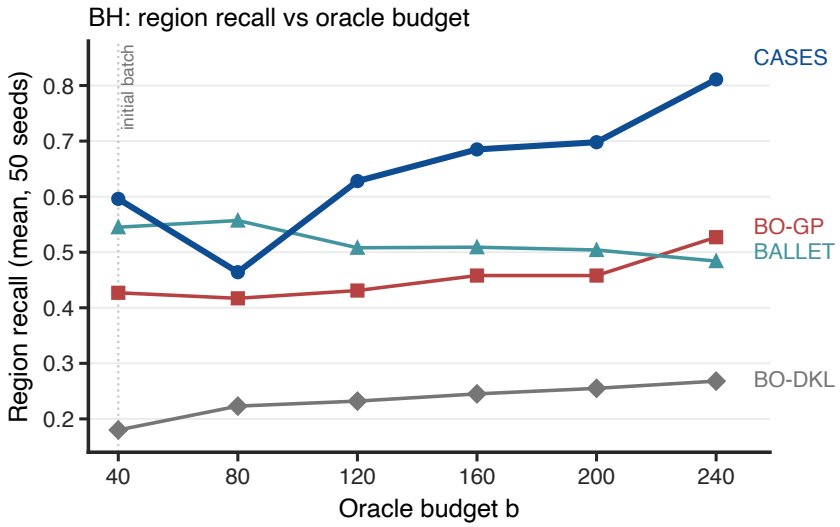


Figure 7: **Region recovery through the query budget.** The CASES advantage over BO-GP and BO-DKL grows after the initial 40-query batch; BALLET is higher at budget 80. Curves are means over 50 seeds.

D.2.2 The value of organization depends on target concentration

Table 18: BH coverage-AUC at $\theta = 1\%$. This is a separate 3,955-reaction sensitivity face.

Method	40	80	120	160	200	240
CASES graph	.776	.803	.850	**.896**	**.936**	**.916**
TPE	.756	.794	.808	.828	.848	.884
BALLET	.678	.652	.772	.761	.772	.824
Random	**.841**	**.863**	**.856**	.861	.856	.854
BO-GP	.219	.205	.219	.242	.219	.197
BO-DKL	.219	.197	.249	.268	.276	.272

Random broad coverage is initially competitive; CASES overtakes this floor from budget 160 onward. At broader $\theta = 10\%$, indiscriminate coverage can again become competitive. Constructed organization therefore has greatest leverage when the scientific objective requires prioritizing a concentrated, structured subset rather than broadly touching an admissible set.

D.2.3 GB1 TERMINAL SUPPORT REVEALS MORE THAN THE MEAN

Table 19: CASES terminal best fitness support over the 50-seed GB1 face.

Terminal fitness	CASES seeds	Interpretation
8.7620 (FWAA)	22	global optimum
8.0452 (FYAA)	1	second-highest variant
7.5569	7	above FWCA
7.5547 (FWCA)	2	same terminal value as GP-BO
7.3127	8	—
7.2823	3	—
6.8779	1	—
5.7720	6	—

GP-BO terminates at FWCA (7.5547) in all $50/50$ runs on this face and never reaches the global optimum. Level-set AL reaches FWAA in $4/50$ runs and FYAA in $1/50$; Random reaches FWAA in $1/50$. The terminal support therefore reveals extreme-tail access that is not summarized by a single mean.

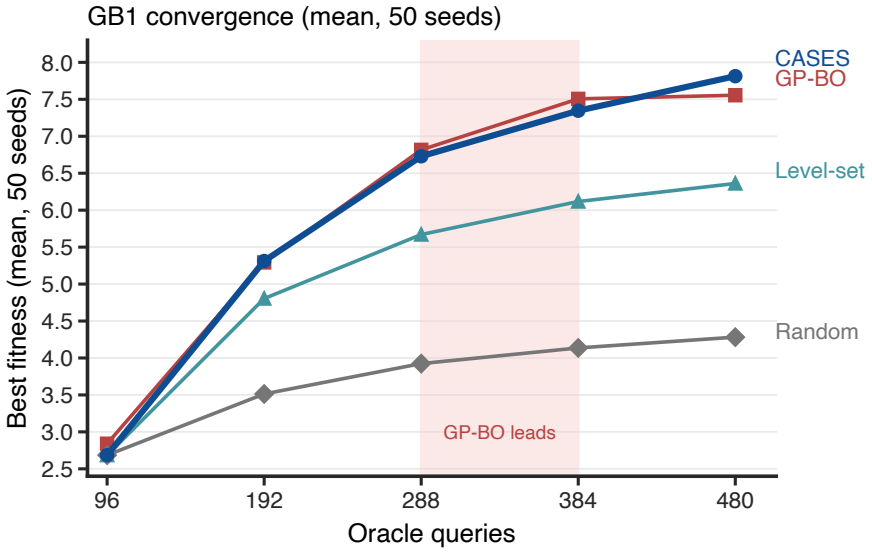


Figure 8: **GB1 best fitness trajectories.** Mean best fitness across 50 seeds at five registered query checkpoints. GP-BO leads at the intermediate checkpoints; CASES has the higher mean at 480 queries.

