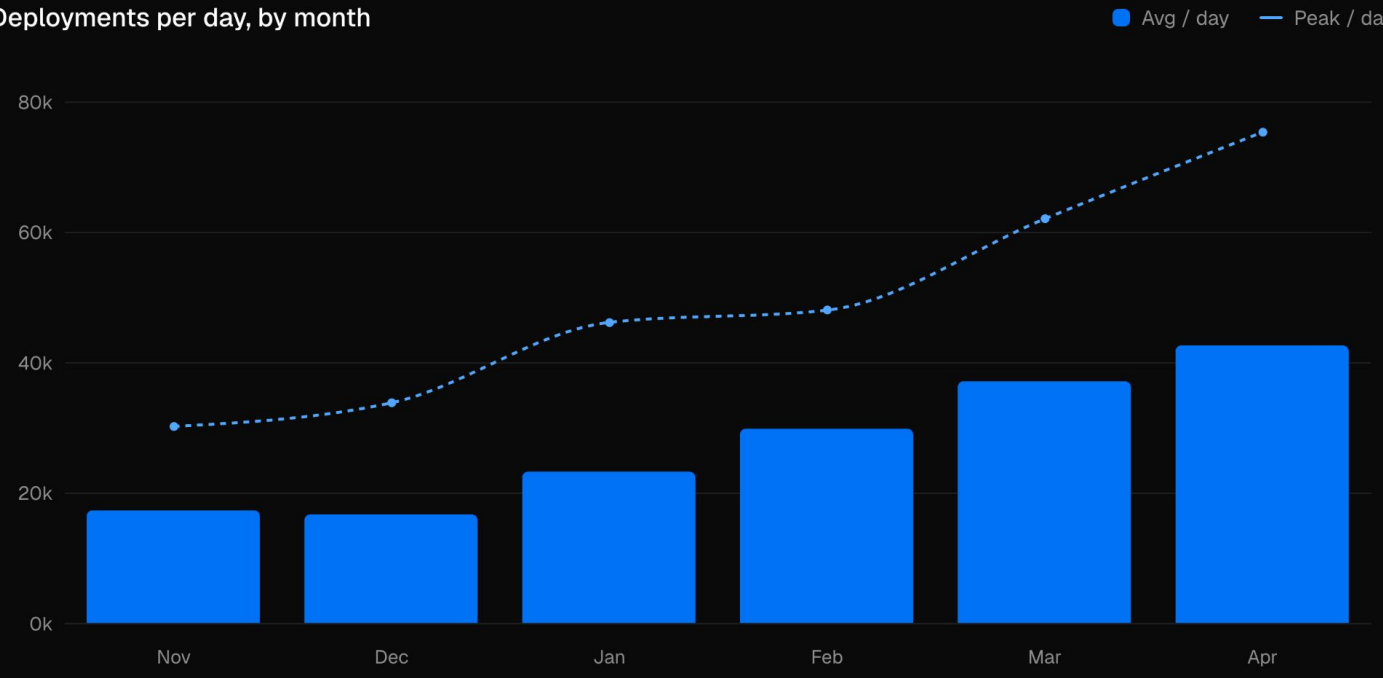
From Code to Confidence: The Missing Layer in AI-Powered Development

