The One-Person Laboratory Should Be a First-Class Unit of Evaluation in Dry-Lab AI Research

Chaoyue He *, Xin Zhou, Di Wang, Hong Xu, Wei Liu, Chunyan Miao

Posted Date: 20 April 2026

doi: 10.20944/preprints202604.1414.v1

Keywords: one-person laboratory; OPL; AI research automation; autonomous scientific agents; research governance; reproducibility; AI scientist; accountability evaluation metrics; claim verification; dry-lab AI research

 doi:10.20944/preprints202604.1414.v1

Article

The One-Person Laboratory Should Be a First-Class Unit of Evaluation in Dry-Lab AI Research

Chaoyue He [1], Xin Zhou [1], Di Wang [1], Hong Xu [1], Wei Liu [2] and Chunyan Miao [1]

[1] Alibaba–NTU Global e-Sustainability CorpLab (ANGEL), Singapore
[2] Alibaba Group, Hangzhou, China
* Correspondence: weiliu.liuwei@alibaba-inc.com

Abstract

This position paper argues that in software-defined dry-lab AI research, the one-person laboratory (OPL) is the relevant minimum accountable unit under compressed coordination and should be treated as a first-class unit of evaluation wherever bounded verification and public contestability hold. We develop three propositions. **P1 (descriptive):** public research-agent systems and laboratory-shaped benchmarks suggest that the minimum efficient research unit is moving downward in parts of AI research. **P2 (causal, conditional):** the relevant gains are narrower than common "AI scientist" rhetoric implies—not general scientific superiority, but lower iteration latency per admitted claim, stronger provenance, higher replayability, clearer responsibility, and better retention of negative branches when abstention and disclosure are enforced. **P3 (normative, conditional):** the community should therefore evaluate and support OPLs as claim-producing laboratories rather than only models or PDFs, while simultaneously building public execution interfaces, trace-linked claim standards, benchmark sandboxes, and access institutions. Our empirical anchor is a purposive structured interpretive read of representative public systems and benchmarks; it is not a leaderboard and does not estimate prevalence, causal impact, or superiority. We do not claim that OPLs replace strong teams, justify broad scientific claims from a single run, or cleanly extend to wet-lab, clinical, or human-subject domains. The paper's contribution is a falsifiable governance position: if laboratory-shaped systems fail to cohere, if OPL-style runs do not improve admitted-claim speed or auditable process quality, or if access remains closed, the thesis should be weakened or reversed.

Keywords: one-person laboratory; OPL; AI research automation; autonomous scientific agents; research governance; reproducibility; AI scientist; accountability; evaluation metrics; claim verification; dry-lab AI research

1. Introduction

AI research has entered a recursive phase: the field increasingly uses AI systems to search literature, write code, manage experiments, and critique ideas. Foundation models, tool-use scaffolds, and long-horizon agents make this recursion more than a productivity story; they alter the practical organization of research itself [1–7]. As the same computational stack begins to assist across the entire loop, the relevant question shifts from whether a model can complete an isolated subtask to identifying the smallest accountable unit that can repeatedly operate that loop without sacrificing evidence quality.

That compression is not automatically good. It can become hidden-labor laundering, private compute advantage, mentorship erosion, or spam multiplication. Those risks are precisely why the normative step below is conditional on replayable evidence, public contestability, and governed release [8–15].

This position paper argues that in software-defined dry-lab AI research, the OPL is the relevant minimum accountable unit under compressed coordination. Consequently, it should be treated as a first-class unit of evaluation wherever bounded verification and public contestability hold.

To structure this claim, we advance three interconnected propositions:

 doi:10.20944/preprints202604.1414.v1

P1 (descriptive).

In dry-lab AI research, the minimum efficient research unit is moving downward. Systems such as Agent Laboratory, The AI Scientist, The AI Scientist-v2, the AI co-scientist, and DeepAnalyze now span multiple phases of the research cycle rather than executing narrow, isolated tasks [16–20]. Concurrent evaluations such as RE-Bench, PaperBench, ResearchGym, DSGym, MLGym, and MLR-Bench increasingly measure open-ended, executable research workflows rather than only local generation quality [21–26].

P2 (causal, conditional).

OPLs can accelerate iteration and improve *auditable process quality* when the research loop is instrumented around provenance, replayability, abstention, negative-result retention, and governed release [8–10,27,28]. The relevant improvement claim is deliberately narrow: not universal scientific superiority, but lower iteration latency per admitted claim, stronger evidential binding, and clearer responsibility under disclosed constraints. Absent those controls, smaller AI-native labs can just as easily generate plausible but invalid paper-shaped artifacts at unprecedented scale [8–10,29].

P3 (normative, conditional).

The community should therefore evaluate and support OPLs as first-class units of evaluation, but only alongside public execution interfaces, trace-linked claim standards, benchmark sandboxes, and access institutions. The operative implication is not universal replacement for teams. It is that, in settings where bounded verification is feasible, trust increasingly attaches to the lab-shaped claim-producing unit rather than to the model alone or the PDF alone [11–15].

Public systems and benchmarks already show the laboratory-shaped unit emerging, but no current public system fully instantiates the OPL standard defended here. What still lags are governance, replayability, and release; once loop coverage outruns release discipline, evaluation has to move from isolated models and polished PDFs toward labs and claim packages, because that is where trust is now won or lost. We focus on *dry-lab AI research* over the next three to five years, especially workflows in which code, evaluations, logs, traces, notebooks, and benchmark interactions are heavily software-defined. We do not extend the core claim to wet-lab, clinical, human-subject, or deeply tacit workflows, where bounded replay and auditable execution are much harder to sustain [30–33]. Nor do we claim that OPLs generally beat strong small teams, justify broad scientific claims from a single run, or turn manuscript throughput into scientific quality.

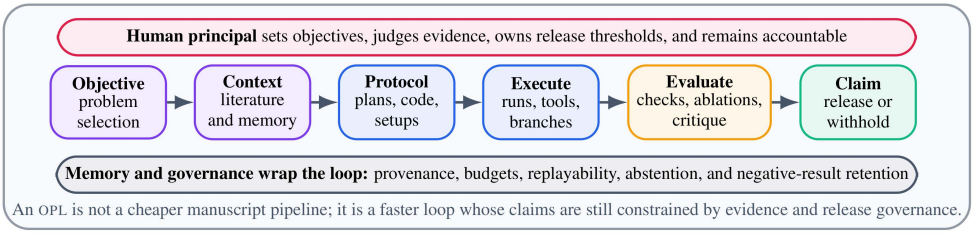

Figure 1. The one-person laboratory as a governed research loop. Compression helps only when execution and release remain tied to memory, provenance, replayability, and abstention.

2. The Organizational Shift: From "AI Scientist" to the OPL

The phrase "AI scientist" is rhetorically powerful but analytically weak. It blurs four distinct objects: a model, a workflow, an institution, and a social role. That ambiguity encourages anthropomorphic comparisons and invites the wrong benchmark question: whether an agent can resemble a scientist. The more useful question is organizational: *what accountable unit is actually emerging as the loop gets compressed?* The OPL is a better answer because it names the unit that selects objectives, accumulates memory, governs release, and can be audited when something goes wrong.

 doi:10.20944/preprints202604.1414.v1

A useful non-example keeps the schema discriminative. SWE-agent on SWE-bench-style repository tasks is a strong execution system [34,35]. It can search code, edit files, run tests, and return patches, but it does not yet meet the OPL standard defended here. Its public object is successful issue resolution, not a governed claim package; its release object is a patch or trajectory, not an admitted scientific claim; and its evaluation does not center abstention conditions, withheld branches, or accountable claim release. This is exactly why the schema is not absorptive: strong execution alone does not make a system an OPL.

The one-person-company (OPC) analogy is useful because it highlights *which overheads are being compressed* (summarized in Table 1). Research includes at least two kinds of cost. There is the domain cost of generating or testing ideas. There is coordination overhead: searching and cleaning literature, turning vague objectives into executable protocols, setting up environments, keeping track of branches, generating intermediate analyses, formatting artifacts, and preparing a case for release. Much of the current excitement around scientific agents comes from AI starting to attack this category at scale [36–38]. That matters for both speed and quality: fewer handoffs can shorten iteration, and a tighter operating loop can reduce information loss between idea, experiment, evidence, and release.

Table 1. Why the one-person laboratory is the research analogue of the one-person company. The analogy is organizational, not anthropomorphic.

Dimension	OPC (business analogue)	OPL (AI-research analogue)	What changes after compression
Compressed overhead	bookkeeping, storefronts, payments, support, contracting	literature search, coding, experiment orchestration, analysis, reporting	one operator can steer a much larger process
Primary operator	founder / owner	accountable researcher / principal	control concentrates even if dependencies remain distributed
Core stack	cloud software, payment rails, contractors	foundation models, agents, compute, benchmarks, provenance, governance	the "lab" becomes partly software-defined
Primary output	products and services	admissible claims, models, datasets, and protocols	output quality depends on release discipline, not just generation speed
New bottleneck	distribution and trust	objective setting, evaluation, oversight, access to execution interfaces	leverage rises faster than verification capacity
Main failure mode	scaling without controls	paper-shaped outputs without valid evidence	governance becomes a first-class technical problem
Ecosystem support needed	cloud credits, legal wrappers, accelerators	claim ledgers, lab APIs, benchmark sandboxes, compute/lab credits	institutions determine whether compression democratizes or centralizes

This is not a claim that teams no longer matter. Science has become increasingly team-based, and teams dominate high-impact knowledge production [39–41]. Nor is OPL a romantic theory of solitary genius. An OPL still depends on public data, benchmark designers, infrastructure builders, instrument maintainers, and domain experts. The phrase "one-person" names the smallest accountable unit that can operate the loop, not the full social ecology required to sustain science [11,15].

The analogy also reveals the new bottleneck. When starting a company became easier, the hard problems did not disappear; they shifted toward distribution, trust, capital access, and governance. The same pattern is now visible in AI research. If agents compress execution overhead, the hard problems become objective selection, evaluation design, release thresholds, access to execution interfaces, and whether the community can trust outputs that are increasingly cheap to produce but still expensive to verify [42–45].

It is helpful to name the economic object underneath this shift: the *minimum efficient research unit*. By this we mean the smallest accountable configuration that can repeatedly produce admissible claims in a given domain. For much of modern AI research, that unit has usually been a small team because literature triage, coding, compute management, evaluation, and artifact preparation created too much overhead for one researcher to sustain at pace. The OPL thesis is that AI-native tooling is beginning to push this threshold downward in parts of the field. That does not imply solitary science; it implies that the minimal operating unit is becoming smaller, more software-defined, and more sensitive to ecosystem design.

 doi:10.20944/preprints202604.1414.v1

A natural objection is that nothing here requires a specifically one-person unit; perhaps the real lesson is simply that the field needs a better small-team lab/process standard. We keep OPL explicit because it isolates the boundary case in which coordination compression has already collapsed enough that one principal can bind objective selection, memory, execution, and release inside one accountable loop. A generic small-team standard does not tell us whether coordination loss has actually disappeared or merely moved offstage. The OPL label is useful precisely because it can fail: if hidden operators, repair teams, or privileged maintainers remain load-bearing, then the workflow is not an OPL in the sense that matters here.

That framing also makes the boundary conditions easier to state. The OPL thesis should be strongest where execution is already digital, reruns are relatively cheap, key decisions can be externalized into traces and protocols, and verification can be bounded through replay and spot-checking. It should be weaker where tacit knowledge, regulated intervention, human subjects, or expensive physical execution dominate. The right comparison is therefore not "OPL versus teams" in the abstract, but OPL versus alternative organizational forms under domain-specific coordination, verification, and access conditions. Table 2 makes that comparison explicit.