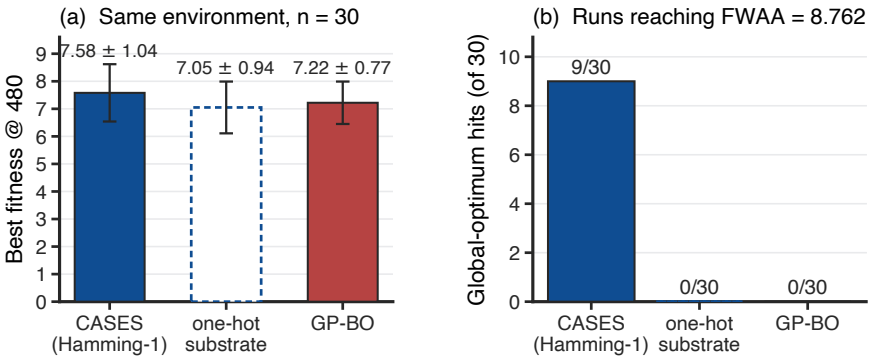


Figure 9: **GB1 fixed-selector intervention.** With the candidate library, environment, budget, and selector held fixed, Hamming-1 and one-hot organizations give different terminal best fitness and global optimum hit counts across 30 seeds.

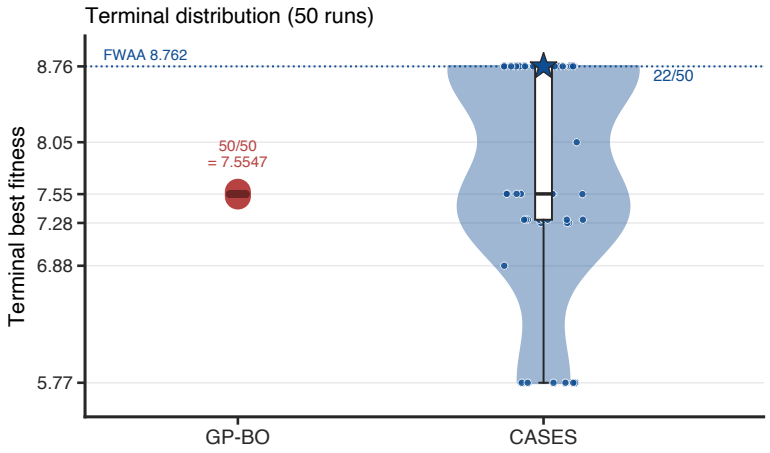


Figure 10: **GB1 terminal support across 50 seeds.** CASES reaches the global optimum FWAA in $22/50$ runs; GP-BO ends at FWCA (fitness 7.5547) in all $50/50$ runs on this protocol. The distribution exposes the extreme-solution outcome behind the mean best fitness comparison.

D.2.4 Same-environment intervention and equal-pool reachability

Condition / method	Best fitness@480	Global optimum hits
GP-BO	7.22± .77	0/30
CLADE	5.99± 1.01	1/30
Level-set AL	5.84± 1.00	1/30
ALDE	6.93± .97	0/30
Hill-climb	6.96± 1.42	8/30
Random	4.31± 1.22	1/30
One-hot substrate*(CASES selector)*	7.05± .94	0/30
Hamming-1 substrate *(CASES, ours)*	**7.58 ± 1.04**	**9/30**

The Hamming-1 versus one-hot contrast is the controlled intervention used in Sec. 3.3. No direct paired significance test is available for that contrast.

When all compared methods are instead restricted to a common 25k candidate pool, GP-BO recovers more gold solutions in all $20/20$ paired seeds, while best fitness differences are not significant. Thus solution-membership recovery $\neq$ access to extreme solutions.

D.3 Complementary Analysis of Frozen-Substrate Reuse

D.3.1 A representative MADE discovery trajectory

Table 21: Representative MADE cumulative stable discoveries. Illustrative, not inferential.

Method	q_{10}	q_{20}	q_{30}	q_{40}	Final stable
CASES (ours)	4	7	15	**24**	**24**
Length-matched summary	**5**	**8**	**16**	18	18
Chemeleon	2	2	3	5	5
Random	0	0	1	3	3

CASES and the Length-matched summary behave similarly through the first 30 queries and diverge late in the episode. The paired aggregate CASES–Length-matched summary gain is $+.0175$ mSUN (Wilcoxon $p = .0346$); this trajectory shows how later budget allocation can differ as evidence accumulates.

D.3.2 Aggregate performance hides material-system heterogeneity

Table 22: MADE hold-out mSUN by material system.