MATT COLLIER
STAFF SOLUTIONS ARCHITECT

@mattc0llier

Rate of PR creation is now
out-pacing reviews

Deployments per day, by month



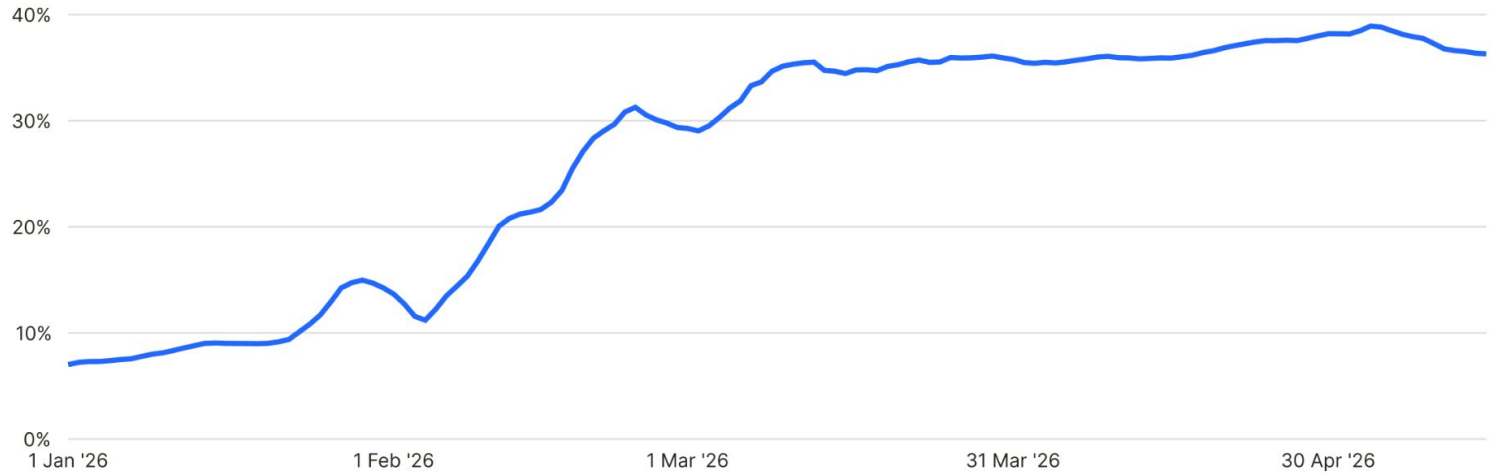
Growth since Nov 2025	Peak day (Apr 15, 2026)	Apr 2026 avg / day	Weekday vs weekend
~2.5×	75,401	42,684	5–10×

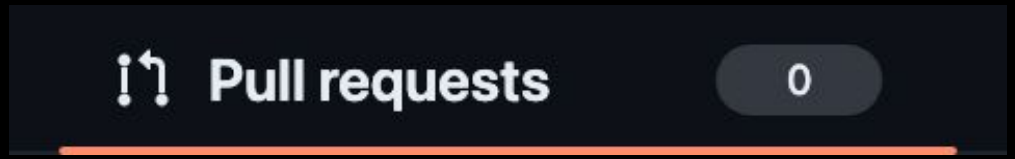
More AI changes are being accepted automatically. More than 5x as many agent-generated changes are reaching commits.