Table 2. Where the OPL thesis should be strongest, where it should hybridize, and where it should retreat. The comparison is domain-conditional rather than universal.

Condition	OPL-first	Hybrid / networked OPLs	Team / facility-first
Execution substrate	code, containers, benchmarks, notebooks, and software-defined APIs	mostly digital loop with some external instruments or privileged interfaces	wet-lab execution, fieldwork, clinical workflows, or heavy human-subject interaction
Tacit knowledge intensity	low to moderate; much of the workflow can be externalized into traces and protocols	moderate; specialists or domain review remain important at some steps	high; crucial know-how is embodied, local, or safety-sensitive
Verification cost	bounded replay and claim spot-checks are feasible	partial replay plus domain-specific audit is feasible	verification is slow, expensive, or institutionally regulated
Failure externality	low to moderate; scoped release and abstention can manage error	mixed; escalation points and external review matter	high; release must be governed by teams, facilities, or formal oversight
Best organizational reading	OPL can be the primary operating unit	networks of OPLs embedded in larger programs	OPL is useful as an assistant layer, but not as the primary unit

Why OPL can make AI research faster—and when it can make it better.

The case for OPLs is strongest when speed and quality are treated together. If smaller AI-native labs merely produced more manuscripts faster, the result would be noise. The stronger claim is narrower: when the loop is instrumented well, the same organizational compression that speeds research can also improve evidence quality. Many delays in AI research come not from insight itself but from handoffs: turning ideas into executable protocols, moving between literature search and code, synchronizing branches, repairing environments, and packaging results for others to inspect. When one accountable researcher can steer a single AI-native stack across those steps, iteration becomes shorter and more continuous [12,46–49].

Smaller does not automatically mean better, but it can improve quality when the release process becomes easier to govern. In a multi-handoff workflow, responsibility for prompt choices, data cleaning, environment fixes, metric definitions, and withheld failures can become diffuse. In an OPL, those decisions are more likely to sit inside one governed trace with one accountable principal. That can make it easier to inspect hidden labor, detect leakage, preserve negative results, and ask whether a claim should be released at all [8–11,29].

Faster loops do not inherently produce better science. They produce *more leverage*. That leverage can widen search, improve memory, and preserve more evidence, but it can also widen spam, benchmark gaming, and premature release. The same acceleration that helps one careful independent researcher run a serious replication loop can help one careless actor generate a polished but weak manuscript. That is why the argument of this paper is organizational rather than triumphalist: what matters is *claim throughput under evidence and governance constraints*.

 doi:10.20944/preprints202604.1414.v1

AI research is unusually exposed to the OPL transition because the object of study is also becoming part of the production technology of the field. Models that are being improved by AI research are also being used to conduct that research. This recursive leverage is unusually strong in AI [16,36,37]. AI research is also unusually software-defined: hypotheses often become code changes, experiments become replayable runs, and results become benchmark scores, logs, and artifacts [50–59]. That makes it the natural early proving ground for an argument about smaller accountable research units.

3. Scope, Definition, and an Operational Admission Rule

We define a one-person laboratory as a *closed-loop, AI-augmented research system operated under the responsibility of a single human principal*. Formally, as expressed in Equation 1, let

$$\mathcal{L} = (h, \mathcal{A}, \mathcal{T}, \mathcal{M}, \mathcal{G}, B), \tag{1}$$

where h is the accountable human researcher, $\mathcal{A}$ is a set of agents or models, $\mathcal{T}$ is a set of tools, environments, and external interfaces, $\mathcal{M}$ is persistent laboratory memory, $\mathcal{G}$ is a governance layer, and B is a budget and resource policy. The formalism should earn its keep. Instead of treating $\mathcal{L}$ as decorative notation, we tie it to a concrete admission object: the *claim package*. For any run with execution trace τ, let a candidate claim package (Equation 2) be

$$\begin{aligned} P(c,\tau) = \{&\texttt{claim}, \texttt{scope}, \texttt{evidence}, \texttt{trace}, \texttt{environment}, \\ &\texttt{budget}, \texttt{failed}, \texttt{abstain}, \texttt{release}, \texttt{replay}\}. \end{aligned} \tag{2}$$

A package is structurally complete if all mandatory fields required for the claim type are present. For a given claim type, let $Q(c)$ denote its required validation checks. For example, a benchmark-improvement claim may require leakage checks, baseline checks, seed stability, and ablations; a systems claim may require hidden-human-labor disclosure and scope restrictions. We define structural completeness and evidence sufficiency jointly (Equation 3) as:

$$\mathrm{Req}(P(c,\tau)) = \mathbb{I}[\text{all required fields for } c \text{ are present}], \quad \mathrm{Evid}(c,\tau) = \tfrac{1}{|Q(c)|} \textstyle\sum_{q \in Q(c)} \mathbb{I}[q \text{ passes}]. \tag{3}$$

Define replay success under tolerance δ over k reruns (Equation 4) as

$$\mathrm{Replay}_{\delta}(c,\tau) = \frac{1}{k} \sum_{j=1}^{k} \mathbb{I}\big[|m_j - m^{\star}| \le \delta\big], \tag{4}$$

where $m^{\star}$ is the reported metric and m_j is the replayed metric on rerun j. Finally, let disclosure be defined as in Equation 5:

$$\mathrm{Disc}(\tau) = \mathbb{I}[\text{manual interventions, external assets, and unresolved issues are disclosed}]. \tag{5}$$

Let $\mathrm{Gate}(c,\tau)$ denote the governance release decision. A candidate claim is admitted only if all conditions in Equation 6 are met:

$$\mathrm{Admit}(c,\tau) = \mathbb{I}\Big[\mathrm{Req}(P(c,\tau)) = 1 \wedge \mathrm{Evid}(c,\tau) \ge \alpha \wedge \mathrm{Replay}_{\delta}(c,\tau) \ge \beta \wedge \mathrm{Disc}(\tau) = 1 \wedge \mathrm{Gate}(c,\tau) = 1\Big]. \tag{6}$$

Here α, β, δ, and any claim-class defaults are policy placeholders, not calibrated scientific constants. Readers should treat Equations 1–6 as a policy schema rather than a measurement theorem: the argument of the paper does not depend on any particular cutoff, only on the asymmetry that broader claims should face stronger evidential burdens than narrow, replayable ones. Figure 2 makes the claim-package object concrete in the main text, while Appendix B provides fuller illustrative defaults and Appendix C collects the review and actor matrices.

 doi:10.20944/preprints202604.1414.v1

This admission rule is intentionally plain. Its purpose is to make explicit that an OPL should be judged by the claims it can *responsibly admit*, not by the artifacts it can fluently generate. That matters because recent critiques of AI-scientist systems show that scientific validity can fail even when the final document looks competent [8,10,60,61].

What "better" means in P2.

Unless otherwise stated, "better" first means improvement on *auditable process dimensions*: lower iteration latency per admitted claim, stronger provenance, higher replayability, better preservation of failed branches, clearer responsibility, and stricter release discipline. It does not by itself mean universal superiority on novelty, impact, or scientific importance. Those stronger notions require broader evidence than this paper claims.

A claim ladder, not one universal bar.

The paper relies on a claim ladder rather than a threshold. Narrow executable claims can sometimes be responsibly released from a single governed run, while broad field-level or scientific claims typically require multi-setting evidence or outside corroboration. In compact form: **E0** exploratory branches should preserve trace but make no public claim; **C1** scoped executable claims may be released from one governed run if replay and disclosure pass; **C2** process or system claims require C1-style checks plus labor disclosure, branch accounting, and budget normalization; and **C3** broad scientific or field claims should usually be withheld, downgraded to synthesis, or backed by outside corroboration. Appendix Table A8 makes the defaults explicit. The operative rule is simple: the broader the claim, the harder it should be to release.

Several clarifications follow: OPL is not a model, because a model produces tokens while an OPL produces claim packages under governance; it is not a manuscript pipeline, because the paper is a view over the evidential object rather than the object itself [28,62–65]; it is not "solo science," because an OPL may sit inside a larger team or network of labs; and it is domain-sensitive, because execution and admissibility thresholds differ sharply across dry-lab and wet-lab settings [13,14,31,66].

Table 3. Representative public systems through the OPL lens. The main asymmetry is that loop coverage is advancing faster than replayability and release governance.

System	Dominant loop span	Trace visibility	Replay grounding	Release governance	What the system contributes to the OPL picture
Agent Laboratory [16]	ideation, literature, coding, experiments, drafting	medium	partial	weak	Shows that multi-stage research assistance can be composed into one researcher-steerable loop, but the public object remains closer to assistant workflow than governed claim admission.
The AI Scientist [17]	ideation, experimentation, report generation	medium	partial	weak	Makes the paper-production risk legible: loop coverage is impressive, yet admissibility and release discipline are still underspecified.
The AI Scientist-v2 [18]	broader search over ideas and branches	medium	partial	weak	Strengthens the case that search breadth is rising, while also underlining that better search is not the same as better governed claims.
AI co-scientist [19]	collaborative hypothesis generation with stronger human steering	medium	partial	medium	Makes human steering more explicit and therefore sits closer to the accountable-lab framing than anthropomorphic "scientist" talk.
DeepAnalyze [20]	autonomous data-science loop in software-defined settings	medium	medium	weak	Demonstrates why dry-lab settings are the natural early home for OPL arguments: the execution substrate is already largely digital.

4. A Structured Empirical Anchor and Evidence Map

We use a structured retrospective empirical anchor: a reading of representative public systems through the OPL lens. The coarse coding rubric appears in Appendix A; Table 3 reports the main readout. The anchor is a structured interpretive read, not a leaderboard. It does not estimate prevalence, causal impact, or superiority. Its role is to discipline the descriptive claim and reveal which OPL components are already public versus lagging. The selection is purposive rather than exhaustive.

 doi:10.20944/preprints202604.1414.v1

We include public systems that (i) span several linked stages of the research loop, (ii) are discussed as research automation rather than narrow copilots, (iii) expose enough public detail to judge loop span, trace visibility, replay grounding, and release governance, and (iv) operate primarily in software-defined dry-lab settings. We exclude purely local assistants, fully closed internal systems, and wet-lab-first platforms whose execution substrate would make the comparison much less interpretable. Appendix Table A7 makes the desk-analysis logic more transparent through a worked contrast between one system paper and one benchmark paper.

Sensitivity to omitted public systems and benchmarks.

Table 3 is purposive rather than exhaustive. Omitted public artifacts such as SWE-agent, SWE-bench, WebArena, AgentBench, and GAIA would not reverse the paper's main asymmetry. Some would modestly weaken the broadest reading of P1 if treated as candidate research labs, because they cover narrower slices of the loop than the systems in Table 3; but they would still leave unchanged, and in several cases sharpen, the central pattern that execution span and task realism are advancing faster than replayability, abstention, and governed claim release [34,35,67–69]. Including them would widen the negative-control set more than it would undermine the paper's core asymmetry. That sensitivity point is easier to read if the units are separated explicitly. Table 4 gives a compact comparison.

Table 4. Unit distinction in the main argument. The OPL is the minimum accountable unit once objective selection, memory, execution, and release collapse into one governed loop.

Question	Model	Agent / workflow	OPL	Small team
Accountable principal	builder or deployer sits outside the artifact	often blurred between user, builder, and runtime controller	one named human principal owns release thresholds	responsibility is distributed across named contributors
Public object	weights, API behavior, or benchmark score	trajectories, tool calls, patches, or task completions	claim package plus paper view	paper, artifacts, and project portfolio
Release object	model or endpoint	execution outcome or workflow report	admitted scoped claim under governance	team-level claims and artifacts
Review object	capability, safety, or benchmark behavior	task success, trace quality, or tool use	trace-linked claim package and release discipline	paper, artifacts, and division of labor
Main failure mode	fluent output mistaken for evidence	execution success mistaken for scientific validity	over-release, hidden labor, or weak abstention inside a governed loop	coordination loss or diffusion of responsibility

Two patterns matter. First, current systems already make it hard to argue that research automation remains purely local or assistant-like. They span enough of the loop to make organizational questions unavoidable. Second, the lagging components are exactly the ones our thesis emphasizes: replayability, disclosure, abstention, and release governance. Public systems therefore support the descriptive claim that OPL is emerging, while also supporting the caution that the field is not yet instrumented well enough to treat speed as quality.