System	CASES	Length-matched summary	Chemeleon	Random
Au–K–Tb	.355	.340	**.535**	.335
Ba–Nd–Ni–W	.205	**.240**	.085	.050
Ca–Fe–Gd–Pb–Tb	.420	**.460**	.370	.310
Cd–Gd–Mn–Na–Ta	**.375**	.275	.080	.055
Co–Hg–Mg–Sr–W	**.510**	.460	.135	.095
Co–Mg–Na	.020	.045	**.050**	.010
Eu–Nb–Sn–Tl	**.500**	.465	.330	.265
Mg–Sn–Sr	.590	.550	**.700**	.460

CASES leads the Length-matched summary in five of eight systems; the summary leads in three, and Chemeleon is highest in two. Across systems, the paired aggregate CASES–Length-matched summary gain is $+.0175$ mSUN (Wilcoxon $p = .0346$).

D.3.3 Breadth and polymorph depth can trade off

For Eu–Nb–Sn–Tl, CASES discovers 16 stable structures across seven compositions, whereas the Length-matched summary discovers nine stable structures across eight compositions. The Length-matched summary therefore covers slightly more compositions, while CASES recovers substantially greater within-composition polymorph depth:

compositional breadth vs. within-composition depth.

This is an illustrative allocation pattern, not a general mechanism claim.

D.4 Generalization and Regimes of Usefulness

D.4.1 Seed-level external consumption in PiEvo

Table 23: Frozen PiEvo task–seed outcomes.

Task	Seed	Native	Character-matched summary	CASES readout
SPO	s1	30.69	65.84	**95.52**
SPO	s2	28.40	**96.81**	**96.81**
TMC	s1	453.81	457.89	**494.36**
TMC	s2	357.78	441.76	**494.36**
MBO	s0	3.400	**5.075**	4.144
MBO	s1	3.196	**4.540**	4.349
NHO	s1	1.704	**2.010**	1.491
NHO	s2	1.000	1.011	**1.039**

The selective task level pattern is visible without averaging across incompatible scales. With only two paired seeds per task, the observations remain descriptive.

D.4.2 Reintroducing an LLM creates a proposer bottleneck

Table 24: HypoSpace Recovery-AURC under different LLM proposal backbones.

Proposal backbone	Causal	Voxel	Boolean
GLM-5.3-Flash	**.506 ± .296**	**.507 ± .348**	**.062 ± .049**
GPT-OSS-20B	.460± .289	.180± .229	.042± .035
Qwen3.6-35B	.020± .039	.096± .219	.011± .018
Qwen3.8-27B	.108± .112	.200± .221	.015± .019

Removing the proposer does not remove the construction effect, but reintroducing an LLM creates an independent bottleneck. Recorded voxel duplicate shares are .454 for GLM-5.3-Flash, .228 for GPT-OSS-20B, .601 for Qwen3.6-35B, and .000 for Qwen3.8-27B. By construction, uniqueness is 1.0 in the no-LLM pool protocol. Duplicate proposal is therefore a proposer pathology rather than substrate collapse.

D.4.3 ResearchBench gains are heterogeneous across disciplines

Table 25: ResearchBench domain-level matched scores.

Domain	CASES	Direct Refinement	ResearchAgent	Highest arm
Astronomy	2.53	2.93	**3.13**	ResearchAgent
Biology	3.70	3.70	**3.85**	ResearchAgent
Business	**3.48**	3.18	3.10	CASES
Cell Biology	**3.43**	3.28	3.28	CASES
Chemistry	**3.60**	3.28	3.20	CASES
Earth Science	3.27	3.36	**3.42**	ResearchAgent
Energy Science	**3.20**	3.05	2.92	CASES
Environmental Science	**3.73**	3.38	3.65	CASES
Law	3.24	3.38	**3.47**	ResearchAgent
Material Science	2.88	**2.92**	2.67	Direct Refinement
Math	**3.16**	2.82	3.00	CASES
Physics	**3.09**	3.00	3.04	CASES

CASES is highest in seven of twelve disciplines, ResearchAgent in four, and Direct Refinement marginally in Material Science. A post hoc qualitative reading suggests larger CASES gains often occur in tasks with explicit mechanistic, conditional, or mediator structure. This taxonomy was not preregistered and is therefore explanatory only.

D.5 How Objective and Setting Shape the Outcome

Table 26: Outcome patterns across objectives and settings.

Setting	Observed contrast	What the contrast distinguishes
ResearchBench	Direct Refinement and ResearchAgent close much of the average gap; ResearchAgent $p = .06$	How online reconstruction substitutes for explicit construction
HypoSpace	Voxel Hypothesis Search differs by $+.0003$ RR-AUC with $444/645$ exact ties	Recovery order in a nearly saturated finite pool
BH	BALLET leads at budget 80; random coverage is competitive for broader target sets	Budget and target concentration
GB1	Primary GP-BO mean comparison $p = .069$; equal-pool GP-BO recovers more gold solutions in $20/20$ seeds	Membership recovery versus access to the global optimum
MADE	CASES aggregate advantage over Length-matched summary $p = .0346$; several systems reverse the ordering	Variation among material systems
PiEvo	Summary exceeds CASES on MBO and NHO; two paired seeds/task	Dependence on the consumer objective

The contrasts point to three conditions for substrate utility: the task contains structure that matters within the query budget, construction exposes that structure faithfully, and the downstream explorer can use it. The same data also separate recovery of many satisfactory candidates from access to an extreme optimum.

