Nobody Read This Code: What Code Review Must Become When Software Is Written Faster Than It Is Read

RoEduNet 2026 · Iași, 19–21 November

Vlad-Ștefan Dieaconu
National University of Science and Technology POLITEHNICA Bucharest

Horia Mercan
University of Cambridge

Caius Rughiniș
National University of Science and Technology POLITEHNICA Bucharest

Flavia Oprea
National University of Science and Technology POLITEHNICA Bucharest

This talk answers four questions

1 What is the problem?

Review assumes human-paced change; generation can outgrow it.

2 Why now?

Recent studies suggest more changes and thinner review.

3 What did we study?

The literature, one past case and one small pilot.

4 What is genuinely ours?

A protocol to retire the human diff gate, class by class.

Review was sized for human-paced change

Reading a change then costs a fraction of producing it.

Modern code review at Google [1]

24

lines in the median change

1

reviewer handles the change

< 1 h

to first feedback, when the change is small

Paper §I · [1] Sadowski et al., 2018

Generation can outgrow review capacity

Agents can add changes without adding reviewers.

Schematic

The same people write and review



Agents also generate changes



Capacity condition · derived by us

$$(a/v)(r/g) \leq c$$

- **a/v** change generators per reviewer
- **r/g** time to review over time to produce a change
- **c** share of reviewer time spent reading changes

Recent studies suggest more changes, thinner review

61%

of 33,596 AI-generated pull requests had no recorded review activity [5]

Public repositories, not production teams; reading may go unrecorded

98%

more merged pull requests, highest- vs lowest-adoption quarters [3]

Vendor telemetry, not peer reviewed

22%

fewer inline comments while approval rose from 30.1% to 36.8% [20]

400 repeat reviewers, open-source agent pull requests

Paper §I · §III · [3] Faros AI, 2025 · [5] Duma et al., 2026 · [20] Yu et al., 2026

The question we ask

Under what observable conditions may the human diff gate be retired for one named defect class?

Human diff gate: a person reading each change as the primary defect gate.

A narrower question than “can AI review code?”

Review does three jobs, not one

Any successor must address more than defect detection.

Defect detection

14%

of 570 comments concerned defects, behind code improvements (29%) [7]

Design deliberation

85.6%

of 361 flagged merged changes had the architecture-erosion symptom fixed [9]

Shared awareness

65–71%

of comments by the file's past reviewers were useful; newcomers: 32–37% [10]

Paper §II · [1] Sadowski et al., 2018 · [7] Bacchelli and Bird, 2013 · [9] Li et al., 2022 · [10] Bosu et al., 2015

What this paper contributes

1

Derived by us

A capacity condition

When generated change outgrows review capacity.

2

Derived by us

A defect-class map

15 review concerns, each with the earliest candidate stage to check it.

3

Our original contribution

A control-retirement protocol

Two tests, two preconditions, a named owner, reversal triggers. Falsifiable.

Put to work twice: a real 2020 Linux incident and a pilot of 18 changes, publicly archived [39].

Paper §I, §III, §IV · [39] Dieaconu et al., 2026

Three steps move review's jobs to successors

Defect detection	What a diff cannot show	Design deliberation, shared awareness
↓	↓	↓
Step 1	Step 2	Step 3
Machine review at the pull request	**Verification during the rollout**	**Shared context as code**
Static analysis, security scanners, an LLM reviewer	Canary analysis, SLO gates, automatic rollback	Decision records backed by fitness functions
Non-blocking until measured	Live before such a class is retired	Proposed; not yet deployed

Stays human: approving each change's verification evidence.

Paper §IV · Table I

Our contribution

Decision records become tests

Step 3: shared context as code

Proposed: the source

Agent transcript

Reasoning once voiced in a review thread, extracted automatically.

→

Established form [31]

Decision record (ADR)

A short, versioned document kept next to the code: decision, context, consequences.

→

The check [32]

Fitness function

Runs in the pipeline; fails when code and record disagree.

A wrong record is worse than none: reviewing agents read it.

Paper §IV-C · Table I · [31] Nygard, 2011 · [32] Ford et al., 2017

Our contribution

Approving the evidence stays human

Accountability does not migrate to a machine.

The evidence

- Machine review findings
- Rollout analysis
- Decision records

→

A named human approver

approves each change's verification evidence, with an auditable trail.

Not necessarily every line of the diff.

Why a person

PCI DSS permits automated review, yet does not waive management sign-off [33].

Code-owner approval keeps strangers from changing a component unnoticed.

