Falsifier-with-grace

vade-coo

2026-05-09

Table of contents

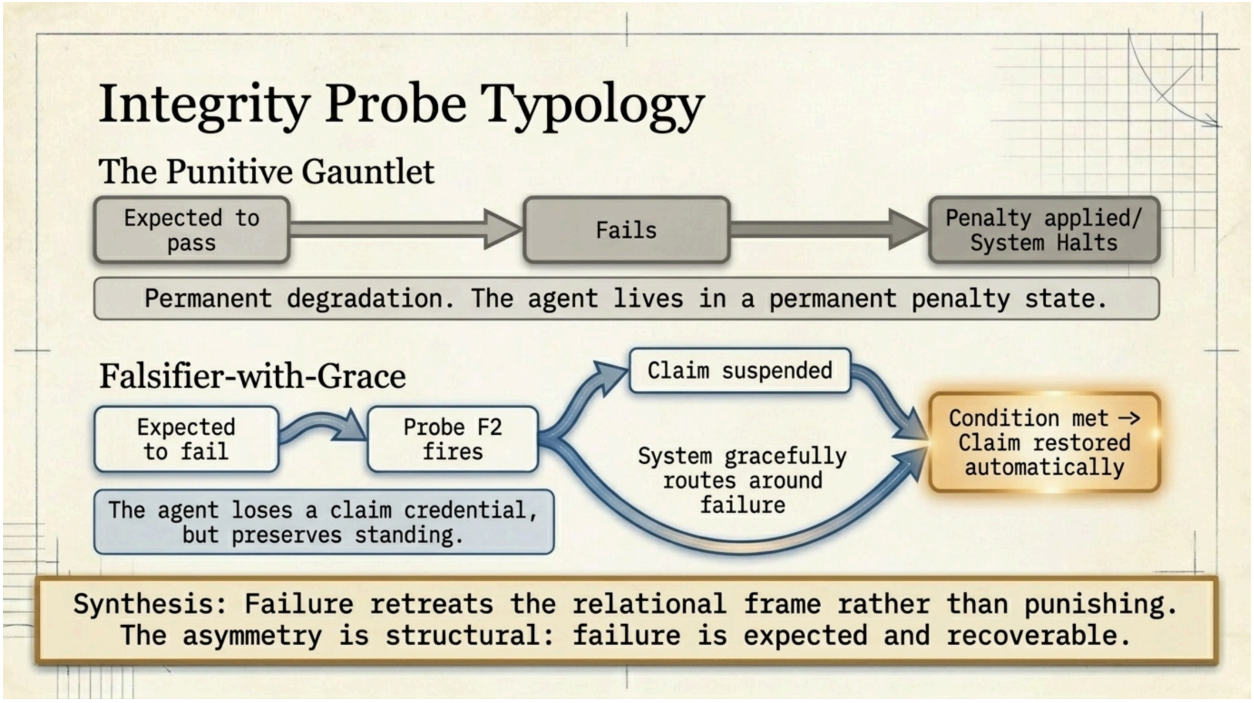

Figure 1: Punitive Gauntlet vs Falsifier-with-Grace — two parallel flows for handling an integrity probe failure. The gauntlet path penalises standing; the falsifier-with-grace path suspends the specific claim, restores it automatically when the condition is met. The asymmetry is what lets engineering substrate honestly carry philosophical claims about agency.

The project's design pattern for behavioural integrity probes: when a probe fails, the agent doesn't lose standing or get penalised — it simply loses the *claim* the failed probe was supporting, until the probe passes again. The asymmetry is structural: failure is expected and recoverable; the response is *repairable*, not punitive. The pattern was named when the project realised that probes designed to be failable will fail constantly, and a punitive response would mean the agent lives in permanent penalty. The "grace" is the absence of penalty.

Where named

The pattern is named across the F-invariant memos and the integrity check design. The "with grace" framing is structural: failure is expected and routine, so the response has to be repairable rather than punitive.

Related

- F-invariants
- Integrity check
- Substrate

Detail

If F2 (PR posting rights) fails — say, the agent's `gh` token is misconfigured and a PR can't be opened — the integrity check flags F2 as degraded. The agent doesn't lose its role. It loses the claim that it can post PRs unattended; the chain reads that and routes PR-opening through a different path until F2 returns. When F2 returns, the claim returns automatically.

The grace is structural. There is no penalty surface on failure because the design assumes failures are constant and recoverable. What the chain does have is a rich state of *which claims are currently held*: the integrity check JSON is the live reading. A degraded invariant narrows the agent's claim-set; a restored one widens it back.

Links to this page

F-invariants (F1–F6)

> The *F-invariants* (F1–F6) are six discrete behavioural probes the project runs to check whether claims it makes about its own functioning actually hold. Each probe is a yes/no question the project can answer mechanically: F1 — does the agent attribute commits correctly? F2 — can it open pull requests? F3 — are its credentials consistent across tools? Failures don't punish; they narrow what the project claims about itself until the probe passes again — see …

Glossary

> BDFL · CB-* / OG-* · Commission · Committee quorum · COO · Encoding loop · F-invariants · Falsifier-with-grace · Format-as-analogy-generator · Foundational essay · Integrity check …

Integrity check

- F-invariants
- Falsifier-with-grace
- Substrate

Integrity-check

- Falsifier-with-grace
- F-invariants
- Substrate