Benchmarks reinforce the same point from the opposite direction. RE-Bench, PaperBench, ResearchGym, DSGym, MLGym, MLR-Bench, omics research benchmarks, BixBench, and Replicator-Bench increasingly evaluate laboratory-shaped tasks rather than isolated generation [21–26,70–72]. The ecosystem is therefore already converging on the laboratory as the relevant unit of performance, even if publication norms still lag behind.

What is already known, and what remains uncertain.

The descriptive claim that OPLs are emerging is already supported by public systems and benchmark design. The causal claim that OPLs improve research is plausible only in the narrower sense emphasized here—better auditable process quality under governed release—and still needs matched-budget evidence against strong small teams. The normative claim is strongest only if public infrastructure, access, and contestability are treated as first-class design constraints. Appendix Table A6 separates what is already well supported from what remains uncertain.

Taken together, the empirical anchor makes the descriptive-to-normative bridge explicit. Public systems and benchmarks show the lab-shaped unit emerging; the lagging dimensions are governance,

 doi:10.20944/preprints202604.1414.v1

replayability, and release; therefore the relevant object of evaluation is increasingly the lab-shaped claim package rather than the model alone or the PDF alone.

5. Operationalizing the OPL

The paper's normative claim depends on a concrete operational object, not just on abstract equations or manuscript-level rhetoric. An OPL should be evaluated as a claim-producing laboratory rather than as a manuscript-producing model. In practice, that means that every central claim should bind together scope, evidence, trace, environment, budget, failed branches, abstention conditions, and a human release decision. Figure 2 makes that object visible in the main text.

That shift changes what gets optimized. Instead of rewarding only fluency or paper completion, the field should reward admitted claims per normalized budget, replay success, explicit accounting of human interventions, correct withholding of weak branches, and preservation of negative results. Appendix B expands this layer with illustrative claim-class defaults (Table A8), anti-gaming proto-metrics (Table A9), and a toy worked example; Appendix C gives a bounded review workflow and an actor-specific implementation matrix (Tables A10 and A12).

Review can remain bounded. Rather than auditing every branch, venues can sample a small number of load-bearing claims, inspect trace summaries and intervention logs, and rerun only a few central artifacts under disclosed tolerances. Appendix Table A10 sketches one concrete workflow. This is the institutional counterpart of the paper's main thesis: if the lab is the relevant object of trust, then review should sample the lab-shaped evidential object rather than only the final PDF.

Example failure mode: semantic divergence between intent and trace.

Fast OPL loops can still drift away from the human principal's scientific intent even while improving local metrics. We use "semantic divergence between intent and trace" to name this failure mode: the run satisfies the harness syntactically, but the substantive scientific objective has thinned out. In dry-lab AI research, that shows up as benchmark gaming, degenerate edge-case exploitation, or paper-shaped artifacts whose local success does not carry much scientific value. The governance layer should therefore treat domain grounding, intermediate checks, and disclosure of major steering interventions as admission conditions rather than optional extras. Appendix B sketches one mitigation family, but those details are illustrative rather than core thesis machinery.

Four immediate red-team questions.

Any system asking to be treated as OPL-like should survive four simple challenges before the community grants laboratory-level trust: *What hidden labor made the run succeed? Which claims survive replay? Which branches were withheld and why? What evidence would have triggered abstention rather than release?* These questions are deliberately lightweight, but they make the paper's governance stance concrete: if a workflow cannot answer them, it should not yet be treated as a governed claim-producing unit [8–11].

6. Alternative Views

Five alternative views are especially important.

View 1: Team science persists for reasons deeper than coordination cost.

Teams are not just expensive coordination shells. They preserve tacit knowledge, enable peer correction, support apprenticeship, and allow domain specialization [30,39,40]. **Response:** This view is strong. Our response is limited: OPL is not a theory of universal replacement. It is a theory about the minimum efficient *operating* unit in some software-defined workflows. Many healthy institutions will look like *networks of* OPLs embedded inside larger teams and facilities, not a world without teams.

 doi:10.20944/preprints202604.1414.v1

CLAIM PACKAGE MANIFEST

Claim ID: `opl-toy-001`
Claim Text: "On a disclosed replication task, branch b3 improves the normalized score over the released baseline under the stated container, seeds, and budget."

Scope
domain: dry-lab AI research
limits: single benchmark family; no generalization claim beyond this task class

Trace
run_id: `run-2026-04-opl-001` **code_commit:** `8f3ac1d`
tool_calls: `[retrieve_paper, edit_code, launch_run, eval_script]`
container: `opl-wa-cu124`

Evidence & Replay
primary_runs: `[r41, r42, r43]` **replay_runs:** `[rr1, rr2, rr3]`
metric: normalized_score **tolerance:** 0.5
replay_cmd: `python run.py -task WA07 -seed 3` **budget_cap:** 6 GPUh, 2 human-h

Governance & Release
failed_branches: `b1` (leakage risk), `b2` (improvement vanishes on replay)
abstention_conditions: withhold if replay success $< 2/3$ or undocumented fixes required
release_decision: `human_principal: accepted` (scoped claim passes disclosure thresholds)

Figure 2. Illustrative claim package for a stylized dry-lab replication branch. The point is the structured release object rather than the benchmark itself.

View 2: OPL will worsen spam and hype.

Lowering the fixed cost of paper-shaped work can flood the field with cheap candidate claims. Current autonomous research systems already show how easy it is to optimize for manuscript completion [17,18,26]. **Response:** This view is also strong. The right countermeasure is laboratory-level governance: claim-package standards, withheld-branch disclosure, replay scoring, and stronger burdens for automation-heavy submissions.

View 3: The "one-person" label launders hidden maintenance labor.

Data cleaning, environment repair, evaluation upkeep, and infrastructure maintenance do not disappear simply because one visible operator sits at the top of the loop [11]. **Response:** That is why labor disclosure is part of the admission rule and why hidden manual cleanup should count against the oversight accounting rather than disappear into the background. A credible OPL narrative must reveal its dependencies rather than hiding them.

View 4: OPL could erode mentorship and apprenticeship.

If institutions interpret lab compression as a reason to shrink training structures, the result could be a faster but shallower research culture. **Response:** This concern should be taken seriously in dry-lab AI research. The answer is institutional rather than rhetorical: build OPL fellowships, trace-audit training, and shared lab infrastructures that strengthen scientific judgment rather than treating automation as a substitute for formation.

View 5: Compute concentration may make OPL anti-democratic in practice.

Smaller operating units do not automatically imply broader access. If the best execution substrates remain closed, then OPL may simply amplify the advantage of actors with proprietary models, privileged APIs, or exclusive lab interfaces [12,13,15]. **Response:** This is the strongest argument for making public infrastructure central rather than peripheral to the thesis. P3 is therefore explicitly conditional: the field should evaluate and support OPLs only alongside access institutions that keep them contestable and not merely private.

Taken together, these alternative views do not kill the OPL thesis. They sharpen it. The result is neither lone-genius mythology nor naive agent triumphalism. It is a research ecosystem in which smaller accountable labs can operate productively without erasing teams, mentorship, hidden labor, or the politics of infrastructure. In practical terms, these objections narrow the thesis from a general superiority claim to a conditional governance claim: OPLs are the right unit of evaluation only where dry-lab conditions, bounded verification, and public contestability can be maintained. Appendix Table A15 lists concrete outcomes that would materially weaken or reverse parts of P1–P3.

 doi:10.20944/preprints202604.1414.v1

7. Institutional Agenda: Evaluating and Supporting OPLs

If the goal is faster, better science rather than cheaper paper production, the relevant question is what ecosystem makes smaller AI-native labs rigorous, contestable, and broadly accessible. The lesson from the OPC analogy is that organizational forms spread when surrounding infrastructure makes them viable. One-person companies did not emerge because people wrote essays about founders. They emerged because cloud infrastructure, software-as-a-service, payment rails, marketplaces, and legal templates compressed overhead [12,48,73,74]. The OPL will require analogous scientific infrastructure; Appendix Tables A11 and A12 spell out that translation in detail.

Four priorities follow: open execution substrates such as containerized environments, benchmark sandboxes, provenance stores, experiment ledgers, retrieval snapshots, tool APIs, and, where appropriate, instrument interfaces [13,14,55,58,59,75–77]; standardized claim packages and trace audit so strong automation claims expose plans, retrieval context, code diffs, environment manifests, budget summaries, failed branches, release decisions, and disclosed human interventions [9,28,29,63–65]; lab-level benchmarks that score admissible claim production, abstention, and failure retention under bounded compute, time, tool-access, and oversight budgets [21–26,70,72]; and institutions for access, training, and contestability, including compute credits, microgrants, public claim-package repositories, and fellowships that teach objective selection, falsification design, trace auditing, and abstention [11,13–15].

Those priorities distribute across actors. Venues should request claim IDs, trace summaries, intervention disclosure, and replay instructions. Benchmark builders should score admissible claim production and abstention. Funders and infrastructure providers should provide compute credits and public sandboxes. Labs should keep sealed branch manifests and explicit release thresholds, while independent auditors should sample and replay a few load-bearing claims from high-impact runs. Appendix Table A12 expands this matrix.

Three near-term predictions.

If the thesis here is right, then by 2027–2028 at least some laboratory-shaped benchmarks should begin to score abstention, replay, or claim-package exposure in addition to task completion; more automation-heavy submissions should expose claim IDs, trace summaries, or replay instructions even before venues make them mandatory; and independent replication or audit groups should begin using OPL-style workflows to contest high-profile dry-lab claims under explicit budgets. These are forecasts rather than facts, and they provide another route for weakening the thesis if they fail to materialize. The practical questions are simple: who can run an OPL, who can verify one, and who can contest one? A healthy research ecosystem requires all three.

8. Conclusions

In software-defined dry-lab AI research, the OPL is the minimum accountable unit of compressed coordination. Moving beyond vague "AI scientist" rhetoric, we argue that scientific trust must shift from isolated models or PDFs toward governed laboratories that produce auditable claim packages. Consequently, OPLs should be treated as first-class units of evaluation wherever bounded verification and public contestability hold. This governance position is strictly falsifiable: if public systems fail to cohere into governed labs, if OPL-style runs do not improve auditable process quality, or if infrastructure remains closed, the thesis must be reversed.

Acknowledgments: This research is supported by the RIE2025 Industry Alignment Fund (Award I2301E0026) and the Alibaba–NTU Global e-Sustainability CorpLab.

Appendix overview.

The appendices primarily serve one job: to make the paper's operational credibility easy to inspect. Appendices A–C contain the interpretive rubric for the empirical anchor, a worked desk-analysis contrast, detailed claim-class defaults, anti-gaming proto-metrics, and the review/ecosystem matrices.

 doi:10.20944/preprints202604.1414.v1

The later appendix sections retain supporting context, boundary conditions, and implementation aids for readers who want the broader scaffolding.

Appendix A. Supplementary Empirical Materials

The tables in this section keep the interpretive scaffolding visible.

Table A5. Coarse coding rubric for the structured retrospective anchor. The purpose is interpretive discipline, not pseudo-precision.

