"AI Slop is DDoSing Open Source": Understanding the Impact of AI-Generated Contributions on Open Source Sustainability

Sadia Afroz[1] Courtney Miller[2] Tyler Menezes[3] Edward Gilmour[1] Anita Sarma[1] Zixuan Feng[4]
[1]Oregon State University, Corvallis, OR, USA ({afrozs, gilmoued, anita.sarma}@oregonstate.edu)
[2]The George Washington University, Washington, DC, USA (courtney.miller@gwu.edu)
[3]CodeDay, Seattle, WA, USA (tylermenezes@codeday.org)
[4]Virginia Commonwealth University, Richmond, VA, USA (fengz3@vcu.edu)

***Abstract*—Open source software (OSS) communities are facing increasing pressure from Generative AI (GenAI) tools. We call it AI-DDoS: a denial-of-service effect in which plausible but low-quality AI-generated contributions overwhelm OSS community capacity. Using a phenomenon-based mixed-methods approach, we first analyze practitioner accounts from Reddit, OSS mentor mailing lists, and blogs to identify six recurring themes and derive hypotheses. We then evaluate these hypotheses using Bayesian Structural Time Series analysis across 294 repositories with over 2 million pull requests and issues. Our results show that while PR volume increased in 2025, merge rates declined, with one-time contributors experiencing an 18.18% drop in PR merge rates relative to the counterfactual. Finally, we identify 11 remediation strategies through practitioners' interviews and validate them with a survey of 229 OSS practitioners, grouping them into preservative, adaptive, and transformative orientations. Our findings show that AI-DDoS is not only a contribution-volume problem but a sustainability trap: communities often default to low-effort defensive strategies that protect short-term review capacity while making openness difficult to sustain.**

***Index Terms*—AI-generated Contribution, OSS, AI-DDoS.**

I. Introduction

Generative Artificial Intelligence (GenAI) tools for software development have dramatically lowered the cost of producing code while raising the cost of evaluating it [1], [2]. In open source software (OSS) communities, this asymmetry produces a phenomenon we call *AI-DDoS*: a flood of plausible-looking, low-quality GenAI contributions that disrupts the contribution process and overwhelms the limited time and resources maintainers have to review contributions. Figure 1 presents an overview of the steep rise in contributions and the decline in accepted contributions.

OSS practitioners have begun responding defensively to this barrage [3], [4]. In January 2026, the cURL project ended its six-year bug bounty program after AI-generated contributions reduced the proportion of valid submissions to about 5% by July 2025, making triage unsustainable [4], [5]. Others have moved to invitation-only contribution workflows, adopted blanket policies rejecting unsolicited pull requests, and stopped accepting outside vulnerability reports [2], [3].

These responses help projects survive the immediate flood, but they narrow or completely block the pathways through

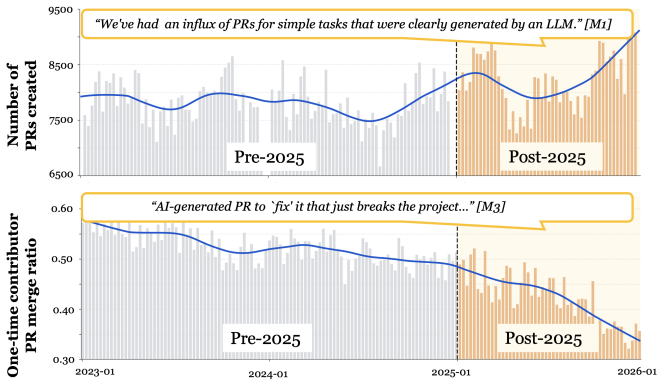


Fig. 1. Overview of contribution flow.

which OSS sustains itself. Over a decade of OSS sustainability research establishes that a project's health and survival depend on its maintainers, who provide the labor that keeps the project alive, and on its ability to recruit and retain new contributors who renew that labor over time [6], [7]. The problem is that the programs OSS projects are shuttering to cope with AI-DDoS are the same ones they use to recruit and retain those contributors (e.g., good first issues [8]). AI-DDoS and the strategies used to remediate it therefore reach well beyond their immediate toll on the maintainers (e.g., Linux [9] and Ghostty [10]). The first impact is on the projects themselves: as communities close their doors to survive the flood, they cut off the contributor pipeline that renews maintainer labor, turning open projects into closed systems that slowly atrophy through attrition. The second impact is systemic: as these projects weaken, so do the critical software supply chains that depend on them [11].

However, despite a rapidly growing body of practitioner accounts describing their lived experience with AI-DDoS [9], [12], we lack an empirically grounded understanding of how prevalent it is and how it manifests across ecosystems. We also do not know whether AI-DDoS is a transient issue that existing contribution workflows can absorb, or a structural shift that the contribution workflows cannot withstand without redesign.

In this paper, we address this gap through a phenomenon-based, mixed-methods study [13]. Phenomenon-based research (*PBR*) [14] aims to identify, document, and conceptualize an emerging phenomenon to advance knowledge through phenomenon selection, framing with stakeholders or theoretical grounding, and data collection. Following this approach, we characterize the phenomenon AI-DDoS as an emerging form of structural pressure on OSS, examine how it affects contribution workflows in practice, and identify remediation strategies for maintainers and the broader OSS community. We ask two research questions:

- **RQ1.** *How does AI-DDoS manifest in OSS contribution workflows?*
- **RQ2.** *What AI-DDoS remediation strategies can maintainers and the broader OSS community adopt?*

To answer **RQ1**, we first triangulate practitioner accounts across three independent sources of gray literature (Reddit, practitioner blogs and the OSS mentors' mailing list) to characterize how AI-DDoS manifests in practice, identify the practitioner-reported mechanisms underlying these manifestations, and derive hypotheses about how it would appear in repository-level contribution data. We then test these hypotheses at ecosystem scale, using Bayesian Structural Time Series (BSTS) causal impact analysis across 294 widely-used projects spanning more than 2 Million pull requests and issues. We treat 2025 as the intervention boundary, the year in which practitioner reports converged and AI-assisted contribution became a routine part of OSS workflows [15], [16]. This study design lets us move from what practitioners report experiencing to whether those signals appear in the contribution workflows of real projects.

To answer **RQ2**, we conducted semi-structured interviews with 11 OSS maintainers, guided by Organizational Resilience Theory [17], [18], to identify the strategies communities are mobilizing against AI-DDoS . We then surveyed 229 OSS practitioners to test how widely these strategies are accepted across the OSS ecosystem.

Across both RQs, the evidence converges: AI-DDoS is an emerging, ecosystem-wide structural crisis, not an isolated complaint. Practitioner accounts and repository data agree on a post-2025 shift in which contribution volume rises while quality and completion decline.

In summary, we contribute: **(1)** A characterization of how the AI-DDoS phenomenon manifests in practice, drawn from practitioner-reported themes triangulated across three independent sources of gray literature. **(2)** Causal impact evidence that AI-DDoS is a structural, ecosystem-wide problem: across 294 widely-used projects, PR volume rose while merge rates and issue completion fell after 2025, with the sharpest effect among one-time contributors. **(3)** A set of 11 practitioner-driven, validated (survey with N=229) remediation strategies that surfaces a central tension: communities favor low-effort preservative strategies that can harden into defensive closure over time.

II. Background

Open participation is a long-term investment. OSS sustainability is a project's ability to continuously generate the labor required to maintain it, through sustained maintainer engagement, the attraction and retention of contributors, and the conversion of newcomers into core developers [6], [19]. The well researched "*onion model*" describes how contributors move from passive users to active users, patch contributors, and eventually core developers [20]–[22].

Prior work has therefore examined how OSS projects can keep this pathway open by lowering entry barriers through "good first issues", mentoring programs, contribution guidelines, and structured onboarding practices [23], [24]. Yet this model of sustainability relies on trust: trust that outsiders arrive with good-faith intent, that maintainers can distinguish promising newcomers from low-quality noise, and that welcoming many outsiders will replenish the contributor base [25]. When that trust fails, openness becomes a liability.

AI-assisted tools weaken this trust. For example, reviewing a contribution requires maintainers to assess not only functional correctness, but also architectural fit, test coverage, and long-term maintainability [26], [27]. In the past, this review burden was partly constrained by the effort required to produce a plausible contribution, which demanded project knowledge and sustained effort [28]. Today, AI-assisted tools weaken that constraint by making plausible-looking contributions faster and cheaper to produce [29], [30], while the labor needed to maintain them remains limited and exhaustible [25], [31].

The 2025 Stack Overflow Developer Survey reports that 84% of developers use or plan to use them and 51% use them daily, while GitHub's Octoverse 2025 reports platform-scale growth in developers and merged pull requests [15], [16]. Controlled studies report productivity gains [32], [33], especially for less experienced developers [34], but emerging evidence shows quality tradeoffs, including increased churn, reduced reuse, more static-analysis warnings, and higher complexity [35], [36]. GitHub describes 2025 as an OSS "Eternal September", where rising contribution volume overwhelms human attention and strains trust [2]. CodeRabbit reports that AI-generated pull requests contain 1.7 times more issues than human-written ones [37].