D.6 Summary of Complementary Findings

ResearchBench and no-LLM HypoSpace locate the contribution of construction under shared consumers and fixed candidate pools. Buchwald–Hartwig and GB1 show why the scientific target matters: recovering solution-bearing regions, accessing a global optimum, and retrieving many gold-set members are different outcomes. MADE and PiEvo then follow the activated substrate into downstream use, where the value of its readout varies by material system and consumer objective.

Together, these analyses locate substrate value in the alignment among task structure, faithful organization, and the downstream consumer's ability to use it.

Table 27: **HypoSpace without an LLM proposer.** Causal, Boolean, and Voxel each report official validity (V), uniqueness (U), recovery (R), and recovery-ranked AUC (AUC). Values are means over paired episodes with fixed candidate pools and query budgets; paired tests appear in Appendix D and protocols in Appendix F. Bold marks the highest informative mean among methods shown.

	Causal				Boolean				Voxel			
Method	V	U	R	AUC	V	U	R	AUC	V	U	R	AUC
History	.170	1.00	.558	.309	**.152**	1.00	**.561**	.343	.227	1.00	.920	.727
Hypothesis Search ICLR'24	.166	1.00	.549	.299	.150	1.00	.557	.335	.227	1.00	.920	.785
Diverse search arXiv'26	.176	1.00	.573	.284	.135	1.00	.509	.286	.227	1.00	.920	.780
Fair stratification	.064	1.00	.210	.101	.147	1.00	.544	.321	.227	1.00	.920	.602
IID	.064	1.00	.201	.101	.142	1.00	.528	.272	.227	1.00	.920	.456
CASES	**.184**	1.00	**.586**	**.321**	**.152**	1.00	.559	**.353**	.227	1.00	.920	**.786**

Table 28: **PiEvo transfer.** Best ever scores; two paired seeds per task.

Task	Summary	Native	CASES
SPO	81.33	29.55	**96.17**
TMC	449.82	405.80	**494.36**
MBO	**4.81**	3.30	4.25
NHO	**1.51**	1.35	1.27

E Future Directions

CASES constructs and authorizes a substrate before an exploration episode, then holds it fixed while evidence accumulates. This protocol makes the effect of construction measurable. It also opens a next set of questions: when should an explorer keep its substrate, when should it revise one, and when should several authorized organizations coexist?

E.1 Closing the Construct–Explore–Lift–Evolve Loop

Construction ends at activation in the evaluated protocol. A future loop could use discoveries from one episode to organize the next:

$$\mathsf{Construct} \to \mathsf{Explore} \to \mathsf{Lift} \to \mathsf{Evolve}.$$

Here, exploration accumulates evidence on an activated substrate, while Lift interprets recovered structure back into scientific hypotheses, relations, or solution-bearing regions. Evolve would then ask whether another exploration episode should continue on the current substrate or whether the organization itself should be reconsidered.

Evidence can revise beliefs within an existing substrate or reveal that the substrate itself needs revision. An observation may change which alternatives look promising; repeated observations may expose a missing distinction, a misleading neighborhood, or an inappropriate operation. A future explorer would need to decide which kind of change the evidence calls for.

Closing this loop also creates an attribution problem. If scientific performance improves after both new evidence and substrate reconstruction, the improvement may arise from the observations, the exploration policy, the new organization, or interactions among them. Future evaluation protocols will need to separate these effects rather than allowing evidence accumulation and representational change to co-evolve invisibly.

E.2 Evidence-Driven Substrate Evolution and Scientific Versioning

Allowing substrates to evolve makes representational change an explicit scientific operation. New evidence could motivate adding a previously absent distinction, revising a relation, splitting a heterogeneous region, replacing an inappropriate neighborhood structure, introducing a newly justified operation, or changing the correspondence between scientific meaning and computational realization.

An especially interesting possibility is to treat **persistent representational failure as evidence about the scientific organization itself**. Repeated local prediction failures may suggest an omitted variable; disconnected high-value solutions may expose a missing relation; abrupt transitions may reveal that a smooth geometry is inappropriate; and systematic failures of evidence propagation may indicate that the current notion of neighborhood is scientifically misleading. In such cases, the appropriate response may not be another hypothesis within the same substrate, but a targeted revision of the substrate.