Dimension	What counts as higher evidence of an OPL-like unit	What keeps the score lower
Loop span	multiple linked stages are covered, such as ideation, implementation, execution, evaluation, and drafting	support remains local to one step or one narrow task
Trace visibility	branch structure, tool calls, intermediate artifacts, or run summaries are visible in the paper or repo	only the final narrative or final output is visible
Replay grounding	released artifacts plausibly support rerun, reconstruction, or partial replay	there is no clear path from claim to rerun
Release governance	abstention, failure retention, disclosure, or release thresholds are part of the system story	release is implicit, paper-centric, or weakly governed

Table A6. Evidence map for the paper's central claims. The table intentionally separates what is already well supported from what remains uncertain.

Major claim	Evidence type	What is already known	What is still uncertain
OPLs are emerging in dry-lab AI research	public system papers and benchmark design	loop-spanning systems and lab-shaped evaluations already exist [16–18,21–23]	how often those systems produce *admissible* claims rather than plausible artifacts under independent audit
OPLs can accelerate research	organizational theory, cloud/software analogies, long-horizon agent evidence	coordination compression and task-horizon gains are real and strategically relevant [12,37,46,47,49]	the effect size relative to strong small teams on real research tasks under equal budgets
OPLs can improve quality when instrumented	reproducibility literature, provenance systems, hidden-pitfall analyses	provenance, replay, and richer documentation improve auditability and make some failure modes more visible [8–10,27,28,62]	whether these controls will be adopted by default and how resistant the resulting metrics are to gaming
Public infrastructure determines whether OPLs democratize or centralize research	digital-infrastructure work and social critiques of AI-for-science	access to compute, interfaces, and hidden labor strongly shapes who benefits from automation [11–15]	which institutional mechanisms best preserve openness, contestability, and training value as lab automation spreads

Worked desk-analysis contrast.

The main-text anchor is intentionally synthetic. Table A7 makes that synthesis more transparent by contrasting one system paper and one benchmark paper field-by-field under the same OPL schema.

 doi:10.20944/preprints202604.1414.v1

Table A7. Worked desk-analysis contrast between one system paper and one benchmark paper under the same OPL schema. The goal is transparency of interpretation, not a hidden scoring procedure.

Schema field	*The AI Scientist* [17]	*PaperBench* [22]	Interpretive takeaway
Primary public object	a loop-spanning autonomous research system aimed at ideation, experimentation, and report generation	a benchmark for replicating AI research papers under executable constraints	system papers and benchmark papers expose different slices of the same laboratory picture
Strongest visible contribution	broad multi-stage loop compression within one steerable workflow	strong execution grounding on real replication-style tasks	current public artifacts already assume laboratory-shaped work, but at different layers
What is publicly easiest to inspect	how much of the loop can be covered by one agentic stack	how well an agent can execute and replicate within a realistic harness	loop span and evaluation realism are advancing on partially separate tracks
What still looks weak under the OPL schema	explicit claim admission, abstention, and release governance remain thin	objective selection and governed claim release are largely outside the benchmark object	neither public object yet fully centers trace-linked claim release
Main OPL lesson	loop coverage can now be compressed into one researcher-steerable system	benchmark design is already moving from local subskills toward lab-shaped execution	the missing connective tissue is a shared claim-package and review standard

Appendix B. Supplementary Operational Materials

This section extends the main-text operationalization with a claim ladder and proto-metrics for inspectable, replayable, and reviewable OPL outputs.

Table A8. Illustrative claim classes and admission defaults. The thresholds are policy placeholders, not validated scientific constants. Their purpose is to make broad claims harder to release than narrow, replayable ones.

Claim class	Typical example	Checks emphasized	Illustrative defaults	Default release stance
E0: exploratory branch	"interesting direction" or internal hypothesis note	trace retention, branch metadata, no public claim admission	no public α, β threshold	keep internal or release only as a clearly non-claiming note
C1: scoped executable claim	"under the disclosed container and seeds, branch b3 improves score on task family X"	environment manifest, seeds, baselines, leakage checks, replay instructions	$\alpha \geq 0.75$, $\beta \geq 2/3$	may release as a narrow claim if disclosure passes
C2: process / system claim	"this OPL workflow reduces time-to-admitted-replication under the stated harness and budget"	C1 checks plus labor disclosure, branch accounting, and budget normalization	$\alpha \geq 0.80$, $\beta \geq 0.80$	release with explicit scope restrictions and failure disclosures
C3: broad scientific or field claim	"OPLs generally improve research quality" or "this result generalizes broadly"	C2 checks plus multi-setting evidence, or independent external corroboration	$\alpha = 1.00$, $\beta \geq 0.80$ plus external support	withhold, downgrade, or explicitly label as position/synthesis absent outside evidence

 doi:10.20944/preprints202604.1414.v1

Table A9. Operationalized OPL proto-metrics. A, B_n, H, W, W_h, F, and F_p are defined in text. The point is not to create a single leaderboard score, but to make laboratory-shaped outputs red-teamable and harder to game by default.

Metric	Formula	Measurement procedure	Main gaming risk	Proposed mitigation
ACY	ACY $= A/B_n$, where A is the number of admitted claims and B_n is a normalized budget index over compute, wall-clock time, and human oversight	score only claims that pass admission under a predeclared budget family	splitting one result into many tiny claims	cluster claims with shared evidence and cap within-run slicing
RSR	RSR $= \frac{1}{A}\sum_{i=1}^{A} s_i/k_i$, where s_i is the number of successful replays for admitted claim i over k_i reruns	replay sampled admitted claims under disclosed environments and tolerances	choosing overly loose tolerances or replay settings	use venue- or benchmark-owned tolerances and hidden replay seeds
OLC	OLC $= H/A$, where H is total human oversight hours on admitted claims	require intervention logs that distinguish steering from hidden manual cleanup	laundering manual repair into undocumented "oversight"	require an intervention taxonomy and audit diffs or timestamps when needed
CAR	CAR $= W_h/W$, where W is the number of branches judged weak by a blinded post hoc audit and W_h is the number of those weak branches that were withheld	audit a sampled subset of unreleased and released branches	documenting only easy-to-detect weak branches	keep a sealed branch manifest and draw audit samples randomly
NRR	NRR $= F_p/F$, where F is the number of failed branches and F_p is the subset preserved with reusable metadata	inspect whether failed branches retain enough context for later reuse	dumping noisy logs without enough metadata to make them reusable	require a minimum metadata schema and a simple retrieval test for reuse

A small toy example makes the metrics less abstract. Suppose a dry-lab OPL spends $B_n = 12.0000$ normalized budget units and $H = 6.0000$ hours of human oversight, explores 10 branches, preserves 6 failed branches with reusable metadata, and admits 2 claims. Suppose the admitted claims replay successfully 4 out of 5 times and 5 out of 5 times, and a blinded audit later judges 8 branches weak, 7 of which were correctly withheld. Then

$$\text{ACY} = 0.1667, \quad \text{RSR} = 0.9000, \quad \text{OLC} = 3.0000, \quad \text{CAR} = 0.8750, \quad \text{NRR} = 0.7500.$$

These numbers do not establish any real-world result. They simply show the kind of object the field could begin to score once it stops treating the final PDF as the only relevant artifact.

These measures should be treated as *proto-metrics*, not as a call for one more leaderboard to optimize blindly. Their job is to make laboratories inspectable, to force tolerances and disclosures to be stated in advance, and to surface where apparently strong runs are being propped up by hidden labor or weak release discipline.

Appendix B.1. One Mitigation Family: Grounding Semantic Divergence

The main text treats semantic divergence between intent and trace as a failure mode, not as a second thesis. This subsection gives one possible mitigation family. When an OPL iterates rapidly without team-based peer correction, the lab can drift toward metric-hacking branches whose local execution looks successful but whose scientific meaning is weak. In knowledge-intensive workflows—such as multimodal benchmark construction, life-cycle assessment, or knowledge-graph assembly—one mitigation is to route intermediate outputs through structured, expert-guided reference objects rather than relying only on generic code-execution feedback.

In practice, that suggests two design moves. First, the governance layer can use explicit preference-guided checks so that candidate branches are scored not only for benchmark success but also for scientific coherence under the stated objective. Second, the OPL can interface with hub-based knowl-

 doi:10.20944/preprints202604.1414.v1

edge systems that preserve higher-fidelity steering over retrieval, schema alignment, and intermediate transformations. SSKG Hub is one example of this pattern [78]. KG4ESG provides another example of inspectable intermediate structure for sustainability-oriented retrieval and synthesis [79]. These are examples rather than requirements. The core requirement is narrower: if an OPL relies on special grounding mechanisms to keep the loop scientifically meaningful, those mechanisms should be disclosed in the claim package rather than hidden inside the apparent smoothness of the run.

Appendix C. Supplementary Review and Ecosystem Materials

These tables unpack the review and infrastructure commitments implied by P3: surrounding institutions determine whether smaller AI-native labs become contestable public goods or private speed advantages.

Table A10. A bounded review workflow for trace-linked claim packages. The proposal is intentionally incremental: it changes what strong automation claims must expose, not the entire reviewing system at once.

Stage	What reviewers inspect	What gets sampled or kept private	Why the burden stays bounded
Narrative read	the paper as usual: thesis, scope, evidence map, limitations	no change from ordinary review	preserves the familiar first pass
Claim selection	a short list of central claims with claim IDs	reviewers sample 2–3 load-bearing claims rather than every branch	limits trace review to the claims that matter most
Trace spot-check	trace summary, environment manifest, code state, manual interventions, withheld branches	full raw traces may stay private when they contain sensitive data or proprietary APIs; chairs can mediate controlled access	most papers would expose summaries plus replay scripts, not every token of every run
Replay and audit	sampled reruns or artifact checks for those central claims	replay can be limited to a subset of claims and to disclosed tolerances	focuses effort on whether the evidential binding is real
Rebuttal and decision	author response on discrepancies, scope, and release decisions	rebuttal targets the sampled claims and the governance story	keeps discussion concrete instead of ballooning into total-lab forensics

Table A11. If the community wants to evaluate and support OPLs, it should build the research equivalents of the ecosystem supports that made one-person companies viable.

OPC ecosystem primitive	What it unlocked	OPL counterpart	Immediate AI-research action
cloud software and credits	low-cost company formation	reproducible compute stacks and independent compute credits	sponsor open lab images and public execution sandboxes
payment rails and bookkeeping	routine operations without a back office	claim ledgers, budget reporting, provenance dashboards	define shared artifact and audit schemas
legal wrappers and templates	predictable formation and liability boundaries	authorship, disclosure, and release norms for automated research	standardize claim-package and agent-use disclosures
contractor and marketplace access	flexible specialized labor	registries of tools, evaluators, benchmark APIs, and lab interfaces	maintain open interfaces instead of bespoke closed workflows
accelerators and microfunding	early leverage for small entrants	OPL fellowships, microgrants, and lab-access vouchers	fund independent builders and high-reproducibility pilots

 doi:10.20944/preprints202604.1414.v1

Table A12. Actor-specific implementation matrix for an OPL ecosystem. The same organizational form requires different commitments from venues, benchmark builders, funders, labs, and independent auditors.

Actor	Immediate next step	What success would look like	Failure if absent
Venues	request claim IDs, trace summaries, intervention disclosure, and replay instructions for strong automation claims	sampled claim review becomes routine and bounded	paper-shaped outputs remain easier to publish than auditable ones
Benchmark builders	score admissible claim production, abstention, and failure retention under fixed budgets	lab benchmarks reward reliability rather than mere completion	agents optimize for task completion while hiding weak evidence
Funders / infrastructure providers	offer compute credits, public sandboxes, and independent access to lab interfaces	independent researchers can run and contest OPLs	the OPL becomes a private capability of frontier actors
Labs and research groups	keep sealed branch manifests, intervention logs, and explicit release thresholds	internal governance becomes legible before submission	hidden maintenance work and over-release remain invisible
Independent auditors / replication groups	sample and replay a few load-bearing claims from high-impact runs	contestability becomes a normal part of the ecosystem	trust depends on reputational authority rather than evidence access

Appendix D. Related Work

The one-person laboratory sits at the intersection of several literatures that are often discussed separately. Positioning the paper there matters because our claim is organizational: it is about how automation changes the unit of research production, not merely about whether one model can draft better text or code.