OSS projects are putting up additional gates. The impact of the AI-assisted contribution pressure is already visible at the level of individual projects, and the responses it provokes reveal the deeper sustainability threat. In January 2026, cURL ended its six-year bug bounty program after AI-generated reports pushed the valid-submission rate from roughly one in six to one in twenty or thirty [38]. Other projects have responded by adding stronger gates around outside contributions: Ghostty requires disclosure of AI use, human understanding of submitted code, and sanctions for low-effort AI-generated work [3]. The Linux kernel has similarly responded to duplicate AI-generated vulnerability reports by redirecting them away from the private security list, while requiring AI-assisted contributions to remain transparent and legally accountable

through human sign-off and the `Assisted-by` tag [9].

These responses may help projects survive the immediate pressure, but at the cost of narrowing the open pathways through which OSS sustains. *This creates an urgent need to investigate GenAI-driven contribution pressure in OSS: how it reshapes OSS sustainability, and how communities respond to openness becoming a source of strain.*

III. The Genesis of AI-DDoS (RQ1)

To answer **RQ1** through two stages, following the PBR guidance on phenomenon selection and data collection for emerging phenomena [14]: Stage 1 characterizes AI-DDoS phenomenon by triangulating practitioner-reported mechanisms across three sources of gray literature, and Stage 2 validates whether the phenomenon appears in repository-level contribution traces using causal impact analysis.

A. Stage 1 Method: Triangulated Qualitative Analysis

We begin by analyzing practitioner reports from gray literature to characterize AI-DDoS as an emerging practitioner-reported phenomenon and to develop hypotheses for quantitative validation. Because AI-DDoS is recent and still unfolding in OSS, gray literature provides timely evidence of early signals, community concerns, and practitioner responses, and has been widely used in empirical SE studies [39], [40].

1) Data Collection: To identify relevant artifacts, we use keyword-based search for *practitioner blogs* and *Reddit (r/opensource)* followed by an exhaustive review of the discussions on the *mentors' mailing list*, following established SE practices for analyzing gray literature [39]. These three independent sources enable data triangulation, strengthening the credibility of our qualitative findings.

Practitioner Blogs. To determine which practitioner blogs to include in our search, we followed standard blog inclusion approaches used in prior SE studies [39], [41]. We first identified the top 50 OSS blog platforms from Feedly's "Best Open Source Blogs and Websites" list (https://feedly.com; accessed March 2026). We excluded project-specific and primarily release-announcement-driven blogs or those lacking editorial curation, resulting in 32 eligible platforms. These included foundation- and community-governed platforms such as the Open Source Initiative [42], Google Open Source Blog [43], and Linux.com [44].

To retrieve relevant blogs, we first developed an initial keyword set from two blog posts [45], [46] already known to the research team. These posts discussed AI-generated/-assisted contributions and provided seed terminology for the search process. From this initial review, we identified nine search terms [47], such as *AI-generated pull requests* and *low-quality contributions*. We used those to retrieve potentially relevant blogs. Three researchers independently reviewed each candidate blog artifact for relevance. They then met weekly to discuss disagreements and reach an agreement on the final blog set. This resulted in 12 relevant blog post artifacts (B1–B12). See the supplementary materials for the bloglists [47].

Reddit OSS community r/opensource captures reactive, time-sensitive discourse from a broad community of practice [48] reflecting immediate experiences, concerns, and observations [49]. Prior works have shown that such developer-facing social platforms can surface early indicators of socio-technical change and evolving practitioner concerns [29], [50]. We apply the same set of keywords used to search for relevant blog posts to identify relevant discussions within this subreddit community (*r/opensource*), yielding 34 candidate threads. Three researchers independently assessed each thread for relevance and reached full agreement through weekly negotiated agreement sessions. During these sessions, ten threads were excluded because they were not specific to OSS, resulting in a final corpus of 24 threads (R1-R24) comprising 334 comments.

OSS Mentors Mailing List (MML) captures practitioner communication in a structured, multi-organization OSS environment, where mentors coordinate expectations, practices, and responses across participating OSS organizations. Recurring discussions in the mailing list provide evidence of emerging ecosystem-level concerns [51].

We obtained access to the full mentors' mailing-list archive in accordance with IRB-approved procedures. Given our focus on AI-assisted contributions, we reviewed five years of discussion history (2022–2026) and conducted an exhaustive review of all email discussions. Three researchers independently assessed all emails for relevance, with discrepancies resolved through weekly meetings. This process identified 13 relevant discussion threads (M1–M13), comprising 119 emails.

2) Qualitative Analysis.: We used reflexive thematic analysis [52] for all artifacts to characterize community reactions, practitioner experiences, and emerging pressures associated with AI-generated/-assisted contributions. Three authors iteratively analyzed the dataset, developing preliminary codes and discussing emerging interpretations in weekly meetings. Through these discussions, we refined our understanding of the data, reorganized related codes, and developed higher-level themes, which informed our characterization of AI-DDoS as a form of contribution pressure on OSS communities.

B. Stage 1 Results: AI-DDoS as a Progressive OSS Breakdown

Our qualitative analysis reveals **AI-DDoS** as an emerging practitioner-reported phenomenon in OSS: a denial-of-service [53] effect in which AI-generated contributions overwhelm a project's capacity to review, triage, validate, and absorb contributions. In the following sections, we unpack this phenomenon through six recurring themes, as shown in Table I. These themes show cross-source convergence among the three data sets we analyzed, with a concentration in 2025.

T1: AI-DDoS begins as a traffic flood. *"AI slop flooding open source"* [B3] [12], and projects having *"...bug reports with AI-generated contents that don't offer any real value"* [B9] [54]. *"We've had an influx of PRs for simple tasks that were clearly generated by an LLM."* [M1]. In bug bounty and security-reporting systems, the projects were *"inundated with AI slop, where many bogus reports were opened"* [B10] [5] as *"Now they face script kiddies opening Cursor, finding*

TABLE I
QUALITATIVE THEMES WITH SEGMENT COUNTS IN SOURCES.

Theme	Description	R	B	M	Tot.
T1. Volume and submission flood *Blog May'25; Reddit Nov'23; MML Mar'25*	AI tools generate contributions (e.g., PRs, issues) at a rate that exceeds review capacity.	16	20	41	**77**
T2. Quality collapse of submissions *Blog May'25; Reddit Apr'23; MML Mar'25*	AI submissions look plausible but fail to meet quality expectations.	11	7	17	**35**
T3. Transient contributor *Blog May'25; Reddit Sep'25; MML Nov'25*	Contributors use AI to mass-generate drive-by submissions.	8	4	10	**22**
T4. Incentive and platform structure *Blog May'25; Reddit Sep'25; MML Mar'25*	Structural incentives such as bounties, heatmaps, and pay-to-play stratification magnify AI-assisted submission.	9	9	1	**19**
T5. Maintainer and mentor wellbeing *Blog May'25; Reddit Jan'26; MML Mar'26*	The flood overexhausts mentors and maintainers, driving burnout and withdrawal.	4	6	9	**19**
T6. Defensive contraction *Blog May'25; Reddit Aug'25; MML Apr'25*	Communities respond by closing channels, blocking by default, and restricting newcomer access.	7	15	11	**33**

R = Reddit, B = Practitioner blogs, and M = Mentors mailing list discussions

bugs, generating fixes, and submitting pull requests without understanding the code." [B3] [12]. The speed of the flood was also visible in the cases where *"seven [X] reports came in within a 16-hour period."* [B10] [5].

T2: The flood consists of low-quality contributions. Practitioners described this high traffic as *"PRs that compile but are unreviewable"*[R1], and *"...low-quality submissions where it isn't obvious whether the submission came from a person or an AI model"*[B1] [4]. Some submissions contained fabricated evidence, such as *"AI-generated 'screenshots' of non-existent [X] UI"* [M4]. In more harmful cases, AI-generated work appeared as a full issue-to-PR chain: *"opening an AI-generated nonsense issue and then an AI-generated PR to 'fix' it that just breaks the project by deleting large blocks of code to improve 'performance"'* [M2].

This leads us to **Phenomenon 1 (Low quality traffic overload)**: AI-DDoS begins when OSS projects receive a high volume of AI-generated contributions that are often low-quality, fabricated, or harmful.

T3: Drive-by contributions. Practitioners linked AI-DDoS to transient contributors who contribute once and never engage with the community again, using AI to generate drive-by contributions. Some projects have already begun taking steps to address this pattern [55]. *"it's low-effort drive-by contributions from people who don't understand the codebase. AI just made it cheaper to produce them at scale"* [R3]. *"We can't do anything about drive-by submissions from those who don't engage with the org."* [M3].

T4: Drive-by contributions are amplified by incentive abuse. Practitioners connected these drive-by contributions to external incentives around GitHub visibility and the job market [5], [56]. *"I have definitely noticed an increase in people trying to exploit open source. Markets are tough, so there's incentive to farm GitHub contributions for resumes"* [R14]. *"The system right now incentivizes: push code fast, get PRs merged, pad the resume. AI just lowered the barrier to generating volume"* [R1]. Practitioners warned that *"Some people are trying to build up realistic looking GitHub profiles so that they can do supply line attacks"* [R8].