Such evolution requires continuity across substrate versions,

$$S_\tau^{(0)} \to S_\tau^{(1)} \to \cdots \to S_\tau^{(k)}.$$

Discoveries made under one version should not silently lose their meaning under the next. Future systems will therefore need explicit cross-version correspondences specifying which measurements, distinctions, realization relations, and previously authorized claims remain valid after a change. This raises problems of selective re-auditing, rollback, provenance, and preservation of scientific results across changing computational organizations.

Substrate evolution would change computational organization while recording the scientific commitments affected by each revision.

E.3 Multiple and Shared Substrates for Scientific Agents

The evaluated protocol activates one substrate per episode. A scientific problem may support several authorized organizations with different useful structure: mechanistic relations, local similarity,

causal interventions, or executable transformations. A future system could maintain a **portfolio of authorized substrates** and allocate exploration across them.

Disagreement between such substrates may itself be informative. Two organizations over the same candidate universe can agree about observed evidence yet recommend different experiments because they encode different notions of scientific relatedness. Their disagreement could therefore become an acquisition signal: experiments may be valuable not only because they reduce uncertainty within one model, but because they distinguish between competing organizations of the scientific problem.

The same idea naturally extends to multi-agent AI scientists. Hypothesis generators, experiment planners, simulators, critics, and optimization agents need not share an internal reasoning architecture, but they could consume different views of a common scientific substrate. Alternatively, different agents could maintain partially overlapping substrates and negotiate correspondences or revisions between them.

This raises new questions of authority and coordination. Which agent may propose a substrate revision? Can several views of the same scientific content be authorized simultaneously? How should contradictory evidence or incompatible organizations be reconciled? More generally, a substrate could become a **shared scientific interface** through which agents coordinate explicit distinctions, evidence, executable operations, and unexplored regions rather than communicating only through free-form text or a flat experiment history.

E.4 Meta-Construction: When and How Should a Substrate Be Built?

The controls reveal different returns to explicit construction. Strong adaptive reasoning can reconstruct useful organization online; a saturated candidate pool can leave little room for improved recovery order; and an evolving summary can sometimes approach a structured readout. This motivates a further decision: **when is construction worth performing?**

A meta-constructor could choose among several regimes: inherit an existing organization, refine it locally, construct a new substrate, maintain several competing substrates, or rely on a sufficiently capable downstream explorer to recover the necessary structure online. Such decisions might depend on properties such as scientific heterogeneity, locality, sparsity of useful regions, reliability of candidate neighborhoods, experimental cost, or the ability of the consumer to reconstruct latent structure during inference.

Experience across previous tasks could also provide reusable organizational primitives: recurring scientific distinctions, neighborhood operators, executable transformations, realization patterns, or audit obligations. The challenge is to reuse such structure without collapsing substrate construction into selection from another fixed menu. A useful meta-constructor must preserve the possibility that a new task requires genuinely new scientific distinctions or computational organization.

An increasingly autonomous scientific system could make this decision explicitly by asking both

> *What should I explore next?*

and

> *Is the organization through which I am exploring still the right one?*

F Models, Prompts, and Experimental Configurations

This appendix records the task environments, models and hosts, prompts and interfaces, baseline adaptations, budgets, pairing rules, metrics, and statistical conventions used in the frozen evaluation. Appendix D analyzes the scientific outcomes.

F.1 Task and Evaluation Configurations

Table 29: Overview of the frozen experimental configurations.

Setting	Paired unit / scale	Main budget	Primary metric	Experimental role
ResearchBench	480 tasks	$\leq$ 4 LLM calls; 48k input / 8,192 output envelope	matched score	construction / reasoning controls
HypoSpace no-LLM	instance$\times$ seed; 5 seeds	100 queries; voxel 80	RR-AUC	remove LLM proposer
HypoSpace LLM	24 instances$\times$ 10 seeds / model-domain group	checkpoints 2/4/6/8	Recovery-AURC	proposer-backbone sensitivity
Buchwald–Hartwig	50 paired initialization seeds	240 oracle calls	region recall	reaction-region recovery
GB1	seed	480 oracle calls	best fitness@480	extreme-solution access
MADE hold-out	8 systems$\times$ 5 seeds	40 queries	mSUN	frozen-substrate sequential reuse
MADE polymorph	6 systems$\times$ 5 seeds	50 queries	mSUN	polymorph recovery
PiEvo	task$\times$ seed	20 turns	task specific best ever	external consumption

F.1.1 RESEARCHBENCH

ResearchBench contains 480 tasks spanning twelve scientific disciplines. The generator receives the research question, background survey, and experiment details; external retrieval, tools, and gold inspirations are disabled. Primary generation uses Qwen3.6-35B at temperature 0; the frozen scorer is Qwen3.8-27B at temperature 0.

The outcome-independent hardA subset is the top third of tasks under

$$\text{complexity}(t) = |q_t| + |b_t| + |e_t|,$$

where q_t, b_t, e_t are the question, background survey, and experiment-details text. This yields $n = 160$. The baseline-defined hardB slice is used only in Appendix D as complementary post hoc analysis.