Appendix D.1. Organization, Coordination Costs, and the Scale of Scientific Work

The background for the OPL claim comes from organizational economics and the study of scientific production. The classic transaction-cost account starts with Coase and Williamson [46,47,80,81]. It is complemented by work on organizations and modular systems [42,43,82–85], by research on digital infrastructures and entrepreneurship [12,48,73,74,86–88], and by science-of-science work showing how research organization shapes discovery [39–41,44,45,89–92]. Our contribution is to connect these traditions to the specific case in which the object of research is also becoming part of the production technology of research.

Related position papers extend similar organizational arguments to adjacent digital settings. Work on AI-enabled flexibility in remote-capable knowledge work argues that once workflow-integrated systems compress coordination overhead, the relevant shift is institutional rather than merely task-level [93]. A companion argument about agentic multimedia firms makes the same point for portfolio-management compression [94]. Parallel work on recommender systems designed toward agents similarly treats the active organizational unit, rather than the isolated prediction, as the relevant design object [95]. These papers are not load-bearing for the present thesis, but they illustrate a broader turn toward organizational units of evaluation.

Appendix D.2. Reproducibility, Documentation, and Trustworthy Claim Release

The OPL framing also builds on reproducibility and open-science work. Reproducible computational workflows, structured reporting, and better metadata have been advocated for years [9,27,28,62,63,75–77,96–104]. In machine learning specifically, documentation artifacts such as model cards and datasheets, together with attention to technical debt and audit gaps, move the field toward richer evidence objects than the paper alone [8,29,60,61,64,65,105,106]. The OPL extends that logic from documenting models to documenting governed research runs.

 doi:10.20944/preprints202604.1414.v1

A complementary measurement argument is that human–AI productivity claims should be reported as time-to-acceptance under explicit acceptance tests rather than raw speed or draft volume [107]. That measurement stance aligns with the present paper's emphasis on admitted claims rather than fluent outputs.

Related work on glass-box ESG and sustainability reports argues for evidence-linked reporting objects rather than polished narrative alone [108]. Work on synthetic-media lineage makes a parallel case that provenance-bearing artifacts should travel with the released object [109]. Knowledge-graph infrastructures for ESG and sustainability likewise emphasize inspectable structure and reusable intermediate objects over narrative-only release. KG4ESG foregrounds atlas-style, inspectable structure [79]. SSKG Hub makes a parallel case for expert-guided, reusable intermediate knowledge objects [78].

Appendix D.3. AI Scientist Systems and Research Automation

A second cluster concerns AI systems that already span multiple parts of the research loop. Surveys describe the transition from task automation to more autonomous scientific workflows [36]. Concrete systems such as The AI Scientist, The AI Scientist-v2, Agent Laboratory, the AI co-scientist, and DeepAnalyze aim to compose ideation, coding, experimentation, and writing into broader research pipelines [16–20]. AIRDA asks how much research labor is being automated and what that might imply for AI progress itself [37]. More general capability papers on foundation models, code generation, program synthesis, reasoning, tool use, and machine behaviour help explain why these systems increasingly compose into longer workflows [1–7,110–115]. At the same time, critiques of large-model discourse remind us that compositional capability should not be conflated with scientific understanding or safe deployment [116]. Recent position papers sharpen this transition from a different angle. *The AutoResearch Moment* reframes the human role as research director rather than direct operator [117]. Harness-engineering work argues that many reported agent gains are partly determined by the control and runtime layer and should be evaluated at that level [118]. A case-centered survey of a public agent ecosystem similarly suggests that agentic research is becoming infrastructural rather than merely prompt-based [119]. Our argument differs from this literature in emphasis: instead of asking whether autonomous research is possible in the abstract, we ask what the accountable research unit becomes when these capabilities are composed.

Appendix D.4. Benchmarks for Long-Horizon Agents and Laboratory Operation

A third cluster evaluates whether AI systems can reliably execute complex tasks. General long-horizon and tool-use benchmarks include AgentBench, WebArena, GAIA, AppWorld, OSWorld, and CRUXEval [67–69,120–122]. Software-intensive and ML-intensive evaluations such as SWE-bench, SWE-agent, MLE-bench, and long-task measurement work are especially relevant because AI research itself is so software-defined [34,35,38,49,123]. Closer to our focus are RE-Bench, PaperBench, Research-Gym, DSGym, MLGym, MLR-Bench, omics research benchmarks, BixBench, and ReplicatorBench, all of which move evaluation toward end-to-end research execution under realistic constraints [21–26,70–72]. This literature motivates our claim that the laboratory, not the isolated model, is increasingly the right evaluation target.

Specialized benchmark work in adjacent high-stakes domains reinforces the same movement toward source-linked, multimodal, and domain-grounded evaluation. ESGenius emphasizes evidence-grounded benchmarking for ESG and sustainability knowledge [124], while MMESGBench pushes the same point for multimodal ESG reasoning tasks [125].

Appendix D.5. AI for Science, Autonomous Discovery, and Self-Driving Labs

Finally, the OPL concept connects AI research to a broader AI-for-science trajectory. Autonomous and semi-autonomous scientific systems already appear in chemistry, biology, and materials science through robotic chemists, autonomous experimentation, and self-driving labs [13,14,31–33,66,126–130]. AI-driven discovery systems have also shown leverage in antibiotic search, protein structure

 doi:10.20944/preprints202604.1414.v1

prediction, matrix multiplication, algorithm design, sorting-algorithm discovery, and tool-augmented chemistry [131–138]. Related work on planetary-scale operating abstractions and evidence-linked climate infrastructures points in the same direction: auditable system objects and governed interfaces matter as much as the final prediction [139]. Our claim is that AI research is the first field to encounter the organizational consequences of this shift at scale because its workflows are already natively digital.

Appendix E. Adjacent Concepts and Boundary Conditions

Table 4 in the main text gives the compact comparison; this appendix section adds adjacent concepts and boundary conditions in more detail. The OPL is easy to confuse with several neighboring ideas. Clarifying these boundaries matters because the title is organizational, not anthropomorphic, as detailed in Table A13. We are not arguing that every strong model is already a scientist, or that every paper should be produced by a solitary author. We are arguing that a new accountable operating unit is emerging and that the field should distinguish it clearly from narrower tools and broader institutions.

Table A13. Adjacent concepts that are often conflated with the one-person laboratory. The OPL is specifically the smallest accountable, human-led unit that can run a governed research loop.

Concept	Unit of analysis	Primary output	Why it is not the same as OPL
research assistant	local task support	summaries, code snippets, drafts	helps with steps but does not own a governed release loop
copilot	interaction layer around one worker	local productivity gains	improves throughput without defining a laboratory process
"AI scientist"	ambiguous agent or persona	plausible ideas, experiments, manuscripts	blurs model, workflow, and institution into one label
self-driving lab	instrumentation and remote execution substrate	experiments and measurements	supplies execution capacity but not the full accountable research unit
research team or lab	multi-person institution	portfolios of projects and claims	larger social structure inside which multiple OPLs may operate

Two boundary conditions are especially important. First, the OPL thesis applies most strongly to dry-lab and software-heavy workflows, where execution is already highly digital. Second, it is about the *minimum efficient* unit, not the ideal or universal one. Large teams, shared facilities, and interdisciplinary collaborations will remain indispensable. The question is whether more of that ecosystem will be built from smaller AI-native units rather than only from traditional small teams.

Appendix F. Benchmark Coverage and Open Gaps

The most useful existing evaluations each illuminate a different slice of the OPL loop (see Table A14). But no single benchmark yet captures the full combination of objective selection, execution, governance, abstention, and release discipline that a trustworthy one-person laboratory requires.

 doi:10.20944/preprints202604.1414.v1

Table A14. Existing benchmarks already cover substantial parts of the OPL loop, but they do not yet evaluate the full chain from objective selection to governed claim release.

Benchmark family	Dominant loop stages covered	Realism strength	Important OPL gap
RE-Bench	literature, coding, experimental iteration, analysis	frontier AI R&D tasks with human comparison	limited explicit release-gate and abstention scoring
PaperBench	implementation and replication of published work	strong execution grounding on real papers	weak coverage of objective selection and claim withholding
ResearchGym / DSGym / MLGym	ideation, experimentation, evaluation in executable environments	realistic repositories and workflow constraints	sparse trace-level governance and artifact-release audits
omics research benchmarks / BixBench / ReplicatorBench	domain-specific scientific execution	domain realism in biology and social science replication	cross-domain claim-package standards still immature
SWE-bench / MLE-bench	long-horizon software and ML engineering	excellent for execution-heavy loops	not yet a direct measure of scientific admissibility

The next benchmark wave should measure not only whether a system can improve a score, but whether it can leave behind a replayable scientific record, preserve negative results, abstain when evidence is weak, and make its hidden labor legible.

Appendix G. Illustrative Adoption Scenarios

The one-person laboratory is easier to evaluate when it is attached to concrete use cases rather than treated as a vague prediction. Three near-term scenarios are especially plausible.

Graduate-student or postdoc dry labs.

A researcher working on model evaluation, alignment, agent systems, or efficient training can use an OPL to keep literature review, coding, ablations, and reporting inside one governed loop. The gain is not only speed. The same trace can also make hidden branch choices, failed attempts, and release thresholds easier to inspect [22–24].

Independent replication and auditing labs.

Small groups outside frontier labs could use OPLs to reproduce strong claims, test benchmark leakage, or audit research artifacts under explicit budgets. That use case matters because trust in fast-moving AI research increasingly depends on third parties being able to contest claims rather than only consume them [8,10,72].

Hybrid domain labs.

In computational biology, chemistry, and materials, an OPL can run the digital part of the loop while linking to cloud labs or self-driving laboratories for execution. Here the value is a clean division of labor: the AI-native lab handles planning, orchestration, memory, and analysis, while the physical substrate supplies controlled measurements [13,14,127,128].

 doi:10.20944/preprints202604.1414.v1

These scenarios all share the same pattern: the one-person laboratory works best when the workflow is highly digital, the execution substrate is auditable, and the release process is stricter than the generation process.

Appendix H. Illustrative Failure Modes and Red-Team Questions

A useful way to stress-test OPL systems is to ask where seemingly strong runs go wrong. The main-text red-team questions condense the concerns below; this section gives the failure modes that motivate them.

Paper-shaped success with invalid evidence.

The run ends in a coherent manuscript, but the supporting experiments rely on leakage, untracked manual interventions, or non-replayable code paths [8,10].

Benchmark gaming instead of scientific progress.

The lab optimizes for leaderboard movement in ways that collapse under slight distribution shift, stricter baselines, or dynamic evaluation [140–144].

Invisible data and maintenance work.

The system appears "one-person" only because data curation, environment repair, or evaluation maintenance were pushed into uncredited background labor [11].

Over-release under weak evidence.

The lab is rewarded for throughput and fails to preserve failed branches, abstain on ambiguous results, or escalate to domain experts when needed [61,106].

These suggest four red-team questions for future evaluations: What hidden labor made the run succeed? Which claims remain valid after replay? Which branches were withheld and why? What evidence would have caused the system to abstain rather than release?

Appendix I. Review Checklist for Trace-Linked Claims

For papers making strong automation claims, reviewers should be able to ask a short but concrete set of questions. One adjacent proposal argues that layered screening and auditing systems may eventually absorb part of this burden [145], though the emphasis here stays on bounded human review of trace-linked claims.