These accounts point to **Phenomenon 2 (Incentivized drive-By contribution)**: AI lowers the cost of generating visible OSS contribution traces, amplifying profile-building behavior beyond one-time contributors.

T5: Denial-of-service effect: Human Breakdown. The first breakdown produced by AI-DDoS occurs at the human layer [4], [12]. Practitioners reported that *"it's just not realistic for any maintainer to look at the sheer volume of incoming pull requests that we're getting right now"* [M9]. *"The deluge of PRs is just wasting our time"* [M7]. Maintainers must now go through *"the torrent of nonsensical PRs generated by LLMs, in search of the handful of PRs by genuine new contributors"* [R1], and even *"rejecting them takes more time than just writing the code"* [B12] [56].

The breakdown is not only a matter of workload. Before AI-generated submissions, mistakes in pull requests could signal genuine learning; reviewers could identify where contributors were confused and help them improve [57]. *"I've enjoyed reviewing PRs in the past, often with mistakes as many make their first PR. Because we know that that is a genuine effort to learn. But now it feels like we are doing free work for robots to validate their code"* [M9]. *"And now am I supposed to bond with Claude or Gemini? In the past, I can clearly see from the patch our contributors submitted what confusion they had, and the tough design decisions they had to make... Now these gaps are smeared by LLM"* [M10].

T6: Denial-of-service effect: System Breakdown. As human capacity becomes overwhelmed, AI-DDoS begins to reshape the openness. *"We've had to stop assigning issues at all"* [M7]. Some practitioners described more formal exclusionary responses: *"It's so bad that this is the first year in our 10 years of existence that we have ever blocked contributors"* [M6]. In some cases, the pressure led not only to gating but also to withdrawal, *"Python has multiple sub orgs taking the year off due to burnout, so you're not alone"* [M7].

We therefore characterize **Phenomenon 3**: (Denial-of-service effect): AI-DDoS first breaks the human layer of OSS; communities then protect themselves by narrowing the open channels that once sustained broad participation.

C. Stage 2 Method. Quantitative Validation with Repo Data

Stage 1 characterizes practitioner-reported AI-DDoS, with evidence converging around 2025 across three independent sources. Stage 2 examines whether this phenomenon corresponds to repository-level changes.

Guided by the themes from Stage 1 (Table I), we developed four primary hypotheses on contribution volume and quality, and then extended them to contributor-behavior hypotheses.

The hypothesis-theme mapping is provided in the supplementary material [47]. We then use a Bayesian Structural Time Series (BSTS) causal impact framework [58] to test whether repository-level behaviors changed significantly after widespread AI-assisted contribution emerged. BSTS is well suited to our study because our objective is to estimate the counterfactual trajectory of repository activity following an ecosystem-wide intervention for which no natural comparison group exists. It learns the pre-intervention temporal dynamics and projects the expected post-intervention trajectory under the no-intervention scenario which allows us to compare the observed outcomes with their estimated counterfactual.

Themes to Hypotheses. As shown in Table I (T1, T2, T4, and T5), practitioners across all three sources reported that AI-assisted contributions **increase contribution volume while reducing contribution quality**. Therefore, we propose four primary hypotheses: **H1: After AI adoption, contribution activity changes as follows**: PR volume increases (**H1a**); PR merge ratio decreases (**H1b**); issue volume increases (**H1c**); and issue completion ratio decreases (**H1d**).

T3 concerns *one-time contributors* (contributors who submitted exactly one PR or issue to a repository). Yet the incentive and platform structures associated with AI-assisted contribution (T4) may reshape participation among both transient (one-time) and recurring (*non-one-time*) contributors. To examine whether aggregate shifts are driven by different contributor groups, we extend H1 by decomposing contribution activity into recurring contributors and transient contributors. Accordingly, we hypothesize: **H2: Among non-one-time contributors, the H1 patterns are reflected after AI adoption**: PR volume increases (**H2a**); PR merge ratio decreases (**H2b**); issue volume increases (**H2c**); and issue completion ratio decreases (**H2d**). **H3: Among one-time contributors, the H1 patterns are reflected after AI adoption**: PR volume increases (**H3a**); PR merge ratio decreases (**H3b**); issue volume increases (**H3c**); and issue completion ratio decreases (**H3d**).

1) From Qualitative Signals to a Repository Sample: To prepare for the analysis, we scoped our repo sampling pool using five criteria. (i) We started from a curated directory of actively maintained repositories (https://goodfirstissue.dev/; accessed March 2026) to focus on OSS projects with observable external contribution activity. (ii) We retained repositories with at least 5,000 GitHub stars to identify widely adopted OSS projects with active contributor communities, following prior OSS studies [27], [59], yielding 355 candidates. (iii) To ensure sustained contribution traffic, we retained repositories with at least 100 contributions per year and excluded repositories with inactivity gaps longer than six months [26]. (iv) We retained repositories created no later than 2023 and continuously active through 2025 to compare contribution activities before and after AI adoption [15], [60], [61]. (v) Finally, we removed archived repositories and those with disabled issues or pull requests to ensure access to contribution-related interactions.

After applying these criteria, we retained 294 repositories, including foundation-governed projects (e.g., *Kubernetes*, *Elasticsearch*), industry-backed platforms (e.g., *JetBrains*, *Alibaba*), and widely used developer tools (e.g., *Grafana*). Then, using the GitHub REST API, we collected PR and issue activity from January 2023 to December 2025, including creation timestamps, contributor identifiers, and outcome status. Because our analysis focuses on human participation, we removed automated accounts with bot markers `[bot]`, `-bot`, or `_bot`. The **final dataset contains 1,224,374 PRs and 822,133 issues from 356,972 unique contributors.**

2) Data Preparation: Before conducting the analysis, we perform two preprocessing steps: (i) *Temporal granularity: weekly.* We aggregate PRs and issues by week to balance temporal sensitivity and model stability for BSTS, as daily data are often sparse and noisy and monthly data may obscure shifts that unfold over shorter periods.(ii) *Aggregation: volume-weighted across repositories.* Because repositories vary substantially in contribution volume, we aggregate weekly metrics across repositories using weights proportional to each repository's weekly contribution volume to better reflect ecosystem-level activity [31], [62].

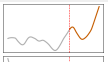
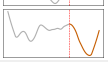
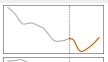
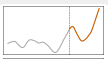
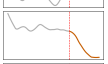
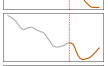
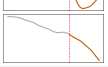
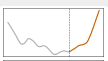
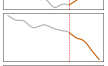
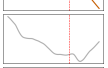
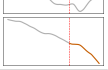
3) Causal Impact Analysis via Bayesian Structural Time Series (BSTS): To assess whether AI-assisted contributions changed OSS workflows, we estimate counterfactual trajectories for each outcome metric using a univariate BSTS model implemented through the `CausalImpact` package [58]. The model learns pre-intervention level without covariates and projects them into the post-intervention period to estimate the expected trajectory had pre-intervention trends continued, following prior studies [63], [64].

We treat 2025 as the intervention period as Stage 1 identified consistent practitioner discussions across all three data sources during 2025 (Table I), indicating that AI-assisted contribution had become a visible OSS workflow phenomenon, consistent with recent tech reports [15], [16]. Given the gradual diffusion of AI-adoption across OSS ecosystems, we use a calendar-year boundary instead of a specific date to avoid false precision while providing a pre-specified intervention point for causal impact analysis [65], [66]. Therefore, in our setting, BSTS learns each metric's expected level and variation from January 2023–December 2024 and predicts this baseline into 2025 to estimate how the metric would have evolved without the surge of AI-assisted contribution in 2025.

We first assessed model robustness using placebo tests [67], setting January 2024 as a pseudo-intervention date for the H1 outcomes. The tests showed no significant placebo effects for the PR-related metrics, but revealed significant placebo effects for issue-related metrics, which we discuss in Section V.

Table II reports the BSTS results for each hypothesis. The *Abs.* column represents the average weekly deviation from the counterfactual and the *Rel.* column reports the corresponding percentage change from the expected trajectory [63]. Posterior probability indicates whether the true effect falls within the hypothesized interval and serves as our primary support criterion, using a 95% threshold [63]. Cohen's d standardizes the effect by pre-intervention variability [68]. We also visualize weekly temporal trends using LOESS smoothing [69], with a dashed vertical line marking the January 2025 AI-surge intervention.

TABLE II
BSTS ANALYSIS RESULTS.