F.1.2 HYPOSPACE

Protocol	Candidate source	Frozen scale	Budget	Primary process metric
no-LLM	deduplicated frozen pool	causal 480; Boolean 610; voxel 645 paired records	100; voxel 80	RR-AUC
LLM	proposer-generated candidates	24 instances$\times$ 10 seeds / model-domain group	checkpoints 2/4/6/8	Recovery-AURC

The task families are causal graphs, Boolean mechanisms, and 3D voxel structures. In the no-LLM protocol, all arms receive the same candidate pool and oracle budget. In the LLM protocol, the language model serves only as proposer; admission, consistency checks, validity-feedback propagation, and the frozen controller are shared within each model group.

F.1.3 BUCHWALD–HARTWIG

The task uses the Doyle–Dreher high-throughput Buchwald–Hartwig amination landscape introduced in the main text. Each candidate is an aryl-halide $\times$ ligand $\times$ base $\times$ additive tuple and the oracle is measured reaction yield. Runs share 40 random initial observations and a total budget of 240 oracle calls over 50 paired initialization seeds.

The headline endpoint campaign uses the archived 4,312-well face. Separate sensitivity and case analyses use the registered 3,955-reaction all-four-component face after removing 177 wells associated with the beyond-paper additive 5-Phenyl-1,2,4-oxadiazole. Quantitative results from these faces are never pooled.

F.1.4 GB1

The GB1 universe contains 149,361 sequence variants. Exploration uses five batches of 96 queries for a total budget of 480. Table 5 reports the common 30-seed primary endpoint face; Appendix D additionally uses the registered 50-seed distribution face for CASES, GP-BO, Level-set AL, and Random. The equal-pool control restricts compared methods to a common 25k candidate pool over 20 paired seeds.

The CASES Hamming-1 organization is constructed from sequence identity before fitness labels are introduced.

F.1.5 MADE

MADE supplies the benchmark environment; the stability oracle uses ORB v3 (Rhodes et al., 2025).

Task semantic	Initial condition	Systems	Seeds/system	Budget	Objective
Hold-out	no stable-phase information	8	5	40	discover new ORB-stable structures
Polymorph recovery	stable compositions initially known	6	5	50	discover structurally novel stable polymorphs

The hold-out systems are Au–K–Tb, Ba–Nd–Ni–W, Ca–Fe–Gd–Pb–Tb, Cd–Gd–Mn–Na–Ta, Co–Hg–Mg–Sr–W, Co–Mg–Na, Eu–Nb–Sn–Tl, and Mg–Sn–Sr. Stability uses ORB v3 with

$e_{\text{above hull}} \leq 0.1\ \text{eV/atom}.$

F.1.6 PiEvo

We use PiEvo with its official Hypothesis / Experiment / Principle multi-agent architecture on SPO, TMC, ChEMBL35-MBO, and NHO with GLM-5.3-Flash and a maximum of 20 turns. The frozen paper face contains **two paired seeds per task** and is treated descriptively. No cross-task averaging is performed because the official task objectives have different scales.

F.2 Baselines and Experimental Controls

F.2.1 ResearchBench

Method	Provenance / role	Intermediate workflow
Fixed Schema	internal control	predefined entities / relations / mechanism / moderators / outcome / novelty schema
Editable Structured	internal control	editable claims / evidence / open-questions / proposed-edits state
CASES w/o Audit-guided revision	architectural ablation	same Construct and Normalize; no Audit-guided revision
Direct Refinement	internal reasoning control	Draft→ Critique → Revise → shared Compose
ResearchAgent-adapted (Baek et al., 2025)	NAACL 2025	literature-grounded proposal / review / revision adapted to the task packet
VIRSCI-adapted (Su et al., 2025)	ACL 2025	three-agent proposal / cross-evaluation / synthesis adapted to the task packet and shared consumer
CASES	ours	Construct→ Normalize → Audit/Revise → shared Compose

"Adapted" indicates that infrastructure specific to the original systems, such as external retrieval environments, is not reproduced when it falls outside the registered ResearchBench task interface.

F.2.2 HypoSpace

Method	Provenance / role	Search rule
History	protocol control	prioritize candidates using previously successful discoveries
Diversity	protocol control	max–min feature spreading
Fair stratification	protocol control	sample across complexity strata
Reject duplicates	protocol control	rejection sampling with duplicate filtering
IID	protocol floor	uniform candidate sampling
Hypothesis Search	ICLR 2024	explicit hypothesis-space search
Diverse Scientific Hypothesis Search (Wang et al., 2026)	ICML 2026; operationalized	farthest-point feature coverage
GT Proposal	oracle diagnostic	ground-truth-proportional sampling; excluded from competitive ranking
CASES	ours	validity-feedback propagation over constructed candidate organization

The Diverse Search condition is an operationalization because official code was unavailable.