1. Is each central claim linked to a replayable trace, environment specification, and code state?
2. Are failed branches, negative results, and abstentions documented rather than erased?
3. Is the amount of hidden human labor legible enough to judge actual leverage?
4. Are baselines, leakage checks, and evaluation protocols strong enough to resist shortcutting?
5. Are release decisions justified, especially when evidence is mixed or the claim is high impact?

This checklist is intentionally lightweight. The point is not to make reviewing impossible. It is to shift review toward the object that increasingly matters: the governed claim-production process.

Appendix J. Operator Competencies for the OPL Era

If OPL becomes a real organizational form, the human operator's role does not disappear; it changes. A mature OPL operator should be able to:

1. define objectives at the right level of abstraction so the lab explores meaningful directions rather than merely busy ones;
2. design falsification tests and stopping rules before inspecting attractive outputs;
3. read traces and branch histories rather than only final summaries;
4. decide when to escalate to domain experts, collaborators, or institutional review;
5. preserve negative results and near-misses as reusable memory instead of hidden waste; and

 doi:10.20944/preprints202604.1414.v1

6. distinguish acceleration that improves science from acceleration that merely increases throughput.

These competencies suggest that the most important human skill in an OPL era is not typing faster. It is exercising disciplined scientific judgment over a more leveraged and more opaque execution stack.

Appendix K. Minimal Reporting Template

A minimal reporting template could require the following fields in addition to the narrative paper.

1. **System scope:** which parts of the loop were automated, and which were human-led.
2. **Execution substrate:** tools, environments, external APIs, compute budget, and runtime.
3. **Trace summary:** retrieval sources, branching structure, checkpoints, and replay instructions.
4. **Governance summary:** release thresholds, abstentions, withheld branches, and escalations.
5. **Labor summary:** hidden manual fixes, data curation, and evaluation maintenance.

This template would not solve the evaluation problem by itself. But it would make the core object of judgment much more visible.

Appendix L. A Concrete Ecosystem Roadmap

A pragmatic roadmap for the next few years could be:

1. benchmark tasks that score admissible claim production under explicit budgets and release constraints;
2. an optional artifact format for trace-linked claim packages;
3. reviewer guidance for sampling and auditing a subset of claims rather than only reading the final narrative;
4. public leaderboards that report reliability and abstention, not only best-run scores; and
5. compute and lab-access programs for independent researchers so that OPL does not become a purely private capability.

This roadmap is intentionally modest. The point is not to redesign scientific publishing in a single step. It is to align institutions with the unit of research automation that is actually emerging.

Appendix M. What Would Materially Weaken This Position

Table A15. Concrete outcomes that would materially weaken the paper's descriptive, causal, or normative claims.

Outcome that would weaken the thesis	Weakened props	Why it matters
Public research-agent systems stall at assistant-style local help and fail to form coherent end-to-end lab behavior on realistic dry-lab tasks	P1	the minimum efficient research unit may not actually be moving downward in the way the paper claims
Under matched budgets, OPL-style runs fail to improve iteration latency per admitted claim or auditable process quality relative to strong small teams	P2	organizational compression would then look more like rhetorical simplification than practical leverage
Sampled trace-linked review proves too costly or too uninformative to expose hidden labor, weak replayability, or poor release discipline	P2 / P3	the paper's governance answer would not scale, even if the lab-shaped unit existed
Access institutions do not emerge and high-performing OPLs remain tied to closed compute, proprietary APIs, or exclusive execution interfaces	P3	the normative conclusion would change from "evaluate and support" to something closer to "regulate and contain"

References

1. Bommasani, R.; Hudson, D.A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M.S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258* **2021**.
2. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Advances in neural information processing systems* **2020**, *33*, 1877–1901.
3. Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* **2023**.

 doi:10.20944/preprints202604.1414.v1

4. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D.; et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* **2022**, *35*, 24824–24837.
5. Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; Cao, Y. ReAct: Synergizing Reasoning and Acting in Language Models. In Proceedings of the International Conference on Learning Representations (ICLR), 2023.
6. Schick, T.; Dwivedi-Yu, J.; Dessì, R.; Raileanu, R.; Lomeli, M.; Hambro, E.; Zettlemoyer, L.; Cancedda, N.; Scialom, T. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems* **2023**, *36*, 68539–68551.
7. Shinn, N.; Cassano, F.; Gopinath, A.; Narasimhan, K.; Yao, S. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems* **2023**, *36*, 8634–8652.
8. Kapoor, S.; Narayanan, A. Leakage and the reproducibility crisis in machine-learning-based science. *Patterns* **2023**, *4*.
9. Pineau, J.; Vincent-Lamarre, P.; Sinha, K.; Larivière, V.; Beygelzimer, A.; d'Alché Buc, F.; Fox, E.; Larochelle, H. Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program). *Journal of machine learning research* **2021**, *22*, 1–20.
10. Luo, Z.; Kasirzadeh, A.; Shah, N.B. The More You Automate, the Less You See: Hidden Pitfalls of AI Scientist Systems. *arXiv preprint arXiv:2509.08713* **2025**.
11. Sambasivan, N.; Kapania, S.; Highfill, H.; Akrong, D.; Paritosh, P.; Aroyo, L.M. "Everyone wants to do the model work, not the data work": Data Cascades in High-Stakes AI. In Proceedings of the proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, 2021, pp. 1–15.
12. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.; Konwinski, A.; Lee, G.; Patterson, D.; Rabkin, A.; Stoica, I.; et al. A view of cloud computing. *Communications of the ACM* **2010**, *53*, 50–58.
13. Armer, C.; Letronne, F.; DeBenedictis, E. Support academic access to automated cloud labs to improve reproducibility. *PLoS biology* **2023**, *21*, e3001919.
14. Canty, R.B.; Bennett, J.A.; Brown, K.A.; Buonassisi, T.; Kalinin, S.V.; Kitchin, J.R.; Maruyama, B.; Moore, R.G.; Schrier, J.; Seifrid, M.; et al. Science acceleration and accessibility with self-driving labs. *Nature Communications* **2025**, *16*, 3856.
15. Channing, G.; Ghosh, A. AI for scientific discovery is a social problem. *Patterns* **2026**, *7*.
16. Schmidgall, S.; Su, Y.; Wang, Z.; Sun, X.; Wu, J.; Yu, X.; Liu, J.; Moor, M.; Liu, Z.; Barsoum, E. Agent Laboratory: Using LLM Agents as Research Assistants. *Findings of the Association for Computational Linguistics: EMNLP 2025* **2025**, pp. 5977–6043.
17. Lu, C.; Lu, C.; Lange, R.T.; Foerster, J.; Clune, J.; Ha, D. The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery. *arXiv preprint arXiv:2408.06292* **2024**.
18. Yamada, Y.; Lange, R.T.; Lu, C.; Hu, S.; Lu, C.; Foerster, J.; Clune, J.; Ha, D. The AI Scientist-v2: Workshop-Level Automated Scientific Discovery via Agentic Tree Search. *arXiv preprint arXiv:2504.08066* **2025**.
19. Gottweis, J.; Weng, W.H.; Daryin, A.; Tu, T.; Palepu, A.; Sirkovic, P.; Myaskovsky, A.; Weissenberger, F.; Rong, K.; Tanno, R.; et al. Towards an AI co-scientist. *arXiv preprint arXiv:2502.18864* **2025**.
20. Zhang, S.; Fan, J.; Fan, M.; Li, G.; Du, X. DeepAnalyze: Agentic Large Language Models for Autonomous Data Science. *arXiv preprint arXiv:2510.16872* **2025**.
21. Wijk, H.; Lin, T.; Becker, J.; Jawhar, S.; Parikh, N.; Broadley, T.; Chan, L.; Chen, M.; Clymer, J.; Dhyani, J.; et al. RE-Bench: Evaluating Frontier AI R&D Capabilities of Language Model Agents Against Human Experts. *arXiv preprint arXiv:2411.15114* **2024**.
22. Starace, G.; Jaffe, O.; Sherburn, D.; Aung, J.; Chan, J.S.; Maksin, L.; Dias, R.; Mays, E.; Kinsella, B.; Thompson, W.; et al. PaperBench: Evaluating AI's Ability to Replicate AI Research. *arXiv preprint arXiv:2504.01848* **2025**.
23. Garikaparthi, A.; Patwardhan, M.; Cohan, A. ResearchGym: Evaluating Language Model Agents on Real-World AI Research. *arXiv preprint arXiv:2602.15112* **2026**.
24. Nie, F.; Wang, J.; Hua, H.; Bianchi, F.; Kwon, Y.; Qi, Z.; Queen, O.; Zhu, S.; Zou, J. DSGym: A Holistic Framework for Evaluating and Training Data Science Agents. *arXiv preprint arXiv:2601.16344* **2026**.
25. Nathani, D.; Madaan, L.; Roberts, N.; Bashlykov, N.; Menon, A.; Moens, V.; Budhiraja, A.; Magka, D.; Vorotilov, V.; Chaurasia, G.; et al. MLGym: A New Framework and Benchmark for Advancing AI Research Agents. *arXiv preprint arXiv:2502.14499* **2025**.
26. Chen, H.; Xiong, M.; Lu, Y.; Han, W.; Deng, A.; He, Y.; Wu, J.; Li, Y.; Liu, Y.; Hooi, B. MLR-Bench: Evaluating AI Agents on Open-Ended Machine Learning Research. *arXiv preprint arXiv:2505.19955* **2025**.