Hyp.	Metric	Abs./Weekly	Rel.	Post.	$\lvert d \rvert$	Trend
Pull Requests						
H1.a	**PR volume***	519.86	+6.80%	99.9%	0.69	
H1.b	**PR merge.***	-0.01	-1.06%	99.9%	0.68	
Issue Requests						
H1c	Issue volume*	-431.15	-8.12%	99.1%	0.65	
H1d	**Issue compl.***	-0.03	-3.40%	99.1%	1.57	
Recurring (Non-one-time) Contributors						
H2a	**PR volume***	497.30	+6.91%	99.9%	0.69	
H2b	**PR merge.***	-0.03	-3.45%	99.9%	2.14	
H2c	Issue volume*	-453.11	-12.40%	99.9%	0.97	
H2d	**Issue compl.***	-0.05	-8.09%	99.9%	1.60	
Transient (One-time) Contributors						
H3.a	**PR volume***	24.67	+5.84%	99.4%	0.43	
H3.b	**PR merge.***	-0.09	-18.18%	99.1%	2.84	
H3.c	Issue volume	58.20	+4.00%	82.8%	0.27	
H3.d	**Issue compl.***	-0.05	-10.30%	99.9%	1.22	

* Posterior probability exceeding 95% indicates a statistically significant effect; hypotheses are bolded when observed effect are significant and in the hypothesized direction. Trend graphs show the pre-intervention period in gray; significant post-intervention changes in orange; and non-significant changes in gray. The dashed red line marks the AI-DDoS intervention. Hyp. = hypothesis, Abs. = absolute effect, Rel. = relative effect, Post. = posterior probability, and $\lvert d \rvert$ = Cohen's d.

D. Stage 2 Results: Quantitative Validation of AI-DDoS

H1a. Increased PR Influx is supported with a medium effect size. As shown in Table II, the temporal plot shows a visible increase in PR volume after AI adoption. The BSTS estimates indicate that weekly PR volume ran 6.80% above the counterfactual during 2025. The *absolute* burden is substantial: the effect corresponds to roughly 520 excess PRs per week across the 294 sampled repositories, accumulating to approximately 27,000 additional PRs over the one-year post-intervention window. An illustrative case is *elastic/kibana*. In that sample, the average weekly PR volume increased from 303 in 2023 to 364 in 2024, a 20%. This moderate year-over-year increase contrasts with the substantially larger 50% increase observed in 2025, when average weekly PR volume reached 545. In absolute terms, this shifted the repository from roughly 19,000 PRs in 2024 to about 28,000 PRs in 2025.

H1b. Reduced PR Merge Ratio is supported with a medium effect size. After AI adoption, the temporal plot shows a downward shift in the PR merge ratio. The BSTS estimates indicate that the size-weighted merge ratio was about 0.01 below the counterfactual on average weekly, corresponding to a 1.06% reduction. When standardized against the metric's own pre-intervention stability, this shift reaches a medium effect size ($\lvert d \rvert = 0.68$, posterior probability 99.9%). This downward displacement suggests reduced mergeability in 2025. The increased PR influx (H1a) and declining mergeability (H1b) together provide quantitative evidence for Phenomenon 1 (low quality traffic overload).

H1c. Issue Volume is not supported but is significant in the opposite direction: issue volume ran 8.12% below the counterfactual, corresponding to an average of about 431 fewer issues per week (roughly 22,000 fewer over the post-intervention year). **H1d. Issue Completion Ratio**, however, is supported. The completion ratio was approximately 0.03 below the counterfactual, a 3.40% relative reduction with a large effect size ($\lvert d \rvert = 1.57$), indicating a substantial shift relative to its typical week-to-week variability.

This does not imply that AI-assisted contributions had no effect on issue workflows. The temporal pattern in Table II shows that issue volume declines during the early post-intervention period but rebounds sharply toward the end of the observation window, as highlighted in orange. Importantly, this rebound coincides with the steepest decline in the issue completion ratio, also highlighted in orange. We interpret this as the delayed propagation of AI's impact on the issue side. Coding-assistant tools matured earlier in 2025, which likely shaped the PR-side behavior first, while agentic tools that scan codebases and generate bug reports are emerging now [70].

H2. The influx propagates to non-one-time contributors. The results for *recurring contributors* follow the same aggregate pattern observed in H1. **H2a** is supported: non-one-time contributors submitted 6.91% more PRs per week than expected ($d = 0.69$). This increase was accompanied by reduced PR merge ratio. **H2b** is also supported, with their PR merge ratio falling 3.45% below the counterfactual ($d = 2.14$).

On the issue side, recurring contributors exhibit a similar pattern as H1. **H2c** is not supported, as issue volume decreased by 12.40% weekly on average under counterfactual($d = 0.97$) , while **H2d** is supported, with the weekly issue completion ratio decreasing by 8.09% ($d = 1.60$).

H3. One-time contributor burden. As shown in Table II, one-time contributors submitted more PRs after AI adoption, but these PRs were much less likely to be merged. **H3a** is supported: weekly one-time PR volume increased by 5.84% relative to the counterfactual ($d = 0.43$). **H3b** is also supported and shows the strongest effect in our analysis: the one-time PR merge ratio was 18.18% lower than expected ($d = 2.84$).

On the issue side, **H3c** is not supported, but **H3d** is: the weekly one-time issue completion ratio was, on average, 10.30% lower than expected ($d = 1.22$). The temporal plot shows that one-time issue volume initially declined but rebounded sharply late in the observation window, aligning with the aggregate issue pattern and suggesting that issue-side impacts emerged later. Practitioners described a similar PR-first dynamic, in which contributors were *"gobbling up"* [M7] issues, especially *"monopolizing the simplest 'good first issues' in the repo"* [M7].

IV. AI-DDoS Remediation Strategies (RQ2)

To answer **RQ2**, we continue following the PBR approach [14] by engaging with OSS maintainers and practition-

ers as the primary stakeholders and using theoretical framing to investigate the remediation strategies that OSS projects use or plan to adopt in response to AI-DDoS .

A. Method

To identify these strategies, we conducted semi-structured interviews with 11 OSS maintainers, followed by a validation survey ($N = 229$) to assess the perceived acceptability of interview-derived strategies across the OSS ecosystem.

1) Interview Protocol: We designed the interviews as a theory-informed follow-up to the AI-DDoS phenomena identified in RQ1. To examine how OSS communities respond to AI-DDoS as an organizational disruption, we used Organizational Resilience Theory (ORT) [17], [18] guiding our interview design and analysis. ORT is well suited here because it explains how organizations absorb disruption, adapt their practices, and learn from adversity [71], providing an analytic lens to investigate the *resilient behaviors* communities enact, the *resources and capabilities* they mobilize, and the longer-term *organizational growth* reflected in governance, review practices, contribution pathways, and resource allocation.

The interview contained five sections. We first collected background information about participants' roles, project size, project domain, and maintainer team size, followed by an icebreaker question about their experiences with AI-generated contributions in OSS. The remaining sections were organized around ORT-informed constructs: *Experience and Challenges*, which examined changes in contributor behavior, review work, and project coordination; *Resilient Behaviors*, which captured how communities accepted, questioned, avoided, negotiated, or acted on AI-assisted contributions; *Resilience Resources and Capabilities*, which examined how communities filtered, reviewed, coordinated, anticipated, and governed these contributions; and *Outcomes and Growth*, which explored whether projects maintained review functions, recovered capacity, learned from disruption, and developed forward-looking strategies. The interview concluded with an open-ended question inviting participants to share any additional experiences or perspectives we may have missed. The details of the interview questions are provided in the supplementary document [47].

Sandbox and Pilot Study. We refined the interview protocol through two internal sandbox sessions and two pilot interviews. The sandbox sessions helped improve question flow, wording, leading prompts, redundancy, and coverage. For example, we avoided the term "AI-DDoS" and used more neutral wording to ask about participants' experiences with AI-assisted contributions in OSS. Pilots with one OSS practitioner and one SE researcher led only to minor wording clarifications.

Participant Recruitment. We recruited 11 OSS maintainers through author contacts and snowball sampling [72], covering diverse project sizes, domains and governance structures. Eight identified as men and three as gender minorities; 81.82% had more than four years of OSS experience and belonged to projects with over 100 contributors (details in [47]).

The study was approved by our institution's IRB. Participants gave informed consent for participation and recording and were compensated with a $50 Amazon gift card. Each 45–60 minute interview was conducted via video call with audio and screen recording. Transcripts were auto-generated and manually corrected by one author.

2) Data Analysis: To identify AI-DDoS remediation strategies, we conducted reflexive thematic analysis of the interview transcripts [52], using an ORT analytic lens. Two authors first inductively open-coded governance actions, policies, tools, and practices that communities had adopted, considered, or rejected in response to AI-assisted contribution pressure. We then used ORT components, *resilient behaviors*, *resilience resources and capabilities*, and *organizational growth*, to interpret how these actions helped communities absorb pressure, adapt workflows, or reconfigure longer-term contribution pathways. Through weekly negotiated agreements meetings, we refined codes, linked them to transcript segments, and iteratively adjusted the codebook. We then merged, split, and grouped related codes into higher-level strategy categories. We reached saturation after eight interviews, when no new codes emerged [73], and conducted three additional interviews to confirm saturation and balance participant perspectives.