F.2.3 BUCHWALD–HARTWIG

Method	Configuration
BO-GP	BoTorch qLogEI (Balandat et al., 2020); one-hot + RBF GP; refit each batch
BO-DKL	Morgan radius-2, 2048-d representation + deep-kernel GP (Singh & Hernández-Lobato, 2024)
BALLET	superlevel-set boundary sampling + region expansion
CASES	audited one-factor substitution graph + Laplacian evidence propagation

F.2.4 GB1

Method	Configuration / proposal rule
GP-BO	RBF-GP on 80-d one-hot; EI; registered 25k proposal-pool cap
CLADE (Qiu et al., 2021)	hierarchical clustering + proportional allocation + within-cluster GP-UCB
Level-set AL	BALLET-style one-hot boundary search near $P(\text{sol}) = 0.5$
ALDE	parent mutation proposals + GP-EI
Hill-climb	unqueried Hamming-1 neighbors of the current best
Random	uniform without replacement
CASES	outcome-blind Hamming-1 organization + Laplacian mean/std propagation

F.2.5 MADE

Method	State / proposal information
Random	official uniform legal-candidate baseline
Chemeleon (Park et al., 2025)	official MADE configuration using the text-conditioned crystal prior
Observed-success	internal control exposing only previously successful entries
Length-matched summary	neutral evolving statistical state of comparable length; no substrate field ranking
Native host	official-style MADE LLM host without CASES readout

Method	State / proposal information
CASES	empirical-field + coverage readout derived from the fixed activated substrate

F.2.6 PiEvo

Condition	Injected information
Native	none
Character-matched summary	generic evidence summary matched to the CASES guidance length
CASES	compact substrate-derived readout containing tested candidates, prioritized untested candidates, the best observed region, and under-covered regions

The only code-path intervention is a guidance hook in the Hypothesis Agent. Native PiEvo prompts, Experiment and Principle agents, official surrogate models, and scoring code remain unchanged.

F.3 Models and Scientific Hosts

Table 38: Model and host configuration.

Setting	Backbone / scorer	Temperature	Host / consumer	Frozen envelope
ResearchBench	Qwen3.6-35B / Qwen3.8-27B scorer (Qwen Team, 2026)	0 / 0	shared final composer	$\leq$ 4 calls; 48k input / 8,192 output
ResearchBench validation	GLM-5.3-Flash (GLM-5 Team, 2026) / same scorer	frozen	same composer	same registered envelope
HypoSpace no-LLM	—	—	frozen query controller	0 LLM calls
HypoSpace LLM	GLM-5.3-Flash, GPT-OSS-20B (OpenAI, 2025), Qwen3.6-35B, Qwen3.8-27B	frozen per group	frozen admission/controller	shared within model group
MADE hold-out	GLM-5.3-Flash	frozen	matched discovery host	40-query campaign
MADE polymorph	deepseek-v4-flash (DeepSeek-AI, 2026)	.7	DSPy / ReAct-style host (Khattab et al., 2024; Yao et al., 2023)	50-query campaign
PiEvo	GLM-5.3-Flash	frozen	official PiEvo multi-agent host	max 20 turns

API-served model aliases are recorded with their resolved evaluation endpoint and date in the frozen run artifacts; the paper reports the model identity used by the registered campaign rather than treating provider aliases as separate methods.

F.4 Prompt and Interface Configurations

F.4.1 ResearchBench CASES workflow

All ResearchBench arms terminate in the same final composer system prompt and user template; the scorer sees only the final hypothesis. The intervention is therefore confined to the intermediate organization delivered to the common consumer.

The CASES workflow uses four functional stages:

1. **Construct:** produce evidence-bound claims, source compilation, active commitments, and computational grounding.
2. **Normalize:** deduplicate and canonicalize claims and terms, retaining only well-formed commitments.

3. **Audit/Revise:** identify the highest-impact information loss and emit one surgical patch; code applies the patch deterministically.
4. **Compose:** synthesize the final hypothesis under the shared consumer and the activated synthesis contract.

The frozen Construct prompt requires 4–8 evidence-bound claims spanning phenomena/causal factors, methods or experimental designs, measures or datasets, and task relevant distinctions. When the evidence supports a causal or conditional link, that link must be represented explicitly rather than leaving the factors as unrelated claims.

The synthesis contract requires the final hypothesis to preserve relational structure rather than flatten it into a factor list. In particular, when the substrate contains a directed or conditional commitment between two factors, the final hypothesis must verbalize that relation.

F.4.2 RESEARCHBENCH EXPANDED-OUTPUT DIRECT CONTROL

Stage	Original Direct cap	Expanded Direct cap
Draft	1024	2048
Critique	512	1024
Revise	1024	2048
Compose	1024	2048

The Direct prompts are unchanged. The control matches the registered role level output envelope on hardA ($n = 160$); other computation budgets remain outside this comparison.