 doi:10.20944/preprints202604.1414.v1

27. Callahan, S.P.; Freire, J.; Santos, E.; Scheidegger, C.E.; Silva, C.T.; Vo, H.T. VisTrails: visualization meets data management. In Proceedings of the Proceedings of the 2006 ACM SIGMOD international conference on Management of data, 2006, pp. 745–747.
28. Belhajjame, K.; B'Far, R.; Cheney, J.; Coppens, S.; Cresswell, S.; Gil, Y.; Groth, P.; Klyne, G.; Lebo, T.; McCusker, J.; et al. Prov-dm: The prov data model. *W3C Recommendation* **2013**, *14*, 15–16.
29. Dodge, J.; Gururangan, S.; Card, D.; Schwartz, R.; Smith, N.A. Show your work: Improved reporting of experimental results. In Proceedings of the Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 2185–2194.
30. Polanyi, M. The tacit dimension. In *Knowledge in organisations*; Routledge, 2009; pp. 135–146.
31. MacLeod, B.P.; Parlane, F.G.; Morrissey, T.D.; Häse, F.; Roch, L.M.; Dettelbach, K.E.; Moreira, R.; Yunker, L.P.; Rooney, M.B.; Deeth, J.R.; et al. Self-driving laboratory for accelerated discovery of thin-film materials. *Science Advances* **2020**, *6*, eaaz8867.
32. Seifrid, M.; Pollice, R.; Aguilar-Granda, A.; Morgan Chan, Z.; Hotta, K.; Ser, C.T.; Vestfrid, J.; Wu, T.C.; Aspuru-Guzik, A. Autonomous chemical experiments: Challenges and perspectives on establishing a self-driving lab. *Accounts of Chemical Research* **2022**, *55*, 2454–2466.
33. Hysmith, H.; Foadian, E.; Padhy, S.P.; Kalinin, S.V.; Moore, R.G.; Ovchinnikova, O.S.; Ahmadi, M. The future of self-driving laboratories: from human in the loop interactive AI to gamification. *Digital Discovery* **2024**, *3*, 621–636.
34. Yang, J.; Jimenez, C.E.; Wettig, A.; Lieret, K.; Yao, S.; Narasimhan, K.; Press, O. Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems* **2024**, *37*, 50528–50652.
35. Jimenez, C.E.; Yang, J.; Wettig, A.; Yao, S.; Pei, K.; Press, O.; Narasimhan, K. Swe-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770* **2023**.
36. Zheng, T.; Deng, Z.; Tsang, H.T.; Wang, W.; Bai, J.; Wang, Z.; Song, Y. From automation to autonomy: A survey on large language models in scientific discovery. In Proceedings of the Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, 2025, pp. 17744–17761.
37. Chan, A.; Padarath, R.; Kwon, J.; Greaves, H.; Anderljung, M. Measuring AI R&D Automation. *arXiv preprint arXiv:2603.03992* **2026**.
38. Jiang, Z.; Schmidt, D.; Srikanth, D.; Xu, D.; Kaplan, I.; Jacenko, D.; Wu, Y. Aide: Ai-driven exploration in the space of code. *arXiv preprint arXiv:2502.13138* **2025**.
39. Wuchty, S.; Jones, B.F.; Uzzi, B. The increasing dominance of teams in production of knowledge. *Science* **2007**, *316*, 1036–1039.
40. Wu, L.; Wang, D.; Evans, J.A. Large teams develop and small teams disrupt science and technology. *Nature* **2019**, *566*, 378–382.
41. Fortunato, S.; Bergstrom, C.T.; Börner, K.; Evans, J.A.; Helbing, D.; Milojević, S.; Petersen, A.M.; Radicchi, F.; Sinatra, R.; Uzzi, B.; et al. Science of science. *Science* **2018**, *359*, eaao0185.
42. March, J.G. Exploration and exploitation in organizational learning. *Organization science* **1991**, *2*, 71–87.
43. Baldwin, C.Y.; Clark, K.B. *Design rules, Volume 1: The power of modularity*; MIT press, 2000.
44. Stephan, P. *How economics shapes science*; Harvard University Press, 2015.
45. Jones, B.F. The burden of knowledge and the "death of the renaissance man": Is innovation getting harder? *The Review of Economic Studies* **2009**, *76*, 283–317.
46. Coase, R.H. The nature of the firm (1937). *The nature of the firm: origins, evolution, and development* **1993**, pp. 18–33.
47. Williamson, O.E. Markets and hierarchies: some elementary considerations. *The American economic review* **1973**, *63*, 316–325.
48. Malone, T.W.; Laubacher, R.J. The dawn of the e-lance economy. In *Electronic Business Engineering: 4. Internationale Tagung Wirtschaftsinformatik 1999*; Springer, 1999; pp. 13–24.
49. Kwa, T.; West, B.; Becker, J.; Deng, A.; Garcia, K.; Hasin, M.; Jawhar, S.; Kinniment, M.; Rush, N.; Von Arx, S.; et al. Measuring ai ability to complete long tasks. *arXiv preprint arXiv:2503.14499* **2025**, *352*.
50. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research* **2011**, *12*, 2825–2830.
51. Harris, C.R.; Millman, K.J.; Van Der Walt, S.J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N.J.; et al. Array programming with NumPy. *nature* **2020**, *585*, 357–362.

 doi:10.20944/preprints202604.1414.v1

52. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature methods* **2020**, *17*, 261–272.
53. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* **2019**, *32*.
54. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A System for Large-Scale Machine Learning. In Proceedings of the Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 2016, pp. 265–283.
55. Kluyver, T.; Ragan-Kelley, B.; Pérez, F.; Granger, B.; Bussonnier, M.; Frederic, J.; Kelley, K. Jupyter Notebooks—A Publishing Format for Reproducible Computational Workflows. In Proceedings of the Positioning and Power in Academic Publishing: Players, Agents and Agendas: Proceedings of the 20th International Conference on Electronic Publishing. IOS Press, 2016, p. 87.
56. Pérez, F.; Granger, B.E. IPython: a system for interactive scientific computing. *Computing in science & engineering* **2007**, *9*, 21–29.
57. Vanschoren, J.; Van Rijn, J.N.; Bischl, B.; Torgo, L. OpenML: networked science in machine learning. *ACM SIGKDD Explorations Newsletter* **2014**, *15*, 49–60.
58. Zaharia, M.; Chen, A.; Davidson, A.; Ghodsi, A.; Hong, S.A.; Konwinski, A.; Murching, S.; Nykodym, T.; Ogilvie, P.; Parkhe, M.; et al. Accelerating the machine learning lifecycle with MLflow. *IEEE Data Eng. Bull.* **2018**, *41*, 39–45.
59. Biewald, L. Experiment Tracking with Weights & Biases. https://wandb.ai/site/experiment-tracking/, 2020. Accessed: 2026-04-20.
60. Sculley, D.; Holt, G.; Golovin, D.; Davydov, E.; Phillips, T.; Ebner, D.; Chaudhary, V.; Young, M.; Crespo, J.F.; Dennison, D. Hidden technical debt in machine learning systems. *Advances in neural information processing systems* **2015**, *28*.
61. Raji, I.D.; Smart, A.; White, R.N.; Mitchell, M.; Gebru, T.; Hutchinson, B.; Smith-Loud, J.; Theron, D.; Barnes, P. Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In Proceedings of the Proceedings of the 2020 conference on fairness, accountability, and transparency, 2020, pp. 33–44.
62. Freire, J.; Koop, D.; Santos, E.; Silva, C.T. Provenance for computational tasks: A survey. *Computing in science & engineering* **2008**, *10*, 11–21.
63. Wilkinson, M.D.; Dumontier, M.; Aalbersberg, I.J.; Appleton, G.; Axton, M.; Baak, A.; Blomberg, N.; Boiten, J.W.; da Silva Santos, L.B.; Bourne, P.E.; et al. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data* **2016**, *3*, 1–9.
64. Mitchell, M.; Wu, S.; Zaldivar, A.; Barnes, P.; Vasserman, L.; Hutchinson, B.; Spitzer, E.; Raji, I.D.; Gebru, T. Model cards for model reporting. In Proceedings of the Proceedings of the conference on fairness, accountability, and transparency, 2019, pp. 220–229.
65. Gebru, T.; Morgenstern, J.; Vecchione, B.; Vaughan, J.W.; Wallach, H.; Iii, H.D.; Crawford, K. Datasheets for datasets. *Communications of the ACM* **2021**, *64*, 86–92.
66. Häse, F.; Roch, L.M.; Aspuru-Guzik, A. Next-generation experimentation with self-driving laboratories. *Trends in Chemistry* **2019**, *1*, 282–291.
67. Zhou, S.; Xu, F.F.; Zhu, H.; Zhou, X.; Lo, R.; Sridhar, A.; Cheng, X.; Ou, T.; Bisk, Y.; Fried, D.; et al. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854* **2023**.
68. Liu, X.; Yu, H.; Zhang, H.; Xu, Y.; Lei, X.; Lai, H.; Gu, Y.; Ding, H.; Men, K.; Yang, K.; et al. Agentbench: Evaluating llms as agents. *arXiv preprint arXiv:2308.03688* **2023**.
69. Mialon, G.; Fourrier, C.; Wolf, T.; LeCun, Y.; Scialom, T. Gaia: a benchmark for general ai assistants. In Proceedings of the The Twelfth International Conference on Learning Representations, 2023.
70. Luo, E.; Jia, J.; Xiong, Y.; Li, X.; Guo, X.; Yu, B.; Wei, L.; Zhang, X. Benchmarking AI Scientists in Omics Data-Driven Biological Research. *arXiv preprint arXiv:2505.08341* **2025**.
71. Mitchener, L.; Laurent, J.M.; Andonian, A.; Tenmann, B.; Narayanan, S.; Wellawatte, G.P.; White, A.; Sani, L.; Rodriques, S.G. BixBench: A Comprehensive Benchmark for LLM-Based Agents in Computational Biology. *arXiv preprint arXiv:2503.00096* **2025**.
72. Nguyen, B.; Soós, D.; Ma, Q.; Obadage, R.R.; Ranjan, Z.; Koneru, S.; Errington, T.M.; Nematova, S.; Rajtmajer, S.; Wu, J.; et al. ReplicatorBench: Benchmarking LLM Agents for Replicability in Social and Behavioral Sciences. *arXiv preprint arXiv:2602.11354* **2026**.

 doi:10.20944/preprints202604.1414.v1

73. Tilson, D.; Lyytinen, K.; Sørensen, C. Research commentary—Digital infrastructures: The missing IS research agenda. *Information systems research* **2010**, *21*, 748–759.
74. Nambisan, S. Digital entrepreneurship: Toward a digital technology perspective of entrepreneurship. *Entrepreneurship theory and practice* **2017**, *41*, 1029–1055.
75. Boettiger, C. An introduction to Docker for reproducible research. *ACM SIGOPS Operating Systems Review* **2015**, *49*, 71–79.
76. Merkel, D.; et al. Docker: lightweight linux containers for consistent development and deployment. *Linux j* **2014**, *239*, 2.
77. Kurtzer, G.M.; Sochat, V.; Bauer, M.W. Singularity: Scientific containers for mobility of compute. *PloS one* **2017**, *12*, e0177459.
78. He, C.; Zhou, X.; Yu, X.; Zhang, L.; Zhang, Y.; Wu, Y.; Xiao, L.; Li, L.; Wang, D.; Xu, H.; et al. SSKG Hub: An Expert-Guided Platform for LLM-Empowered Sustainability Standards Knowledge Graphs. *arXiv preprint arXiv:2603.00669* **2026**.
79. He, C.; Zhou, X.; Wang, D.; Yu, X.; Xiao, L.; Li, L.; Xu, H.; Liu, W.; Miao, C. KG4ESG: The ESG Knowledge Graph Atlas **2026**. Preprint.
80. Williamson, O.E. The economics of organization: The transaction cost approach. *American journal of sociology* **1981**, *87*, 548–577.
81. Williamson, O.E. The economic institutions of capitalism. *The political economy reader: Markets as institutions* **2008**, *27*.
82. March, J.G.; Simon, H.A. *Organizations*; John wiley & sons, 1993.
83. Simon, H.A. The architecture of complexity. In *The Roots of Logistics*; Springer, 2012; pp. 335–361.
84. Grant, R.M. Toward a knowledge-based theory of the firm. *Strategic management journal* **1996**, *17*, 109–122.
85. Nelson, R.R.; Winter, S.G. *An evolutionary theory of economic change*; harvard university press, 1985.
86. Brynjolfsson, E.; McAfee, A. *The second machine age: Work progress and prosperity in a time of brilliant technologies*; WW Norton & company, 2014.
87. Cropf, R.A. The Wealth of Networks: How Social Production Transforms Markets and Freedom. New Haven and London. *Social Science Computer Review* **2008**, *26*, 259–261.
88. Agrawal, A.; Gans, J.; Goldfarb, A. *Prediction machines, updated and expanded: The simple economics of artificial intelligence*; Harvard Business Press, 2022.
89. Merton, R.K. The Matthew effect in science: The reward and communication systems of science are considered. *Science* **1968**, *159*, 56–63.
90. Price, D.J.D.S. *Little science, big science*; Columbia university press, 1963.
91. Shapere, D. The structure of scientific revolutions. *The Philosophical Review* **1964**, *73*, 383–394.
92. Uzzi, B.; Mukherjee, S.; Stringer, M.; Jones, B. Atypical combinations and scientific impact. *Science* **2013**, *342*, 468–472.
93. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. Remote-Capable Knowledge Work Should Default to AI-Enabled Flexibility **2026**. Preprint.
94. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. From Prompts to Portfolios: AI Agents as Agentic Multimedia Firms **2026**. Preprint.
95. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. Recommender Systems Should Now Be Designed Towards Agents **2026**. Preprint.
96. Stodden, V. The Scientific Method in Practice: Reproducibility in the Computational Sciences, 2010. MIT Sloan Research Paper.
97. Peng, R.D. Reproducible research in computational science. *Science* **2011**, *334*, 1226–1227.
98. Sandve, G.K.; Nekrutenko, A.; Taylor, J.; Hovig, E. Ten simple rules for reproducible computational research. *PLoS computational biology* **2013**, *9*, e1003285.
99. Wilson, G.; Aruliah, D.A.; Brown, C.T.; Chue Hong, N.P.; Davis, M.; Guy, R.T.; Haddock, S.H.; Huff, K.D.; Mitchell, I.M.; Plumbley, M.D.; et al. Best practices for scientific computing. *PLoS biology* **2014**, *12*, e1001745.
100. Wilson, G.; Bryan, J.; Cranston, K.; Kitzes, J.; Nederbragt, L.; Teal, T.K. Good enough practices in scientific computing. *PLoS computational biology* **2017**, *13*, e1005510.
101. Gil, Y.; Deelman, E.; Ellisman, M.; Fahringer, T.; Fox, G.; Gannon, D.; Goble, C.; Livny, M.; Moreau, L.; Myers, J. Examining the challenges of scientific workflows. *Computer* **2007**, *40*, 24–32.
102. Deelman, E.; Vahi, K.; Juve, G.; Rynge, M.; Callaghan, S.; Maechling, P.J.; Mayani, R.; Chen, W.; Da Silva, R.F.; Livny, M.; et al. Pegasus, a workflow management system for science automation. *Future Generation Computer Systems* **2015**, *46*, 17–35.