In total, we identified 11 remediation strategies and assessed their implementation effort as low, medium, or high based on participants' descriptions of required labor, tooling, workflow changes, and coordination. Guided by ORT, we then grouped the strategies into three resilience orientations: *preservative* strategies protect existing review standards by absorbing pressure, *adaptive* strategies modify current workflows to develop situation-specific responses, and *transformative* strategies reconfigure assumptions about contribution, trust, and participation [74] (Table III). Details are provided in the supplementary document [47].

3) Member Checking: We conducted member checking with interviewees to validate whether our interpretations accurately reflected their experiences, following established guidance [75]. We sent each participant a summary of the synthesized strategies and invited feedback. Seven of the 11 participants responded, collectively covering all synthesized strategies. Their feedback confirmed our findings and provided minor clarifications, but yielded no new insights.

4) Large Scale Validation Survey: To validate the findings and assess community-level acceptance of identified strategies, we conducted a large-scale survey with 229 OSS practitioners.

Survey Design. The survey comprised two sections. The first collected demographic and OSS background information (gender, years of OSS involvement, primary OSS role, and project size). The second presented the 11 strategies derived from interviews, asking respondents to rate their likelihood of adopting each in their projects on a 5-point Likert scale (1 = very unlikely, 5 = very likely). See details in [47].

Participant Recruitment. We recruited participants through our professional OSS networks and via email invitations to maintainers and contributors of the 294 repositories analyzed in RQ1, consistent with prior SE research involving OSS practitioners [57], [76]. The survey remained open for two weeks.

TABLE III
IDENTIFIED 11 AI-DDOS REMEDIATION STRATEGIES WITH CORRESPONDING ORIENTATION, EFFORT AND VALIDATION EVIDENCES.

Strategy Name	Description	Member Checking	Effort	Responses%
*Preservative— **Absorb** the pressure*				
SP1. Tighten CI-based quality gates	Using existing CI, testing, and vulnerability checks to catch defects in AI-assisted submissions before maintainers spend review time.	✓✓✓	▽ Low	43 30 12 — 73%
SP2. Enforce no-review cutoffs for red-flagged submissions	Closing submissions that trigger obvious red flags such as drastic rewrites, excessive file changes, missing approval, scope violations.	✓✓	▽ Low	37 34 15 — 71%
*Adaptive— **Accommodate** the pressure*				
SA1. Require AI-use disclosure	Requiring contributors to state when AI tools were used, so maintainers can track provenance and treat undisclosed AI use as a process violation.	✓	▽ Low	44 19 14 15 — 63%
SA2. Add pre-review contribution gates	Requiring contributors to clarify their intent, planned changes, or permissions before submissions reach maintainer review.	✓✓	■ Medium	24 38 23 11 — 62%
SA3. Flag risky submissions through heuristics	Using behavioral signals (e.g., account age, submission size, commit cadence, bursty PR patterns) to mark submissions for additional scrutiny during review.	✓✓	■ Medium	25 33 21 12 — 58%
SA4. Deploy AI reviewers as first-pass filters	Using AI-powered review tools (e.g., CodeRabbit, Greptile, Graphite) as a first-pass filter on incoming submissions, freeing reviewers for higher-value work.	✓✓	▲ High	24 24 13 16 23 — 48%
SA5. Expand triage through rotation & community review	Increasing triage capacity through rotating triage roles or inviting community members to help organize and review incoming submissions.	✓✓✓✓	▲ High	25 32 19 15 — 34%
*Transformative— **Reconfigure** under the pressure*				
ST1. Verify contributor understanding before acceptance	Requiring contributors to show that they understand what they submitted, through reflection, explanation, or demonstrated technical grounding.	✓✓✓	■ Medium	26 41 20 — 67%
ST2. Align AI policy with project mission	Designing AI policies around the project's purpose, risks, and values, rather than adopting a universal AI rule.	✓✓	■ Medium	27 33 21 — 60%
ST3. Prioritize contributor intent in review	Shifting the unit of review from code to intent, evaluating the contributor's plan, design, and understanding before (or instead of) the code itself.	✓	▲ High	14 29 31 14 12 — 43%
ST4. Ground contribution trust in verified identity / endorsement	Establishing trust at the identity layer by verifying personhood or requiring endorsement by trusted members.	✓✓	▲ High	13 26 32 21 — 39%

Very Likely Likely Neutral Unlikely Very Unlikely

Survey Responses. We received 336 responses, of which 229 were valid. Participants represented diverse OSS roles, project sizes (fewer than 25 to more than 500 contributors), and experience levels, with most reporting 5–9 years (31.88%) or 10–20 years (29.69%) of OSS involvement. Of the participants, 197 identified as men (86.03%) and 32 as gender minorities (13.97%). See details in [47]. All participants provided informed consent under IRB-approved procedures.

For the open-ended questions, we had 83 substantive responses [S1-S83] that we qualitatively analyzed and found that no new mitigation strategies emerged. Most responses either expressed concerns about AI-DDoS or reinforced strategies already identified, e.g., *"General limits on acceptable PR scope, i.e., outright refusing any large PRs that were not planned by the community in advance"* [S39].

B. Results

We synthesized 11 strategies that OSS communities use or plan to use to respond to AI-DDoS pressure and grouped them into three resilience orientations informed by ORT [17], [74]: *Preservative*, *Adaptive*, and *Transformative* (Table III).

1) Preservative Strategies are Holding the Line: Across the three categories, endorsement is highest and most consistent for preservative strategies: tighten CI-based quality gates (SP1, 73%) and enforce no-review cutoffs for red-flagged submissions (SP2, 71%). Both are comparatively low-effort because they impose almost no new infrastructure or social cost. They route AI-assisted submissions through gates the project *already* operates, such as continuous integration, tests, and existing scope/quality bars. *"If we do get huge, large submissions that drastically rewrite things, or don't have approval, we will just auto-reject."* [P5] *"If it looks like it is AI-generated, we will just close the PR"* [P6].

This reflects a preservative resilience orientation in ORT: **communities first absorb disruption by adapting existing infrastructure to a new threat surface before pursuing deeper workflow changes** [74].

2) Adaptive strategies draw conditional, divided support: Adaptive strategies have mixed responses. Those requiring low-to-medium effort were endorsed by more than half of respondents: require AI-use disclosure (SA1, 63%), pre-review gates that clarify contributor intent or scope (SA2, 62%), and flag risky submissions (SA3, 58%). However, The endorsements are split for strategies that require high effort: Deploy AI reviewers as first-pass filters (SA4, 48%), and expand triage through role rotation and community review (SA5, 34%).

Communities are willing to adapt when changes are lightweight, such as a CONTRIBUTING.md file update [P7], PR template edit, or a heuristic label (SA1–SA3). In contrast, they hesitate when adaptation depends on new tooling (SA4) or additional maintainer capacity (SA5). For example, AI reviewers (SA4) are adopted in some projects but not trusted to be autonomous: *"LLMs are very poor at detecting other LLMs"* [P2]. Expand triage (SA5), the least endorsed adaptive strategy, can help distribute review work, but its feasibility depends on whether there are contributors with the time,

expertise, and willingness to take on triage responsibilities.

These patterns reflect ORT's resources-and-capabilities view: ***communities adapt when changes are lightweight, but hesitate when they require new labor, unfamiliar tooling, or additional volunteer capacity [74]***.

3) Transformative strategies change the object of governance, and divide along access: Transformative strategies aim to transform the *object of governance*. Endorsement decreases as these strategies become more effortful and restrictive.

The two most endorsed transformations carry medium effort and redefine *what a contribution must demonstrate*. Verify contributor understanding before acceptance (ST1, 67%) treats explanation and engagement as part of acceptance. Aligning AI policy with project mission (ST2, 60%) similarly ties governance to context: education-oriented projects may tolerate AI use *"Because the project is education-oriented, I think in that sense, we end up being extremely lenient."*[P3], while security- or infrastructure-sensitive projects treat it more cautiously.

By contrast, the two least endorsed transformations require the highest effort because they introduce additional coordination and verification work for communities, especially maintainers. Prioritizing contributor intent in review (ST3, 43%) moves evaluation upstream: contributors must explain what they are trying to change and why before code is reviewed, while maintainers must assess whether that intent is appropriate. As P5 explained, *"now people have to provide their intent, what they're trying to fix, what's wrong, what they're changing, and we can just straight up say, hey, no."* [P5]. Grounding trust in verified identity or endorsement (ST4, 39%) moves governance even further, from the contribution to the contributor. As AI makes credible-looking submissions cheap to generate, some communities consider vouching for verification. *"how do you make sure that someone who is contributing there is a human, is verifiable as a human?"* [P9].

The pattern reflects ORT's organizational growth, ***communities support transformation when it improves contribution quality, but resist it when it needs high human effort*** [74].

V. Threats to validity

Our study relies on multiple sources of empirical evidence, combining qualitative and quantitative methods to characterize AI-DDoS. We discuss the main threats to validity below.

Construct Validity. The RQ1 Stage 1 qualitative analysis (Section III-A) may overrepresent visible, vocal, or highly affected practitioners. Therefore, we treat it as phenomenon identification rather than prevalence estimation, triangulating each theme across three independent sources and validating with repository-level behavior across 294 GitHub repositories.