F.4.3 HYPOSPACE CONTROLLER

In the LLM protocol the language model is used only as a proposer. The frozen controller uses

$(\lambda_s, \lambda_u, \lambda_c, \lambda_o) = (0.5, 0.2, 0.15, 0.15)$.

The no-LLM protocol bypasses proposal entirely and retains candidate organization plus validity-feedback propagation.

F.4.4 MADE READOUT AND LENGTH-MATCHED CONTROL

Within each MADE semantic, matched LLM arms share host, tools, scorer, validity filters, oracle, and seed. CASES exposes an evidence-conditioned empirical-field / coverage readout while the activated substrate organization remains fixed. The length-matched control injects a neutral evolving summary of comparable length without the substrate field ranking. The comparison holds the text budget fixed while allowing the guidance content to differ.

F.4.5 PIEVO GUIDANCE INTERFACE

The CASES readout is capped at 900 characters and follows the frozen information structure:

> recovered / tested summary; already tested candidates; highest-priority untested candidates; best tested region; under-covered coarse regions.

The corresponding generic summary is character matched and uses the same Hypothesis-Agent hook. No CASES information is injected into the Experiment or Principle agents. This isolates whether an external Hypothesis Agent can operationally consume a compact substrate-derived readout without adopting the CASES exploration policy.

F.5 BUDGETS, PAIRING, AND FAILURE HANDLING

Table 40: Pairing and shared quantities.

Setting	Pairing unit	Shared within pair	Failure treatment
ResearchBench	task	payload, scorer, final composer, registered model envelope	failures retained in registered accounting
HypoSpace no-LLM	instance× seed	candidate pool, oracle budget	deterministic campaign
HypoSpace LLM	model× domain × instance × seed	protocol, admission, controller	failed cells remain under frozen rules
BH	initialization seed	initialization, oracle, readout budget	no unfavorable-run removal
GB1	seed	compatible universe/face, oracle, checkpoint schedule	face compatibility enforced
MADE	system× seed	host infrastructure, filters, tools, oracle, budget	matched retry/repair behavior
PiEvo	task× seed	official stack, config, oracle, turn budget	final paper uses two paired seeds/task; no imputation

The frozen failure rules in Table 40 govern run inclusion. Comparisons that require a common candidate universe or solution-set definition use the compatible experimental face.

F.6 Metrics and Statistical Conventions

Table 41: Primary and complementary evaluation quantities.

Setting	Primary metric	Complementary / diagnostic quantities	Pairing	Statistical convention
ResearchBench	matched score	hardA, hardB, factor ablations	task	two-sided paired sign test; expanded-output hardA also permutation
HypoSpace no-LLM	RR-AUC	official$V/U/R$	episode	paired permutation
HypoSpace LLM	Recovery-AURC	validity / uniqueness diagnostics	model-domain-instance-seed	paired within frozen face
BH	region recall	high-yield hits, budget curves, concentration sensitivity	initialization seed	paired permutation
GB1	best fitness@480	GO hits, terminal support, found recall on compatible faces	seed	200k paired permutations for registered paired comparisons
MADE	mSUN	AUDC, stable counts, composition/polymorph allocation	system-seed	paired Wilcoxon; cluster bootstrap for polymorph analysis
PiEvo	task specific best ever	per-seed outcomes	task-seed	descriptive, two paired seeds/task

For HypoSpace,

$$V = \frac{\#\text{valid queries}}{\#\text{queries}}, \qquad U = \frac{\#\text{unique parsed hypotheses}}{\#\text{queries}}, \qquad R = \frac{|\mathrm{GT} \cap \text{seen valid}|}{|\mathrm{GT}|}.$$

RR-AUC integrates cumulative recovery over normalized query budget. Recovery-AURC integrates LLM-protocol recovery over checkpoints $2, 4, 6, 8$.

For BH, letting $\mathcal{R}_{\mathrm{gold}}$ denote Morgan-kNN ($k = 5$) components containing unqueried high-yield solutions and $\mathcal{R}_{\mathrm{covered}}$ the components touched by top-ranked unqueried archive entries,

$$\text{region recall} = \frac{|\mathcal{R}_{\mathrm{gold}} \cap \mathcal{R}_{\mathrm{covered}}|}{|\mathcal{R}_{\mathrm{gold}}|}.$$

For MADE,

$$\mathrm{mSUN} = \frac{\#\text{new stable structures}}{\#\text{queries}}.$$

PiEvo retains each task's official `best_ever` metric. Scores are never averaged across the four tasks.

F.7 Interpreting the Configurations

Three configuration choices determine how the comparisons in Sec. 3 and Appendix D should be read.

ResearchAgent-adapted, VIRSCI-adapted, and the operationalized Diverse Search condition implement published ideas within the registered interface. The length-matched and character-matched controls hold the text budget fixed while allowing semantic content to vary. Each task chooses its computational form under the shared CASES construction and authorization process.

The recorded protocols identify the alternatives held fixed within each comparison and the properties that differ across tasks.