WHITE PAPER

Agent-*Sourced*

A provenance standard for the agent era.

Author	Published	Publisher	Status
Gyasi Sutton	**June 2026**	**TwiceData**	**Draft for discussion**

ABSTRACT

Open source is fracturing over AI-generated contributions: projects are choosing between accepting them and drowning in unreviewable volume, or banning them and forfeiting the upside. This paper proposes a third path — **agent-sourced**, a provenance tag declaring that a change was created by an agent. Rather than adjudicate whether agents *may* contribute, the tag makes origin legible and routes scarce human review to where it is warranted. We define the term, propose a two-tier verification model (raw vs. human-verified), show why a label honors both sides of the current debate, sketch a concrete implementation (an identifier plus a verification step at pull-request time), and enumerate the open governance problems — attestation, category boundaries, transition, and licensing — that a credible standard must resolve in the open.

1 The problem

Open source has a trust problem, and it arrived faster than its institutions were built to absorb. Through 2025 and into 2026, maintainers began to drown. The curl project ended its bug-bounty program on January 31, 2026, after AI "slop" overwhelmed its security queue: the confirmed-vulnerability rate had collapsed from over 15% to below 5% — "not even one in twenty was real," in the maintainer's words [1]. The Zig project adopted an outright no-LLM policy, barring AI tools from issues, pull requests, and tracker comments [2]. Gentoo expressly forbids contributions made with natural-language-processing AI tools; NetBSD presumes LLM-generated code "tainted"; QEMU banned AI contributions on licensing-compliance grounds (a policy it is reconsidering as of mid-2026); and GIMP and Flathub have banned them outright. LLVM took the opposite tack — a human-in-the-loop policy that permits AI-assisted contributions provided a person reads and vouches for them before review [3].

The trigger is the same across projects: ten-thousand-line pull requests from first-time contributors, hallucinated code that does not compile, and review queues flooded faster than humans can read. The debate crystallized publicly — one camp holding that banning AI betrays open source's founding mission that anyone may change software; the other holding that the bans are triage, and that accountability for quality must remain with a person [5]. Both positions are defensible, which is precisely why the argument does not resolve. The framing forces a binary: **accept everything, or ban everything.**

2 The proposal: agent-sourced

The binary is a category error. "Should agents be allowed to contribute" is the wrong question; the operative question is *how a contribution's origin is declared.*

> **Agent-sourced** *(n., adj.)* — a change, project, or artifact created mostly autonomously by an agent (one or more, not necessarily a population), with human input extremely low. It is a provenance label, the agent-era analog to *crowdsourced*: it tells the audience what they are looking at, so the work can be trusted, reviewed, and used accordingly.

The tag is not a verdict and not a quality claim in itself. It is a declaration of origin. In its clearest form, the agent did most or all of the authoring while the human's input was extremely low — a spec, a prompt, a review, a decision to ship. When the producers of code change — from people to agents acting on their behalf or autonomously — the first obligation to everyone downstream is honest provenance. This reframes the dispute from *permission* to *transparency*.

3 Why labeling beats banning — and beats blind trust

A provenance tag does two things at once. It **flags the contribution for deeper review**, so a maintainer knows where to concentrate scrutiny; and it **still permits agent-driven contribution**, so the project does not forgo the genuine productivity of agent work. It substitutes honest provenance for the false choice between gatekeeping and blind trust.

It is also the low-commitment option, which is the practical reason it could be adopted. A maintainer's only moves today are expensive: fully review and merge, or ban the entire category. A label is the lighter middle — neither vetting everything nor forbidding everyone, but labeling and letting the label route attention. Critically, it ships as metadata, *not* as another automated agent that fixes issues or reviews pull requests. Additional bots add the very load maintainers are banning; a tag changes almost nothing in the workflow and simply makes origin legible.

4 A two-tier verification model

Provenance and verification are distinct axes, and conflating them is what makes a single label feel insufficient. We separate them with two tiers *within* the agent-sourced category — a designation of origin paired with a level of human verification.

TIER 1 — RAW AGENT-SOURCED

The agent's output as submitted, not yet vetted by a person. No party vouches for it. A maintainer may ignore the Tier-1 backlog entirely, without obligation, or review it as time permits.

TIER 2 — HUMAN-VERIFIED AGENT-SOURCED

A person has reviewed the work and vouches for it — yet it remains agent-sourced. Verification does not erase provenance; both facts travel together (*made by an agent, checked by a human*). Tier 2 is a required, non-bypassable designation: nothing enters a release or parent project without it.

The value to the maintainer is a clear ownership model and an explicit choice: ignore the raw tier, and spend scarce attention only on *promoting* worthwhile work from Tier 1 to Tier 2. Verification is deliberate and costly, and it is the prerequisite for graduation into the base project. Provenance categories (agent-sourced, agent-assisted, human-sourced) describe authorship; the tiers describe verification. The two are orthogonal and should not be collapsed.

5 Honoring both positions

Agent-sourced code is double-edged, and the tag denies neither edge. It can conceal latent defects — "bug-bombs" that, merged unverified, take weeks to excavate — which is the skeptic's legitimate fear. It can also deliver fast, substantive innovation, which is the optimist's legitimate hope. The tag validates both at once: *flag it and verify before trusting*, and *welcome it rather than ban it.* It therefore fits the debate without taking a side.