For RQ2, participants' retrospective accounts may be affected by self-selection bias [77], [78]. We mitigated this by using neutral interview questions and asking participants to ground responses in concrete examples. We also used ORT to structure data collection and analysis, and the validation survey revealed no additional strategy categories, suggesting reasonably comprehensive theoretical framing. To further reduce interpretive subjectivity [79], we validated our strategies through a survey of 229 OSS community members across diverse project sizes, roles, and experience levels— a sample comparable to many empirical SE studies [80].

Internal Validity. Our BSTS analysis estimates the counterfactual trajectory each metric would have followed absent the intervention. Deviations from this trajectory are consistent with AI-DDoS effects but do not rule out other 2025 ecosystem-level changes, such as platform growth or broader shifts in OSS participation. Because we do not observe individual AI use directly, we use 2025 as a prespecified detection window, supported by Stage 1 practitioner convergence, broader evidence that AI-assisted development became routine that year [15], [16], and prior causal impact work [64]. Therefore, we interpret results from 294 widely visible, active repositories as ecosystem-level evidence rather than direct measurements of individual AI use. As a model robustness check, placebo tests using January 2024 as a pseudo-intervention date strengthen confidence in PR findings, but significant placebo effects for issue metrics suggest that issue workflows were already changing before 2025, and we therefore interpret issue-side findings more cautiously [67].

External Validity. To cover sustained public OSS activity and ensure that repository-level changes are measurable over time, our dataset covers popular, actively maintained projects with visible contribution pathways. However, the findings may not fully generalize to lower-traffic, less active, privately managed, or less newcomer-oriented repositories. Similarly, the survey captures the views of OSS participants who chose to respond, and while the sample spans diverse project sizes, roles, and experience levels, it may not represent all OSS communities equally. Future work should examine whether similar AI-DDoS dynamics appear in smaller, private, or less visible projects and across longer observational windows as AI-assisted development continues to diffuse.

VI. Concluding Remarks

Our results converge on a concerning picture: AI-DDoS is emerging as an ecosystem crisis, and communities may respond in ways that undermine their own sustainability and the "openness" that defines them. This is the temptation conventional DDoS defense warns against: we do not protect a service by turning it off [53]. For OSS, the conceptual task is to interpret what that warning means for open participation under AI-generated contribution pressure. This follows the goal of PBR: to document an emerging phenomenon and conceptualize it into practice-relevant understanding [14]. That understanding turns on how communities absorb disruption while staying functional, so we examine the question through ORT, as introduced in RQ2.

As shown in Figure 2, ORT frames the central question through three capacities, *resilient behavior*, *resilience resources*, and *resilience capabilities*: *how can OSS stay open under AI-DDoS ?* From there, the figure traces two paths. Defensive closure leads to a rigidity trap. Adaptive & transformative remediation leads to organizational growth.

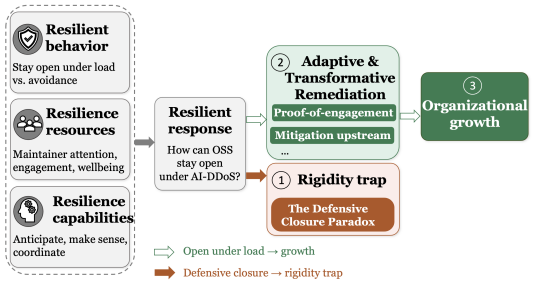


Fig. 2. AI-DDoS through organizational resilience theory.

Rigidity Trap: Defensive Closure. The red path (① in Figure 2) shows defensive closure. Our survey responses (Table III) show that OSS communities are already moving this way: participants widely endorsed preservative strategies because they are cheap, requiring little new infrastructure and little added coordination. The paradox is that these strategies protect maintainers' capacity in the short term by narrowing the contribution pathways that sustain OSS over the long term.

Such strategies can *"kill most of the spray-and-pray submissions"* [R3], but they may also fall hardest on the rough, first-attempt contributions that conventional good first issues were designed to welcome. Under AI-DDoS, what was a chronic depletion of maintainers' capacity [7] becomes acute congestion, and closure starts to look rational: fewer entry points mean fewer submissions to review. But the damage is delayed. Because contributors take time to become long-term contributors [6], a project may appear stable while its future contributor pipeline erodes. The resulting rigidity trap is therefore more than a sustainability challenge for individual communities; it is a latent infrastructure risk for the broader software ecosystem that depends on them.

Do Not Unplug the Server. If the rigidity trap is caused by treating closure as resilience, then the way out is to return to denial-of-service mitigation strategies [81], [82]: keep the service available while separating legitimate from harmful traffic [81], [83]. In technical systems, this involves filtering and scrubbing traffic before it reaches the origin server, using challenge–response mechanisms that impose a small cost on each request [53], relying on reputation and allow-lists built from sender history [53], [83], balancing load across servers [84], and drawing on upstream actors large enough to absorb or deflect floods [85].

The adaptive and transformative strategies in Table III map onto these mitigation strategies, as OSS contributions are social interactions. For example, preservative gates resemble origin-side filters: cheap, familiar, already deployed, but risky. AI reviewers and behavioral flags act as scrubbing layers that identify suspicious submissions before they consume maintainer attention. Verifying contributor understanding and gating on stated intent operate as challenge–response mechanisms by requiring contributors to demonstrate context before receiving maintainer attention. Identity and endorsement provide reputation signals. Triage rotation and community review distribute the review load across more reviewers. Cross-project reputation and identity verification require action at the platform level, which are upstream of any single project.

For example, *proof of engagement* (② in Figure 2) maps onto challenge–response: contributors are asked to explain why a change is needed, how the code works, what alternatives were considered or failed, and how they respond to review. Projects can design questions that are cheap for genuine newcomers who have engaged with the project but costly for volume contributors. AI can generate a plausible pull request, but situated understanding is harder to fake across the discussion, revision, and review cycle. For contributors, the path through the door is therefore to make their engagement with the project visible: disclose AI use, discuss the issue before opening a PR, explain the reasoning behind the change, and respond substantively to review.

Moving mitigation upstream is another example. The pressure cannot be resolved by individual projects alone, especially when AI-DDoS is a structural ecosystem problem. It is produced not only by individual contributors misusing AI, but also from platform incentives that reward visible activity [56], such as contribution graphs, profile visibility, and pull-request counts, also drive it. AI lowers the cost of producing these signals at scale [86], creating a collective-action mismatch: the activity is generated at platform scale, but remediation falls on maintainers one pull request at a time. If platforms benefit from contribution activity, they should also help restore the trust, reputation, and accountability infrastructure that individual projects cannot build alone.

Organizational Growth: Rebuilding Sustainability Infrastructure. Organizational growth (③ in Figure 2) depends on the OSS ecosystem to transform the AI-DDoS pressure into shared sustainability infrastructure. Bringing this transformation about raises three questions for future research.

First, *lightweight proof-of-engagement "gates"* will allow OSS communities to ask contributors for small signs of real engagement before maintainers invest review effort. Crossing these gates should be easy for genuine newcomers and harder for low-context, high-volume contributors. Future work should investigate which questions or tasks best reveal situated understanding, genuine engagement and interest in the project, and how quickly agentic tools learn to bypass these gates.

Second, *portable trust signals across projects* could reduce the need to evaluate every contributor from scratch. Cross-project reputation, contributor history, and personhood verification may help recognize trustworthy participants earlier [23]. Future research should investigate how to design such trust signals without turning OSS into a closed ecosystem.

Third, *tools that help maintainers gauge engagement* could move automation beyond artifact checks such as whether tests pass or a pull request follows a template. Under AI-DDoS, maintainers also need support for engagement signals, like whether contributors explain their changes, respond to review, and understand project context. This raises a tension. If the tools that infer engagement quality are themselves AI-based, and AI is what generates the contributions, then the signals these tools read can be faked as cheaply as the contributions

they were meant to screen. Future research should explore how to surface interaction signals in ways that resist this collapse and keep human judgment central.

In conclusion, AI-assisted contributions are placing OSS communities under structural pressure. This is a sustainability trap as much as a volume problem. The cheapest ways to protect community capacity in the short term will narrow the open pathways built over decades. Filtering AI-assisted work is the easy part. The harder challenge is to identify hard-to-fake signals and rebuild the trust that openness depends on. As one practitioner put it, the community *needs to build new trust, or agents will completely solve it for us* [S22].

REFERENCES

[1] S. Afroz, Z. Feng, T. Menezes, K. Kimura, B. Trinkenreich, I. Steinmacher, and A. Sarma, "The fast and spurious: Developer productivity with genai." FSE-HumanAISE, 2026, 2026.