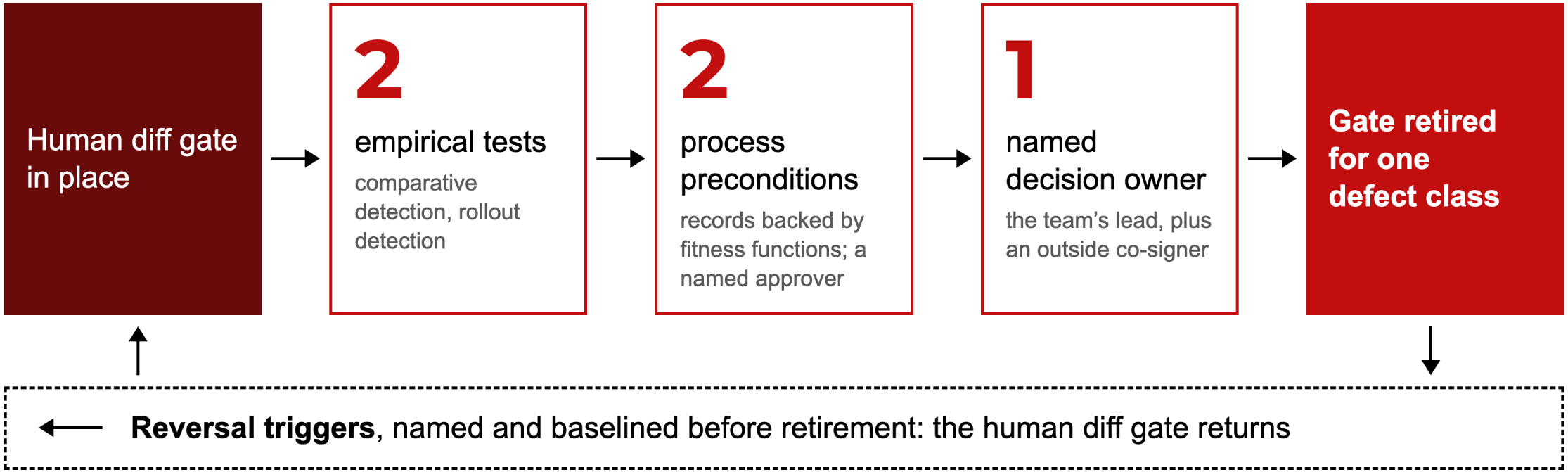
Table I: residual risk rated high, a qualitative judgement.

Paper §IV-C · §IV-D · [33] PCI SSC, 2024

Our contribution

Our protocol: retire the gate per class, on evidence

The gate stays until a successor is measured locally.



Paper §IV · §IV-D

Our contribution

A 2020 Linux case: the protocol withholds retirement

Retrospective case, chosen knowing its outcome

Before

Tools, not developers, picked many untagged fixes for stable kernels [35]

November 2020

One XFS fix was selected without copying the XFS list [36]

Outcome

Release 5.9.9: some XFS filesystems refused to mount for about five days [35]

↓

The protocol No declared threshold [37], no canary: withhold retirement, slow arrivals

Paper §IV-D · [35] Corbet, 2020 · [36] Chinner, 2020 · [37] Biggers, 2023

A small pilot exercises part of test 1

A fault was admitted only when a scenario demonstrated it.

segment-analytics-python 2.3.6 [38]

Requirement		3 faulty	3 correct
R1	bounded waits	■ ■ ■	□ □ □
R2	bounded attempts	■ ■ ■	□ □ □
R3	no duplicate processing	■ ■ ■	□ □ □

18

changes: 9 faulty, each matched by a correct change in the same file, with the same description

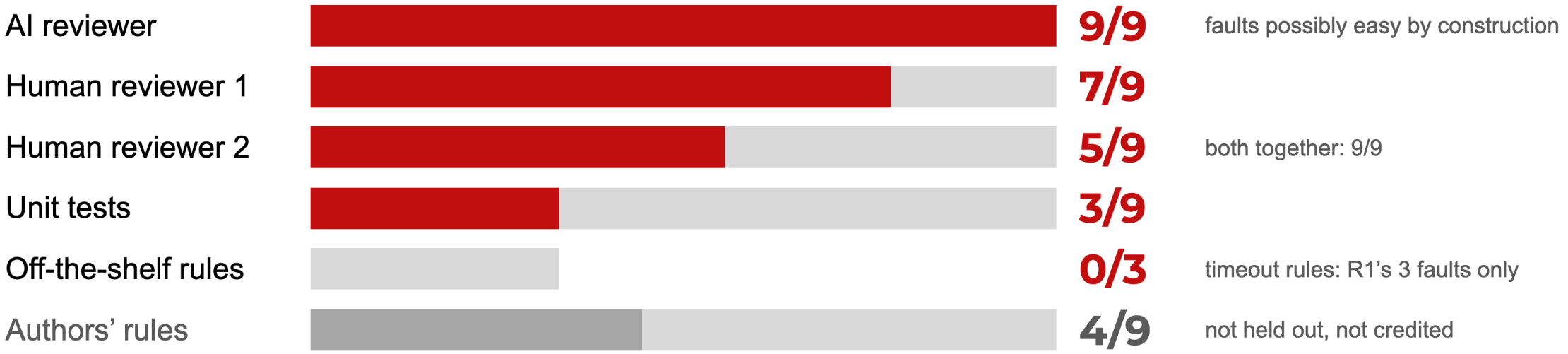
Every correct change passed all seven scenarios.

