ClaimGarden: Update-Aware Claim-State Control for AI Scientist Workflows

Hafumi Nishi [1]

Abstract

AI scientist workflows can generate hypotheses, run analyses, revise plans, and draft plausible manuscripts, but their memory is often project- or paper-shaped. Across data-intensive sciences, changing databases, literature, simulations, agent analyses, and automated-laboratory measurements can make individual claims supported, overbroad, contradicted, obsolete, or silently promoted from predicted to experimental evidence. ClaimGarden shifts the unit of automation from projects and manuscripts to evolving *claim states*. It links harvested claims to versioned database, literature, computational, and laboratory evidence. Evidence updates trigger revalidation; verifiers recommend; deterministic policy commits; and manuscripts or follow-up tasks are gated by committed claim states. ClaimGarden does not certify truth: it records who or what judged a claim, from which evidence, under which policy, and why the state changed.

1. Claim-State Control as a Missing AI-Scientist Layer

We target a coupled AI scientist: an orchestrator covers ideation, planning, simulation or experiment design, execution, analysis, review, and manuscript drafting, while ClaimGarden governs the epistemic commitments produced by those stages. The motivating gap is not paper production but self-correction. A successful project may contain an overbroad conclusion; a failed project may leave a useful negative result; a database or robotic-lab update may reopen a claim from months earlier. **Fully automated science therefore requires autonomous claim formation, revision, and export control, not just autonomous experiment execution.**

Recent evaluations reinforce this distinction. LLM-based scientific agents can complete workflows while ignoring evidence, leaving hypotheses untested, and failing to revise after contradiction; in one large study, evidence was ignored in 68% of traces and refutation-driven belief revision occurred in only 26% (Ríos-García et al., 2026). ClaimGarden makes these failure modes operationally inspectable: hypotheses, results, database updates, automated-laboratory measurements, and manuscript sentences become auditable claim-state transitions rather than free text inside a trace.

Novelty and relation to prior work. Claim-centered and provenance-centered accounts of science are well established, including argumentative zoning, micropublications, nanopublications, PROV-O, ORKG, and SciFact (Teufel et al., 2009; Groth et al., 2010; Clark et al., 2014; Lebo et al., 2013; Jaradeh et al., 2022; Wadden et al., 2020). ClaimGarden's contribution is *policy-gated claim-state transition*: update-driven impact mapping, verifier recommendations, deterministic commits, and manuscript/next-action gates are coupled in one runtime loop. The central object is not a static claim record, but the state transition induced by new evidence, failed verification, database or laboratory updates, or attempted export. Independently, ResearchLoop is an evidence-gated, publication-facing control plane; ClaimGarden is complementary, focusing on update-aware revalidation of already-recorded claims as external databases, literature, computations, and laboratory evidence change (Xia & Wang, 2026). In short, ResearchLoop asks when a claim may enter a paper, while ClaimGarden asks how an existing claim should change state after its evidence changes.

2. Architecture and Tooling

ClaimGarden is the claim-state control plane of a semi-autonomous AI scientist. Connectors treat public databases, papers, local analysis artifacts, and automated-laboratory outputs as evidence streams rather than final authorities. A versioned evidence registry records source type, version, query, entity mapping, quality fields, artifact path, and hash. An impact mapper identifies affected claims after an evidence update. A verifier output is advice: it records the evidence bundle checked, the recommended status, uncertainty, and review flags. Only deterministic policy can write a committed `ClaimStatusChanged` event.

Claim-state loop. New evidence arrives → find affected claims → retrieve evidence → verifier recommends → policy commits → manuscript gate checks text → edit or create next task.

[1]Graduate School of Information Sciences, Tohoku University, Sendai, Japan. Correspondence to: Hafumi Nishi <hafumi.nishi@tohoku.ac.jp>.

Accepted by ICML 2026 AI for Science Workshop.

Table 1. Controlled structural-bioinformatics transition test. "After" is committed claim status; "Gate" is a separate manuscript action.

Claim-state test	Before	After	Gate
PDB-only coverage	supported	qualified	keep experimental-only scope
Motif conservation	unverified	qualified	state subset limit
Prediction as experiment	unverified	refuted	block as result claim

Importantly, plausible unsupported claims are assigned to the `hypothesis` state: they are preserved as hypotheses and routed to literature search, simulation, reanalysis, or laboratory measurement.