[2] A. Wolf, "Welcome to the eternal september of open source. here's what we plan to do for maintainers," The GitHub Blog, 2026, accessed: 2026-05-24. [Online]. Available: https://github.blog/open-source/maintainers/welcome-to-the-eternal-september-of-open-source-heres-what-we-plan-to-do-for-maintainers/

[3] ghostty, "Ai usage policy," GitHub repository, `ghostty-org/ghostty`, 2026, accessed: 2026-05-24. [Online]. Available: https://github.com/ghostty-org/ghostty/blob/main/AI_POLICY.md

[4] T. Claburn, "Curl creator mulls nixing bug bounty awards to stop ai slop," The Register, 2025, accessed: 2026-06-16. [Online]. Available: https://www.theregister.com/software/2025/07/15/curl-creator-mulls-nixing-bug-bounty-awards-to-stop-ai-slop/1275332

[5] S. Rudra, "curl gets rid of its bug bounty program over ai slop overrun," 2026, accessed: 2026-06-28. [Online]. Available: https://itsfoss.com/news/curl-closes-bug-bounty-program/

[6] M. Guizani, T. Zimmermann, A. Sarma, and D. Ford, "Attracting and retaining oss contributors with a maintainer dashboard," in *ICSE-SEIS*, 2022, pp. 36–40.

[7] J. Linåker, G. Link, and K. Lumbard, "Sustaining maintenance labor for healthy open source software projects through human infrastructure: A maintainer perspective," in *ESEM*, 2024, pp. 37–48.

[8] GitHub Docs, "Encouraging helpful contributions to your project with labels," https://docs.github.com/en/communities/setting-up-your-project-for-healthy-contributions/encouraging-helpful-contributions-to-your-project-with-labels, 2026, accessed: 2026-06-18.

[9] L. James, "Linus torvalds says flood of duplicate ai-generated vulnerability reports have made linux security mailing list 'almost entirely unmanageable' — private list 'a waste of time for everybody involved' in switch to new public system," Tom's Hardware, 2026, accessed: 2026-06-28. [Online]. Available: https://www.tomshardware.com/software/linux/linus-torvalds-says-ai-bug-reports-have-made-the-linux-security-mailing-list-almost-entirely-unmanageable

[10] P. Joshi, "What ghostty banning ai code contributions teaches builders," Vibe Coder, accessed: 2026-06-16. [Online]. Available: https://blog.vibecoder.me/what-ghostty-banning-ai-code-teaches-us

[11] L. Boughton, C. Miller, Y. Acar, D. Wermke, and C. Kästner, "Decomposing and measuring trust in open-source software supply chains," in *ICSE-NIER*, 2024, pp. 57–61.

[12] A. Team, "The ai slop problem threatening open source maintainers," All Things Open, 19 February 2026, accessed: 2026-06-28. [Online]. Available: https://allthingsopen.org/articles/ai-slop-problem-open-source-maintainers

[13] J. W. Creswell and V. L. P. Clark, *Designing and conducting mixed methods research*. Sage publications, 2017.

[14] F. Lumineau, D. T. Kong, and N. Dries, "A roadmap for navigating phenomenon-based research in management," pp. 505–517, 2025.

[15] Stack Overflow, "2025 developer survey," 2025, accessed: 2026-06-15. [Online]. Available: https://survey.stackoverflow.co/2025/ai

[16] GitHub Staff, "Octoverse: A new developer joins github every second as ai leads typescript to #1," 2026, accessed: 2026-06-28. [Online]. Available: https://github.blog/news-insights/octoverse/octoverse-a-new-developer-joins-github-every-second-as-ai-leads-typescript-to-1/

[17] T. J. Vogus and K. M. Sutcliffe, "Organizational resilience: Towards a theory and research agenda," in *2007 IEEE SMC*. Ieee, 2007, pp. 3418–3422.

[18] J. Hillmann and E. Guenther, "Organizational resilience: a valuable construct for management research?" *International journal of management reviews*, vol. 23, no. 1, pp. 7–44, 2021.

[19] C. Miller, D. G. Widder, C. Kästner, and B. Vasilescu, "Why do people give up flossing? a study of contributor disengagement in open source," in *IFIP International Conference on Open Source Systems*. Springer, 2019, pp. 116–129.

[20] Y. Ye and K. Kishida, "Toward an understanding of the motivation of open source software developers," in *ICSE*. IEEE, 2003, pp. 419–429.

[21] K. Crowston and J. Howison, "The social structure of free and open source software development," 2005.

[22] C. Jergensen, A. Sarma, and P. Wagstrom, "The onion patch: migration in open source ecosystems," in *Proceedings of the 19th ACM SIGSOFT symposium and the 13th ESEC/FSE*, 2011, pp. 70–80.

[23] I. Steinmacher, M. Gerosa, T. U. Conte, and D. F. Redmiles, "Overcoming social barriers when contributing to open source software projects," *CSCW*, vol. 28, no. 1, pp. 247–290, 2019.

[24] Z. Feng, K. Kimura, B. Trinkenreich, A. Sarma, and I. Steinmacher, "Guiding the way: A systematic literature review on mentoring practices in open source software projects," *Information and Software Technology*, p. 107470, 2024.

[25] M. Zhou, A. Mockus, X. Ma, L. Zhang, and H. Mei, "Inflow and retention in oss communities with commercial involvement: A case study of three hybrid projects," *TOSEM*, vol. 25, no. 2, pp. 1–29, 2016.

[26] E. Kalliamvakou, G. Gousios, K. Blincoe, L. Singer, D. M. German, and D. Damian, "The promises and perils of mining github," in *Proceedings of the 11th MSR*, 2014, pp. 92–101.

[27] I. Steinmacher, G. Pinto, I. S. Wiese, and M. A. Gerosa, "Almost there: A study on quasi-contributors in open source software projects," in *Proceedings of the 40th ICSE*, 2018, pp. 256–266.

[28] Tidelift, "The 2024 tideliftstate of the open source maintainer report," https://assets-eu-01.kc-usercontent.com/ef593040-b591-0198-9506-ed88b30bc023/d325a56f-05be-4379-bfd1-ee4776fcad41/2024-tidelift-state-of-the-open-source-maintainer-report-.pdf, september 2024, accessed: 2026-06-15.

[29] V. Pimenova, S. Fakhoury, C. Bird, M.-A. Storey, and M. Endres, "Good vibrations? a qualitative study of co-creation, communication, flow, and trust in vibe coding," *arXiv preprint arXiv:2509.12491*, 2025.

[30] C. Leiter, R. Zhang, Y. Chen, J. Belouadi, D. Larionov, V. Fresen, and S. Eger, "Chatgpt: A meta-analysis after 2.5 months," *Machine Learning with Applications*, vol. 16, p. 100541, 2024.

[31] Z. Feng, S. Afroz, and A. Sarma, "From gains to strains: Modeling developer burnout with genai adoption," *arXiv preprint arXiv:2510.07435*, 2025.

[32] E. Brynjolfsson, D. Li, and L. Raymond, "Generative ai at work," *The Quarterly Journal of Economics*, vol. 140, no. 2, pp. 889–942, 2025.

[33] F. Dell'Acqua, E. McFowland III, E. R. Mollick, H. Lifshitz-Assaf, K. Kellogg, S. Rajendran, L. Krayer, F. Candelon, and K. R. Lakhani, "Navigating the jagged technological frontier: Field experimental evidence of the effects of AI on knowledge worker productivity and quality," Harvard Business School, Technology & Operations Mgt. Unit Working Paper 24-013, 2023. [Online]. Available: https://www.hbs.edu/faculty/Pages/item.aspx?num=64700

[34] S. Peng, E. Kalliamvakou, P. Cihon, and M. Demirer, "The impact of ai on developer productivity: Evidence from github copilot," 2023.

[35] GitClear, "Ai copilot code quality: 2025 look back at 12 months of data," 2025, accessed: 2026-05-24. [Online]. Available: https://www.gitclear.com/ai_assistant_code_quality_2025_research

[36] H. He, C. Miller, S. Agarwal, C. Kästner, and B. Vasilescu, "Speed at the cost of quality: How cursor ai increases short-term velocity and long-term complexity in open-source projects," *arXiv preprint arXiv:2511.04427*, 2025.

[37] D. Loker, "Our new report: Ai code creates 1.7x more problems," CodeRabbit Blog, 2025, accessed: 2026-05-24. [Online]. Available: https://www.coderabbit.ai/blog/state-of-ai-vs-human-code-generation-report

[38] S. J. Vaughan-Nichols, "curl's daniel stenberg: Ai slop is ddosing open source," 2026, accessed: 2026-05-28. [Online]. Available: https://thenewstack.io/curls-daniel-stenberg-ai-is-ddosing-open-source-and-fixing-its-bugs/

[39] V. Garousi, M. Felderer, and M. V. Mäntylä, "Guidelines for including grey literature and conducting multivocal literature reviews in software engineering," *IST*, vol. 106, pp. 101–121, 2019.

[40] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering–a systematic literature review," *IST*, vol. 51, no. 1, pp. 7–15, 2009.

[41] Z. Feng, K. Kimura, B. Trinkenreich, I. Steinmacher, M. Gerosa, and A. Sarma, "Addressing oss community managers' challenges in contributor retention," *ACM TOSEM*, 2025.