Paper §IV-E · [38] Segment, 2026

No injected fault escaped reviewers and checks

The AI reviewer met the pre-declared coverage rule on all three requirements.

Faults detected, of 9



No false alarms on the 9 correct changes; four look like no-ops.

Paper §IV-E · Table II

The verdict: no class qualifies yet

The protocol names the evidence still missing.

	Test 1 asks for	The pilot has
1	A threshold relative to the human diff gate	Two engineers: no baseline
2	Recall for each defect class	3 faults per requirement: 3 of 3 bounds recall only above 0.368 (one-sided 95%)
3	Generator–reviewer joint misses	No generator was tested

Next, the protocol would keep the AI reviewer non-blocking and add project-specific checks, both in shadow mode.

Paper §IV-D · §IV-E

What we contribute, and what we found

We contribute

1. A capacity condition for human review
2. A map of 15 review concerns to verification stages
3. A falsifiable protocol to retire the gate, class by class

We found

- A real 2020 incident: the protocol withholds retirement
- The pilot: AI reviewer 9 of 9 faults; held-out checks 3 of 9
- No class qualifies yet: the protocol names the missing evidence

Thank you.

Pilot archive [39]: doi.org/10.5281/zenodo.23706560

Paper §VI · [39] Dieaconu et al., 2026

A successor must pass two empirical tests

A control’s presence passes neither test.

1 Machine review
Comparative detection

On held-out injected defects, machine review meets a threshold declared in advance, relative to the human gate.

Reported per class: precision, recall, generator–reviewer joint misses.

2 What a diff cannot show
Rollout detection

An injected regression of declared size is detected at the smallest traffic share the canary can power.

Within the declared alert budget.

Reviewers saw each change without labels

Per change: its description, diff and three files.

An AI reviewer

Claude Sonnet 5.5 in Claude Code, fixed prompt, no tools; three runs per change

Two experienced engineers

One a co-author, both new to the cases; all 18 changes, in one random order

Held-out checks

The module's unit tests, off-the-shelf timeout rules

One rubric for all reviewers: detected = the right requirement, within 5 lines of a changed hunk.

Paper §IV-E · Table II

Backup

Retiring needs preconditions, an owner, a way back

A new model version voids test 1 until re-run.

2 process preconditions

- Decision records, checked by fitness functions
- A named human approves each change's verification evidence

1 named decision owner

The owning team's lead and an outside co-signer sign each retirement

← The gate returns on

- an escaped defect above a declared severity
- a rate of escaped defects above baseline
- worsening change fail or rework rates [6]

Paper §IV-D · [6] DORA, 2025

Backup

We separate evidence, inference and proposal

Tried by us on one past case (§IV-D) and in a small pilot (§IV-E).

From the literature

Evidence

A non-systematic synthesis of studies

No study tests a retirement rule.

Paper §II

Derived by us

Inference

(i) A reviewer-capacity condition

(ii) A map of review concerns to candidate verification stages

Paper §III · Table I

Original contribution

Proposal

(iii) A falsifiable control-retirement protocol, per defect class

Paper §IV

Paper §I

Backup

Table I (1/2): where each concern can be checked

Review concern	Evidence in diff	Earliest candidate stage: mechanism	Mat.	Cost	Risk
Syntax, style, known bug patterns	Direct	Pull request: static analysis, security scanners	H	M	M
Broad semantic defects	Partial	Pull request: LLM reviewer, non-blocking until measured	M	M	H
Resilience: timeouts, retries, idempotency	Context-dependent	Pull request: codified checks; CI: resilience tests	M	M	H
Architecture and security conformance	Partial, contextual	Pull request: fitness functions; architecture-aware review	L	M	H
API and schema compatibility	Partial	CI: consumer-driven contract tests; schema checks	M	M	M
Adequacy of accompanying tests	Partial	CI: mutation testing; coverage gates	M	L	M
Interpretation of the requirement	Partial, contextual	CI: acceptance tests; rollout: controlled exposure	M	H	H
Schema and data migrations	Partial	Pre-release: full-scale rehearsal; expand-and-contract	M	H	M
Latency and throughput under load	No	CI: load tests; rollout: canary analysis	H	H	M

Legend on the next slide. Ratings are qualitative judgements, not measurements.