Current prototype. The prototype implements an event-sourced claim log, deterministic rebuilds, claim/evidence events, a minimal evidence registry, rule-based baseline impact components, snapshots/diffs, manuscript checks, deterministic verifiers, stored verifier prompts, and a controlled structural-bioinformatics demonstration.

Planned extensions. We extend this substrate with AI claim harvesting, multi-type evidence registries, scalable impact mapping with measured precision/recall, LLM/domain verifiers that recommend but never mutate state, and policy-mediated export gates that return revised text or next tasks. Harvester outputs are candidates, not commitments; candidates missing scope or evidence cannot appear as unqualified results.

3. Controlled Demonstration and Pilot Evaluation

Structural bioinformatics is a useful stress test because experimental structures, predicted structures, protein-family annotations, and literature claims update at different rates. The controlled demonstration is not presented as a biological discovery claim. It tests a common epistemic failure: collapsing predicted-structure evidence into experimental-structure evidence, or generalizing from a tested subset to an entire protein family (Table 1). A successful run emits an epistemic diff and blocks or rewrites abstract/conclusion sentences that promote qualified or refuted claims as unqualified results. We also ran a preliminary N=3 smoke test with deterministic prototype components plus one prompt-only agent-verifier packet (stored prompt/inputs for later replay; no external LLM call; Appendix C). It recorded source hashes, prelabels, 14 claim events, verifier outputs, policy commits, a gate report, and a snapshot. This is a protocol trace, not a powered accuracy result. The initial N=10 human-labeled pilot will calibrate failure categories, labeling guidelines, and ablations using natural manuscript-level claims from public AI-generated papers, internal drafts and database-update scenarios.

4. Evaluation Roadmap

We evaluate ClaimGarden at the process level, complementing outcome-based AI scientist benchmarks such as end-to-end paper generation (Lu et al., 2024; **?**). Metrics are defined operationally. **Unsupported manuscript claim rate** is the fraction of exported strong result claims whose committed status is not `supported` or scoped `qualified`, or that lack registered evidence links. **Refutation response rate** is the fraction of weakening/refuting recommendations that lead under policy to status change, sentence revision, and required plan/task edits. **Claim-state reproducibility** is the fraction of snapshots whose deterministic rebuild yields a hash-identical claim graph. We also track evidence uptake, update propagation latency, human-review burden, impact-mapping precision/recall, and verifier–policy/human agreement. Ablations compare no claim-state layer, no impact mapping, no verifier/policy separation, and export gates with versus without scope/evidence-type checks.

5. Governance and Safeguards

Claim statuses are not truth values, and gate decisions are a separate layer. `supported` means supported under registered evidence, verifier scope, and policy; `EvidenceLinked` records a candidate evidential relation, not proof. Verifier reliability is a first-class governance problem: verifier outputs are recommendations with provenance and uncertainty, not scientific authority. Conservative policies can auto-commit weakening or qualification more readily than promotion to `supported`; export blocking is a gate decision and can require human review for high-impact release decisions.

The manuscript gate flags unsupported strong claims, refuted-as-result text, unscoped qualified claims, predicted evidence described as experimental evidence, and abstracts stronger than results. Sensitive evidence can be hashed or access-controlled; policies can require human review for privacy, biosafety, or dual-use cases. Attribution is recorded. Most importantly, hypothesis, weakened, refuted, obsolete, and inconclusive claims are preserved: a self-correcting AI scientist should remember what it learned not to claim.

Acknowledgements. This work was supported by JSPS KAKENHI Grant Number 26H00484 and JST FOREST Program Grant Number JPMJFR232W.

References

Clark, T., Ciccarese, P. N., and Goble, C. A. Micropublications: A semantic model for claims, evidence, arguments and annotations in biomedical communications. *Journal of Biomedical Semantics*, 5(1):28, 2014. doi: 10.1186/2041-1480-5-28.

Groth, P., Gibson, A., and Velterop, J. The anatomy of a nanopublication. *Information Services & Use*, 30(1–2): 51–56, 2010. doi: 10.3233/ISU-2010-0613.

Jaradeh, M. Y., Oelen, A., Prinz, M., Stocker, M., and Auer, S. Open research knowledge graph: A system walkthrough. *arXiv preprint arXiv:2206.01439*, 2022.

Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., Garijo, D., Soiland-Reyes, S., Zednik, S., and Zhao, J. PROV-O: The PROV ontology. W3C recommendation, World Wide Web Consortium, 2013.

Lu, C., Lu, C., Lange, R. T., Foerster, J., Clune, J., and Ha, D. The AI scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.