In practice the structural buffer is the fork. Agent work is permitted to live on a tagged agent-sourced fork, where it can develop freely while remaining quarantined from the base. When a fork yields something demonstrably valuable, a human cherry-picks it and opens a pull request into the base — a human-gatekept graduation. The base stays clean, experimentation stays unconstrained, and a person continues to hold the gate.

6 Implementation: an identifier and a verifier

Concretely, agent-sourced reduces to two buildable components. The first is an **identifier** — the provenance tag itself, carried on a commit or pull request as a signed trailer, a label, or a machine-readable field. The second is a **verification framework** at pull-request time — the mechanism that promotes a contribution from Tier 1 to Tier 2: who attests, what is checked, and where the gate sits in the review flow. Everything else (badges, CI checks, provenance queries, review dashboards) is built on those two primitives.

This paper takes no position on closed or proprietary code. The pressing case is the open-source contribution pipeline; whether the same identifier is adopted internally by a private organization is a decision for that organization, not a prescription of the tag.

7 Open problems

The honest difficulty is governance. A credible standard must answer four questions, none of which one party should decide unilaterally.

7.1 Attestation

Who or what certifies a contribution's provenance and verification level — the agent, the hosting platform, a cryptographic signature, a CI check? Attestation must be cheap to produce and hard to forge.

7.2 The category boundary — and why it must be set in committee

The hardest line to draw is between *agent-assisted* (a human authoring with AI help) and *agent-sourced* (an agent authoring). The label is meant for work that is **mostly the agent's** — where human input is extremely low (a spec, a prompt, a review, a decision to ship) and the agent produced essentially all of the artifact. A workable test is also one of **decisions, not keystrokes**: a contribution is agent-sourced when the agent made the substantive authoring decisions — architecture, algorithms, error-handling, dependency and structural choices — that the human did not dictate; it is agent-assisted when the human made those decisions and the agent supplied mechanical help under direction. Operationally, the safe default is to treat any agent-involved contribution as agent-sourced *unless* the contributor can attest, decision by decision, to having directed the implementation.

But this is a **posit, not a settled rule**, and the difficulty should not be understated. The boundary is judgment-laden: "substantive decision" resists precise definition, heavy prompting blurs who authored what, a single human-changed character can outweigh ninety-nine agent-written lines, and any self-attestation can be gamed. Reasonable practitioners will disagree about identical contributions. And the moment "mostly" enters the definition it invites the questions a standard must answer: how much is "mostly"? what counts as human input — lines, decisions, time, or intent? can input that is low in volume but high in significance — a single pivotal correction — keep a contribution out of the category, and at what point does extremely-low human input tip the work from assisted to sourced? A line this consequential — it governs what gets labeled, scrutinized, and trusted — **cannot be drawn by one author or one company. It must be established, and revised over time, by a body of industry leaders**: maintainers, foundations, platform and tooling vendors, and the labs building the agents, convened to define these categories the way standards bodies have always defined contested terms. Until such a body sets it, every boundary — including this one — is provisional.

7.3 Transition

Provenance is a chain of custody. As a human substantially edits agent-sourced work, it shifts toward agent-assisted, then human-sourced — the tag transitions with authorship and should carry its history. This is the constructive inverse of the present conflict: instead of agent output flooding human projects, agent-sourced work can graduate into human open source through a defined transition.

7.4 Licensing

The legal status of AI-authored work is unsettled. The tag should therefore declare both provenance *and* license, so downstream users know origin and terms, and the rules must specify how license and attribution carry across a transition in authorship.

8 A call to codify

These rules earn legitimacy only by being defined in the open, by the parties they affect — as the Open Source Initiative defined "open source," C2PA defined content provenance, and SPDX standardized license identifiers [4]. This paper is therefore also an invitation: to maintainers and foundations, to the platforms that host code, to the labs whose agents are doing the contributing, and to those who work on licensing — to convene and define the attestation, boundary, transition, and licensing rules an agent-sourced standard requires.

9 Conclusion

The open-source AI debate need not end in either a ban or a flood. It can end in graduated, verifiable trust: agent contributions welcomed, labeled, and reviewed in proportion to their provenance. The mechanism is modest — a tag, two tiers, and a gate — but it converts an unwinnable argument about permission into a tractable problem of transparency. What the industry needs is not a verdict on agents, but a vocabulary for their work. *Agent-sourced* is a proposal for that vocabulary.

References

1. D. Stenberg, “The end of the curl bug-bounty” (Jan 2026) — daniel.haxx.se; and “Death by a thousand slops” (Jul 2025). Coverage: The New Stack; BleepingComputer.
2. L. Cro, “Contributor Poker and Zig’s AI Ban,” Loris Cro’s blog (29 Apr 2026) — kristoff.it/blog/contributor-poker-and-ai
3. Project AI-contribution policies (2025–2026): Gentoo, NetBSD, QEMU, GIMP, Flathub (bans); LLVM (human-in-the-loop permit). Survey: RedMonk, “Generative AI Policy Landscape in Open Source” (Feb 2026).
4. Open Source Initiative (opensource.org); Coalition for Content Provenance and Authenticity, C2PA (c2pa.org); SPDX License List (spdx.org/licenses).
5. ThePrimeagen, “is DHH wrong?” (2026) — video reaction to DHH’s position on AI and open source. youtube.com/watch?v=pkndFYSTr0Y

Gyasi Sutton · June 2026.

Agent-Sourced — a white paper. Published by TwiceData. A draft for discussion, offered to be argued with and improved.

Companion reading: the blog introduction.