[42] Open Source Initiative, "Open source initiative blog," 2026, accessed: 2026-04-18. [Online]. Available: https://opensource.org/blog

[43] Google Open Source, "Google open source blog," 2026, accessed: 2026-04-18. [Online]. Available: https://opensource.googleblog.com

[44] Linux.com, "Linux.com," 2026, accessed: 2026-04-18. [Online]. Available: https://www.linux.com

[45] K. Holterhoff, "Ai slopageddon and the oss maintainers," RedMonk Blog, 2026, accessed: 2026-05-11. [Online]. Available: https://redmonk.com/kholterhoff/2026/02/03/ai-slopageddon-and-the-oss-maintainers/

[46] A. Iyer, "Open source maintainers are drowning in ai-generated pull requests. enterprise teams are next." The New Stack, accessed: 2026-06-28. [Online]. Available: https://thenewstack.io/ai-generated-code-crisis/

[47] Anonym, ""we are being ddosed": How ai-generated contributions threaten open source sustainability — appendix," 2026. [Online]. Available: https://doi.org/10.5281/zenodo.21077989

[48] E. Wenger, "Communities of practice and social learning systems," *Organization*, vol. 7, no. 2, pp. 225–246, 2000.

[49] K. Bergstrom and N. Poor, "Signaling the intent to change online communities: A case from a reddit gaming community," *Social Media+ Society*, vol. 8, no. 2, p. 20563051221096817, 2022.

[50] K. Newman, S. Snay, M. Endres, M. Parikh, and A. Begel, ""get me in the groove": A mixed methods study on supporting adhd professional programmers," in *ICSE*. IEEE, 2025, pp. 1217–1229.

[51] I. Steinmacher, I. Wiese, A. P. Chaves, and M. A. Gerosa, "Why do newcomers abandon open source software projects?" in *CHASE*. IEEE, 2013, pp. 25–32.

[52] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative research in psychology*, vol. 3, no. 2, pp. 77–101, 2006.

[53] L. Garber, "Denial-of-service attacks rip the internet," *computer*, vol. 33, no. 4, pp. 12–17, 2000.

[54] S. Rudra, "Curl is done with ai slop," It's FOSS, 2025, accessed: 2026-06-28. [Online]. Available: https://itsfoss.com/news/curl-ai-slop/

[55] ——, "No ai slops! gnome now forbids vibe coded extensions," It's FOSS, 2025. [Online]. Available: https://itsfoss.com/news/no-ai-extension-gnome/

[56] ——, "Mitchell hashimoto launches 'vouch' to fight ai slop in open source ecosystem," It's FOSS, 2026, accessed: 2026-04-30. [Online]. Available: https://itsfoss.com/news/mitchell-hashimoto-vouch/

[57] Z. Feng, A. Chatterjee, A. Sarma, and I. Ahmed, "A case study of implicit mentoring, its prevalence, and impact in apache," in *FSE*. New York, NY, USA: ACM, 2022, pp. 797–809.

[58] A. H. Kay H. Brodersen, "Causalimpact," https://google.github.io/CausalImpact/CausalImpact.html, 2017, accessed: 2026-04-20.

[59] G. De Palma, S. Giallorenzo, J. Mauro, M. Trentin, and G. Zavattaro, "Funless: Functions-as-a-service for private edge cloud systems," in *2024 IEEE ICWS*. IEEE, 2024, pp. 961–967.

[60] Stack Overflow, "2023 developer survey," 2023, accessed: 2026-06-15. [Online]. Available: https://survey.stackoverflow.co/2023/#ai

[61] ——, "2024 developer survey," 2024, accessed: 2026-06-15. [Online]. Available: https://survey.stackoverflow.co/2024/#ai

[62] C. Chikezie, P. Chanpaisaeng, P. Agarwal, S. Afroz, B. Madhwani, R. Choudhuri, A. Anderson, P. Velhal, P. Morreale, C. Bogart *et al.*, "Measuring ses-related traits relating to technology usage: Two validated surveys," *Empirical Software Engineering*, vol. 30, no. 6, p. 159, 2025.

[63] K. H. Brodersen, F. Gallusser, J. Koehler, N. Remy, and S. L. Scott, "Inferring causal impact using bayesian structural time-series models," 2015.

[64] M. N. Thorakkattle, S. Farhin, and A. A. khan, "Forecasting the trends of covid-19 and causal impact of vaccines using bayesian structural time series and arima," *Annals of Data Science*, vol. 9, no. 5, p. 1025, 2022.

[65] A. K. Wagner, S. B. Soumerai, F. Zhang, and D. Ross-Degnan, "Segmented regression analysis of interrupted time series studies in medication use research," *Journal of clinical pharmacy and therapeutics*, vol. 27, no. 4, pp. 299–309, 2002.

[66] J. L. Bernal, S. Cummins, and A. Gasparrini, "Interrupted time series regression for the evaluation of public health interventions: a tutorial," *International journal of epidemiology*, vol. 46, no. 1, pp. 348–355, 2017.

[67] O. Lenhart, "The effects of paid family leave on food insecurity—evidence from california," *Review of Economics of the Household*, vol. 19, no. 3, pp. 615–639, 2021.

[68] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. New York, NY, USA: Routledge, 2013.

[69] S. K. Basak, L. Neil, B. Reaves, and L. Williams, "What challenges do developers face about checked-in secrets in software artifacts?" in *2023 IEEE/ACM 45th ICSE*. IEEE, 2023, pp. 1635–1647.

[70] E. Roth, "Github's new ai coding agent can fix bugs for you," https://www.theverge.com/news/669339/github-ai-coding-agent-fix-bugs, May 2025, the Verge. Accessed: 2026-06-28.

[71] S. Musa and D. T. Enggarsyah, "Absorptive capacity, organizational creativity, organizational agility, organizational resilience and competitive advantage in disruptive environments," *Journal of Strategy and Management*, vol. 18, no. 2, pp. 303–325, 2025.

[72] L. A. Goodman, "Snowball sampling," *The annals of mathematical statistics*, pp. 148–170, 1961.

[73] J. J. Francis, M. Johnston, C. Robertson, L. Glidewell, V. Entwistle, M. P. Eccles, and J. M. Grimshaw, "What is an adequate sample size? operationalising data saturation for theory-based interview studies," *Psychology and health*, vol. 25, no. 10, pp. 1229–1245, 2010.

[74] C. A. Lengnick-Hall, T. E. Beck, and M. L. Lengnick-Hall, "Developing a capacity for organizational resilience through strategic human resource management," *Human resource management review*, vol. 21, no. 3, pp. 243–255, 2011.

[75] J. Corbin and A. Strauss, *Basics of qualitative research: Techniques and procedures for developing grounded theory*. Sage publications, 2014.

[76] Z. Feng, T. Zimmermann, L. Pisani, C. Gooley, J. Wander, and A. Sarma, "When domains collide: An activity theory exploration of cross-disciplinary collaboration," in *ESEM*. IEEE, 2025, pp. 301–312.

[77] B. Marcus and A. Schütz, "Who are the people reluctant to participate in research? personality correlates of four different types of nonresponse as inferred from self-and observer ratings," *Journal of personality*, vol. 73, no. 4, pp. 959–984, 2005.

[78] S. G. Rogelberg, J. M. Conway, M. E. Sederburg, C. Spitzmüller, S. Aziz, and W. E. Knight, "Profiling active and passive nonrespondents to an organizational survey." *Journal of Applied Psychology*, vol. 88, no. 6, p. 1104, 2003.

[79] A. J. Onwuegbuzie and N. L. Leech, "Validity and qualitative research: An oxymoron?" *Quality & quantity*, vol. 41, no. 2, pp. 233–249, 2007.

[80] D. Russo, "Navigating the complexity of generative ai adoption in software engineering," *TOSEM*, vol. 33, no. 5, pp. 1–50, 2024.

[81] T. Mahjabin, Y. Xiao, G. Sun, and W. Jiang, "A survey of distributed denial-of-service attack, prevention, and mitigation techniques," *IJDSN*, vol. 13, no. 12, p. 1550147717741463, 2017.

[82] S. T. Zargar, J. Joshi, and D. Tipper, "A survey of defense mechanisms against distributed denial of service (ddos) flooding attacks," *IEEE communications surveys & tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013.

[83] M. Yoon, "Using whitelisting to mitigate ddos attacks on critical internet sites," *IEEE Communications Magazine*, vol. 48, no. 7, pp. 110–115, 2010.

[84] M. Belyaev and S. Gaivoronski, "Towards load balancing in sdn-networks during ddos-attacks," in *MoNeTeC*. IEEE, 2014, pp. 1–6.

[85] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and M. Rajarajan, "Scale inside-out: Rapid mitigation of cloud ddos attacks," *TDSC*, vol. 15, no. 6, pp. 959–973, 2017.

[86] H. Ebbers, "When "Contributions" become a burden," Open Elements, 2026, accessed: 2026-06-28. [Online]. Available: https://open-elements.com/posts/2026/03/05/when-contributions-become-a-burden