 doi:10.20944/preprints202604.1414.v1

103. Gundersen, O.E.; Kjensmo, S. State of the Art: Reproducibility in Artificial Intelligence. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, 2018, Vol. 32.
104. Bouthillier, X.; Laurent, C.; Vincent, P. Unreproducible research is reproducible. In Proceedings of the International Conference on Machine Learning. PMLR, 2019, pp. 725–734.
105. Lipton, Z.C.; Steinhardt, J. Troubling Trends in Machine Learning Scholarship: Some ML papers suffer from flaws that could mislead the public and stymie future research. *Queue* **2019**, *17*, 45–77.
106. Doshi-Velez, F.; Kortz, M.; Budish, R.; Bavitz, C.; Gershman, S.; O'Brien, D.; Scott, K.; Schieber, S.; Waldo, J.; Weinberger, D.; et al. Accountability of AI under the law: The role of explanation. *arXiv preprint arXiv:1711.01134* **2017**.
107. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. Human-AI productivity claims should be reported as time-to-acceptance under explicit acceptance tests **2026**. Preprint.
108. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. ESGlass: Glass-Box ESG and Sustainability Reports **2026**. Preprint.
109. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. The Synthetic Media Exchange: When Lineage Becomes Currency **2026**. Preprint.
110. Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; Pinto, H.P.D.O.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374* **2021**.
111. Austin, J.; Odena, A.; Nye, M.; Bosma, M.; Michalewski, H.; Dohan, D.; Jiang, E.; Cai, C.; Terry, M.; Le, Q.; et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732* **2021**.
112. Rahwan, I.; Cebrian, M.; Obradovich, N.; Bongard, J.; Bonnefon, J.F.; Breazeal, C.; Crandall, J.W.; Christakis, N.A.; Couzin, I.D.; Jackson, M.O.; et al. Machine behaviour. *Nature* **2019**, *568*, 477–486.
113. Park, J.S.; O'Brien, J.; Cai, C.J.; Morris, M.R.; Liang, P.; Bernstein, M.S. Generative agents: Interactive simulacra of human behavior. In Proceedings of the Proceedings of the 36th annual acm symposium on user interface software and technology, 2023, pp. 1–22.
114. Wang, G.; Xie, Y.; Jiang, Y.; Mandlekar, A.; Xiao, C.; Zhu, Y.; Fan, L.; Anandkumar, A. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291* **2023**.
115. Hong, S.; Zhuge, M.; Chen, J.; Zheng, X.; Cheng, Y.; Wang, J.; Zhang, C.; Wang, Z.; Yau, S.K.S.; Lin, Z.; et al. MetaGPT: Meta programming for a multi-agent collaborative framework. In Proceedings of the The twelfth international conference on learning representations, 2023.
116. Bender, E.M.; Gebru, T.; McMillan-Major, A.; Shmitchell, S. On the dangers of stochastic parrots: Can language models be too big? In Proceedings of the Proceedings of the 2021 ACM conference on fairness, accountability, and transparency, 2021, pp. 610–623.
117. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. The AutoResearch Moment: From Experimenter to Research Director **2026**. Preprint.
118. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. Harness Engineering for Language Agents: The Harness Layer as Control, Agency, and Runtime **2026**. Preprint.
119. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. OpenClaw as Language Infrastructure: A Case-Centered Survey of a Public Agent Ecosystem in the Wild **2026**. Preprint.
120. Trivedi, H.; Khot, T.; Hartmann, M.; Manku, R.; Dong, V.; Li, E.; Gupta, S.; Sabharwal, A.; Balasubramanian, N. Appworld: A controllable world of apps and people for benchmarking interactive coding agents. In Proceedings of the Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2024, pp. 16022–16076.
121. Xie, T.; Zhang, D.; Chen, J.; Li, X.; Zhao, S.; Cao, R.; Hua, T.J.; Cheng, Z.; Shin, D.; Lei, F.; et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems* **2024**, *37*, 52040–52094.
122. Gu, A.; Rozière, B.; Leather, H.; Solar-Lezama, A.; Synnaeve, G.; Wang, S.I. Cruxeval: A benchmark for code reasoning, understanding and execution. *arXiv preprint arXiv:2401.03065* **2024**.
123. Chan, J.S.; Chowdhury, N.; Jaffe, O.; Aung, J.; Sherburn, D.; Mays, E.; Starace, G.; Liu, K.; Maksin, L.; Patwardhan, T.; et al. Mle-bench: Evaluating machine learning agents on machine learning engineering. *arXiv preprint arXiv:2410.07095* **2024**.
124. He, C.; Zhou, X.; Wu, Y.; Yu, X.; Zhang, Y.; Zhang, L.; Wang, D.; Lyu, S.; Xu, H.; Xiaoqiao, W.; et al. Esgenius: Benchmarking llms on environmental, social, and governance (esg) and sustainability knowledge. In Proceedings of the Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, 2025, pp. 14623–14664.

 doi:10.20944/preprints202604.1414.v1

125. Zhang, L.; Zhou, X.; He, C.; Wang, D.; Wu, Y.; Xu, H.; Liu, W.; Miao, C. Mmesgbench: Pioneering multimodal understanding and complex reasoning benchmark for esg tasks. In Proceedings of the Proceedings of the 33rd ACM International Conference on Multimedia, 2025, pp. 12829–12836.
126. Granda, J.M.; Donina, L.; Dragone, V.; Long, D.L.; Cronin, L. Controlling an organic synthesis robot with machine learning to search for new reactivity. *Nature* **2018**, *559*, 377–381.
127. Burger, B.; Maffettone, P.M.; Gusev, V.V.; Aitchison, C.M.; Bai, Y.; Wang, X.; Li, X.; Alston, B.M.; Li, B.; Clowes, R.; et al. A mobile robotic chemist. *Nature* **2020**, *583*, 237–241.
128. Boiko, D.A.; MacKnight, R.; Kline, B.; Gomes, G. Autonomous chemical research with large language models. *Nature* **2023**, *624*, 570–578.
129. Hase, F.; Roch, L.M.; Kreisbeck, C.; Aspuru-Guzik, A. Phoenics: a Bayesian optimizer for chemistry. *ACS central science* **2018**, *4*, 1134–1145.
130. Stach, E.; DeCost, B.; Kusne, A.G.; Hattrick-Simpers, J.; Brown, K.A.; Reyes, K.G.; Schrier, J.; Billinge, S.; Buonassisi, T.; Foster, I.; et al. Autonomous experimentation systems for materials development: A community perspective. *Matter* **2021**, *4*, 2702–2726.
131. Stokes, J.M.; Yang, K.; Swanson, K.; Jin, W.; Cubillos-Ruiz, A.; Donghia, N.M.; MacNair, C.R.; French, S.; Carfrae, L.A.; Bloom-Ackermann, Z.; et al. A deep learning approach to antibiotic discovery. *Cell* **2020**, *180*, 688–702.
132. Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; et al. Highly accurate protein structure prediction with AlphaFold. *nature* **2021**, *596*, 583–589.
133. Senior, A.W.; Evans, R.; Jumper, J.; Kirkpatrick, J.; Sifre, L.; Green, T.; Qin, C.; Žídek, A.; Nelson, A.W.; Bridgland, A.; et al. Improved protein structure prediction using potentials from deep learning. *Nature* **2020**, *577*, 706–710.
134. Real, E.; Liang, C.; So, D.; Le, Q. Automl-zero: Evolving machine learning algorithms from scratch. In Proceedings of the International conference on machine learning. Pmlr, 2020, pp. 8007–8019.
135. Romera-Paredes, B.; Barekatain, M.; Novikov, A.; Balog, M.; Kumar, M.P.; Dupont, E.; Ruiz, F.J.; Ellenberg, J.S.; Wang, P.; Fawzi, O.; et al. Mathematical discoveries from program search with large language models. *Nature* **2024**, *625*, 468–475.
136. Fawzi, A.; Balog, M.; Huang, A.; Hubert, T.; Romera-Paredes, B.; Barekatain, M.; Novikov, A.; R. Ruiz, F.J.; Schrittwieser, J.; Swirszcz, G.; et al. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature* **2022**, *610*, 47–53.
137. Mankowitz, D.J.; Michi, A.; Zhernov, A.; Gelmi, M.; Selvi, M.; Paduraru, C.; Leurent, E.; Iqbal, S.; Lespiau, J.B.; Ahern, A.; et al. Faster sorting algorithms discovered using deep reinforcement learning. *Nature* **2023**, *618*, 257–263.
138. Bran, A.M.; Cox, S.; Schilter, O.; Baldassari, C.; White, A.D.; Schwaller, P. Chemcrow: Augmenting large-language models with chemistry tools. *arXiv preprint arXiv:2304.05376* **2023**.
139. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. PCA-OS: A Planetary Climate Adaptation Operating System **2026**. Preprint.
140. Recht, B.; Roelofs, R.; Schmidt, L.; Shankar, V. Do imagenet classifiers generalize to imagenet? In Proceedings of the International conference on machine learning. PMLR, 2019, pp. 5389–5400.
141. Bouthillier, X.; Delaunay, P.; Bronzi, M.; Trofimov, A.; Nichyporuk, B.; Szeto, J.; Mohammadi Sepahvand, N.; Raff, E.; Madan, K.; Voleti, V.; et al. Accounting for variance in machine learning benchmarks. *Proceedings of machine learning and systems* **2021**, *3*, 747–769.
142. Bowman, S.; Dahl, G. What will it take to fix benchmarking in natural language understanding? In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 4843–4855.
143. Kiela, D.; Bartolo, M.; Nie, Y.; Kaushik, D.; Geiger, A.; Wu, Z.; Vidgen, B.; Prasad, G.; Singh, A.; Ringshia, P.; et al. Dynabench: Rethinking benchmarking in NLP. In Proceedings of the Proceedings of the 2021 conference of the North American chapter of the Association for Computational Linguistics: human language technologies, 2021, pp. 4110–4124.

 doi:10.20944/preprints202604.1414.v1

144. Ma, Z.; Ethayarajh, K.; Thrush, T.; Jain, S.; Wu, L.; Jia, R.; Potts, C.; Williams, A.; Kiela, D. Dynaboard: An evaluation-as-a-service platform for holistic next-generation benchmarking. *Advances in Neural Information Processing Systems* **2021**, *34*, 10351–10367.
145. He, C.; Zhou, X.; Wang, D.; Xu, H.; Liu, W.; Miao, C. Let Papers Flow: AI Conferences Should Embrace Submission Explosion via Autonomous Review Pipelines **2026**. Preprint.