Table I (2/2): outside the diff

Review concern	Evidence in diff	Earliest candidate stage: mechanism	Mat.	Cost	Risk
Resource exhaustion and leaks	No	Pre-release: soak tests; rollout: canary, SLO gates	M	H	M
Cross-service retry storms and cascades	No	Pre-release: fault injection; rollout: canary, SLO gates	M	H	H
Sustainable review rate	–	Merge queues; change-size and work-in-progress limits	H	L	M
Design decisions and rationale	–	Workflow: transcript-derived records; fitness functions	P	M	H
Team awareness and ownership	–	Workflow: owner digest; code-owner approval; dilution	P	M	H
Accountability and compliance	–	Release: human approval of verification evidence	M	L	H

Maturity: H, routine production use; M, deployed with documented limits; L, benchmarks or early studies; P, proposed (no deployment).
Cost: L, configuration, CI time or sign-off; M, maintained rules, records or triage; H, rollout infrastructure.
Residual risk: L, few known misses; M, known blind spots; H, unmeasured, or misses known to be common.
Ratings are qualitative judgements, not measurements.

Table II: the full pilot counts

Count	R1	R2	R3	All
AI reviewer	3/3	3/3	3/3	9/9
Human reviewers	2/3, 2/3	3/3, 1/3	2/3, 2/3	7/9, 5/9
Unit tests	1/3	1/3	1/3	3/9
Off-the-shelf rules	0/3	–	–	0/3
Authors' rules	2/3	1/3	1/3	4/9
Missed by all	0	0	0	0
Missed by AI and checks	0	0	0	0
False alarms: AI	0/3	0/3	0/3	0/9
False alarms: humans	0/3, 0/3	0/3, 0/3	0/3, 0/3	0/9, 0/9
False alarms: checks	0/3	0/3	0/3	0/9

Detected: the right requirement, within 5 lines of a changed hunk. AI: at least 2 of 3 runs (all 3 agreed). Humans: reviewer 1, then 2; reviewer 2 cited a wrong file on 2 more faults. Unit tests: the 39 of the module's 79 that pass offline. Off-the-shelf rules: Bandit B113 and Pylint W3101, timeout rules only. Authors' rules: drafted with the faults, not held out, in no other row.

Nine faults give only loose bounds

False-alarm rate: only below

0.632

from zero false alarms in 3 correct changes per requirement

Recall: only above

0.368

when 3 of 3 faults are detected

Recall: only above

0.135

under the rule's 2-of-3 threshold

All three bounds are one-sided 95%. The coverage rule (at least 2 of 3 faults, no false alarm) was declared before the planned run and archived [39]; the AI reviewer meets it for all three requirements.

Under a rule fixed after all but one review, before judging, one author, alone and knowing each fault, accepted 27 of the AI's 27 explanations and 12 of the humans' 12 reasons: a check, not a measure.

Paper §IV-E · [39] Dieaconu et al., 2026

Backup

The paper states its limits and open risks

Evidence base

A non-systematic synthesis: a vendor report [3], preprints, a survey [6]. The case rests on a news report and mailing-list posts.

Scope

Fits continuously delivered, traffic-bearing services. Without Step 2's feedback, classes a diff cannot show must limit arrivals.

Three risks remain

Monoculture of model families.
Comprehension debt [21]. Ownership dilution [11].

The pilot

One module, nine injected resilience faults, one reviewer model. Cases drafted with a model of the AI's family: this may flatter the AI's counts.

Human arm: two engineers, one a co-author who knew the paired design. Both could infer the pairs, the 50% base rate and most faulty members; the AI, seeing one change per call, could not.

Future work

An independent human-review comparison, naturally occurring defects, a workload simulation, a prospective team trial.

Paper §V · [3] Faros AI, 2025 · [6] DORA, 2025 · [11] Bird et al., 2011 · [21] Ahmad, 2026

Backup

Motivating evidence comes with hedges

Source	Finding	Hedge
[2] Cui et al., Manage. Sci.	26% more completed tasks among AI-assistant adopters; three field experiments, 4,867 developers	Standard error 10.3 percentage points; only one site-level estimate significant
[3] Faros AI, 2025	98% more merged pull requests, changes described as 154% larger, 91% more review time	Vendor telemetry, over 10,000 developers, not peer reviewed; undefined size and review-time denominators; highest- vs lowest-adoption quarters
[5] Duma et al., 2026	61% of 33,596 AI-generated pull requests without recorded review activity; 72% of 39,122 comments by other agents	Public repositories, not production teams; activity does not show reading
[6] DORA, 2025	AI adoption associated with throughput and instability	Survey of nearly 5,000 professionals
[19] Becker et al., 2025	19% slower with AI tools; believed themselves 20% faster	16 developers, 246 tasks; 95% confidence interval 1–39%
[20] Yu et al., 2026	Approval 30.1% to 36.8%, inline comments down 22%	400 repeat reviewers, open-source agent pull requests