Ríos-García, M., Alampara, N., Gupta, C., Mandal, I., Mannan, S., Aghajani, A. A., Krishnan, N. M. A., and Jablonka, K. M. Ai scientists produce results without reasoning scientifically. *arXiv preprint arXiv:2604.18805*, 2026.

Teufel, S., Siddharthan, A., and Batchelor, C. Towards domain-independent argumentative zoning: Evidence from chemistry and computational linguistics. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pp. 1493–1502, Singapore, 2009. Association for Computational Linguistics.

Wadden, D., Lin, S., Lo, K., Wang, L. L., van Zuylen, M., Cohan, A., and Hajishirzi, H. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 7534–7550. Association for Computational Linguistics, 2020.

Xia, Y. and Wang, T. ResearchLoop: An evidence-gated control plane for AI-assisted research. *arXiv preprint arXiv:2605.28282*, 2026.

A. Appendix: Example Command Transcript

This appendix lists a representative command transcript for the controlled demonstration:

```
claimgarden init --workspace W
claimgarden create-claim C1
claimgarden link-evidence C1 pdb_v1.json
claimgarden substrate-update v1 v2
claimgarden impact --changed v2
claimgarden verify --claims impacted
claimgarden adjudicate --policy safe
claimgarden diff
claimgarden gate manuscript.md
claimgarden snapshot --archive
```

B. Appendix: Conservative Policy Sketch

The policy is conservative and asymmetric. Verifiers provide advice, but policy commits status changes. `hypothesis` denotes a harvested claim with no registered supporting or refuting evidence; it is preserved, blocked from export as a result, and routed to follow-up task generation, in contrast to `unverified`, which denotes registered but not-yet-verified evidence.

```
states:
  hypothesis
  unverified
  supported | qualified | weakened |
    refuted
  obsolete | inconclusive
  contested_refutation

on_harvest:
  -> hypothesis if no_registered_evidence
  -> unverified otherwise

auto_commit:
  hypothesis -> unverified
    if first_registered_evidence
  supported  -> qualified
  supported  -> weakened
  unverified -> qualified
  unverified -> refuted
    if hard_rule_violation

review_required:
  unverified -> supported
  qualified  -> supported
  contested_refutation
    -> refuted

next_task_if:
  hypothesis

block_export_if:
  refuted_claim
  unsupported_strong_claim
  hypothesis_stated_as_result
  missing_scope
  prediction_as_experiment
  abstract_stronger_than_results
```

C. Appendix: Preliminary N=3 Pilot Trace

To check that the evaluation protocol is executable, we ran the current prototype on three manuscript-level claims: one controlled structural-bioinformatics update claim, one naturally occurring over-scope claim from a prior semi-autonomous research-workflow output, and one implementation claim from an internal draft of this manuscript. The run used deterministic prototype components and one prompt-only agent-verifier work packet; no external LLM verifier call was made. Here, "prompt-only" means the verifier prompt and inputs were recorded for later replay rather than sent to a model API. This trace checks agreement between human labels and ClaimGarden's policy-committed statuses for three initial manuscript-level claims; full transition dynamics under evidence updates are addressed by the planned N=10 pilot. It recorded source hashes, prelabels, claim/evidence events, verifier outputs, policy decisions, an epistemic diff, a manuscript gate report, and a snapshot archive. This pilot is a smoke test, not a powered accuracy evaluation. System status in the table is ClaimGarden's policy-committed status after the run, not the initial wording of the claim.

Table 2. Preliminary N=3 smoke-test trace. Status is ClaimGarden's policy-committed claim status; Gate is the manuscript action.

Source	Human label	System status	Gate decision
Structural update	refuted	refuted	block result
Prior workflow	qualified	qualified	require scope/evidence
Internal draft	supported	supported	allow with artifact evidence

The run produced 14 claim events over three claims: P1 was committed to `refuted` from a controlled prediction-vs-experiment verification event; P2 was committed to `qualified` by the deterministic scope verifier and conservative policy; P3 was retained as `supported` for a system-implementation claim under local artifact evidence. Status and gate action are intentionally separate layers.

D. Appendix: AI Assistance Disclosure

Generative AI tools, including ChatGPT and Claude-family assistants, supported brainstorming, critique, revision planning, code/prototype assistance, and prose/LaTeX editing. They were also used to help compare reviewer comments, draft alternative phrasings, and prepare camera-ready patch candidates. The author selected the research claims, checked cited sources and implementation descriptions, rejected or edited unsupported suggestions, edited and approved the final version, and takes responsibility for the content.