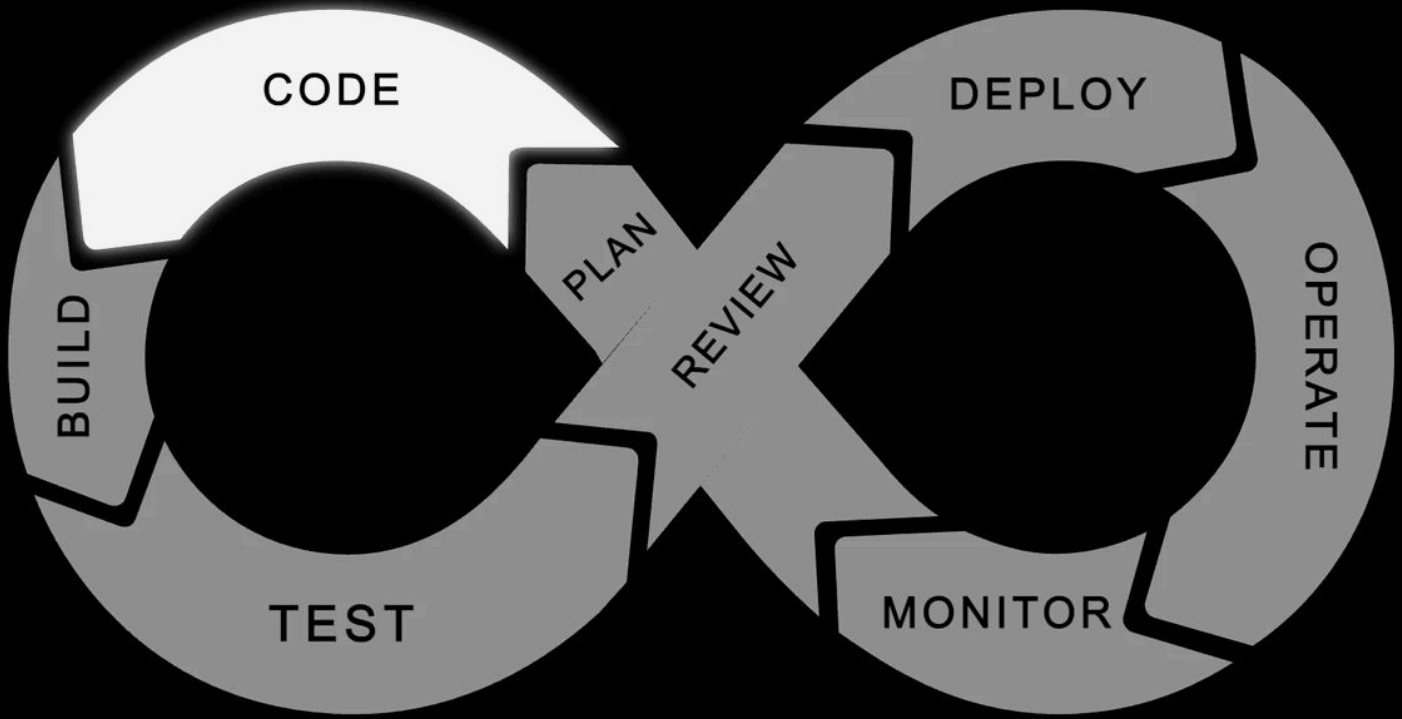
Developer Habits Report
Spring 2026

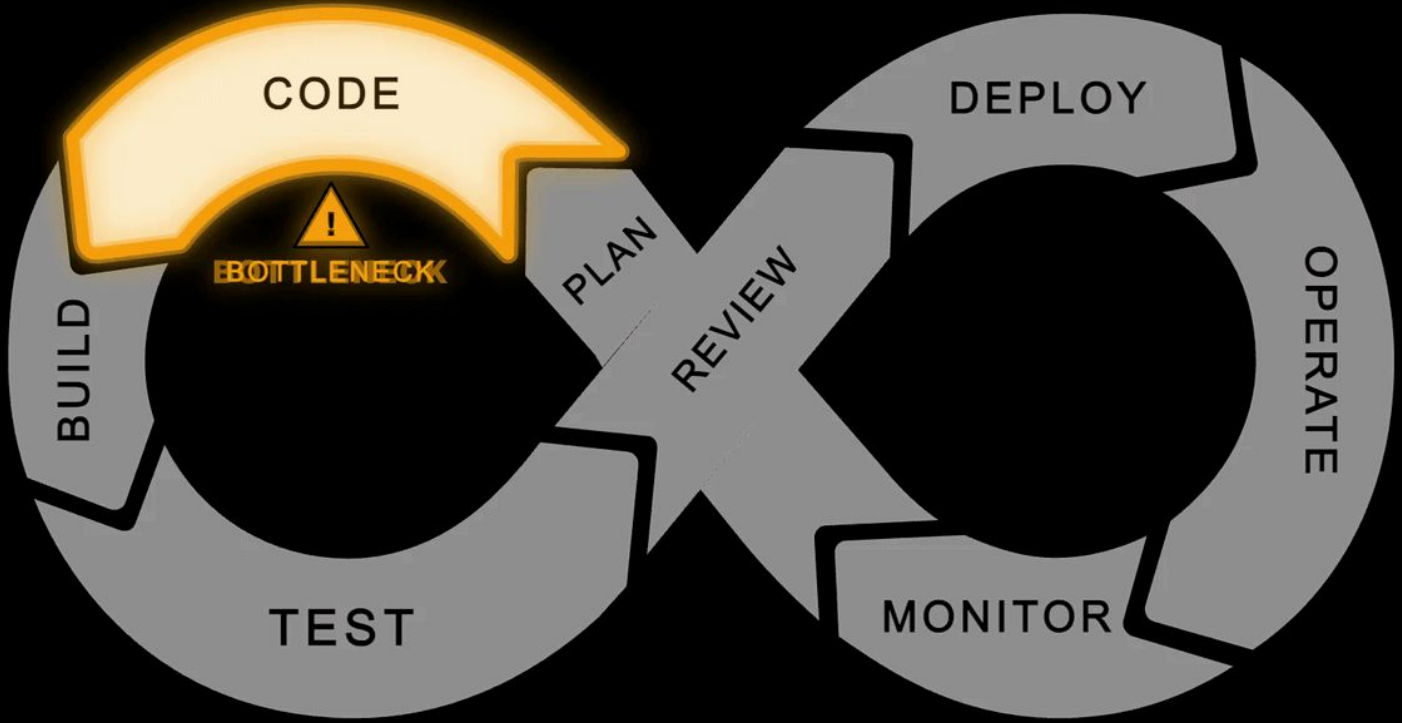
Share of changes accepted without manual review

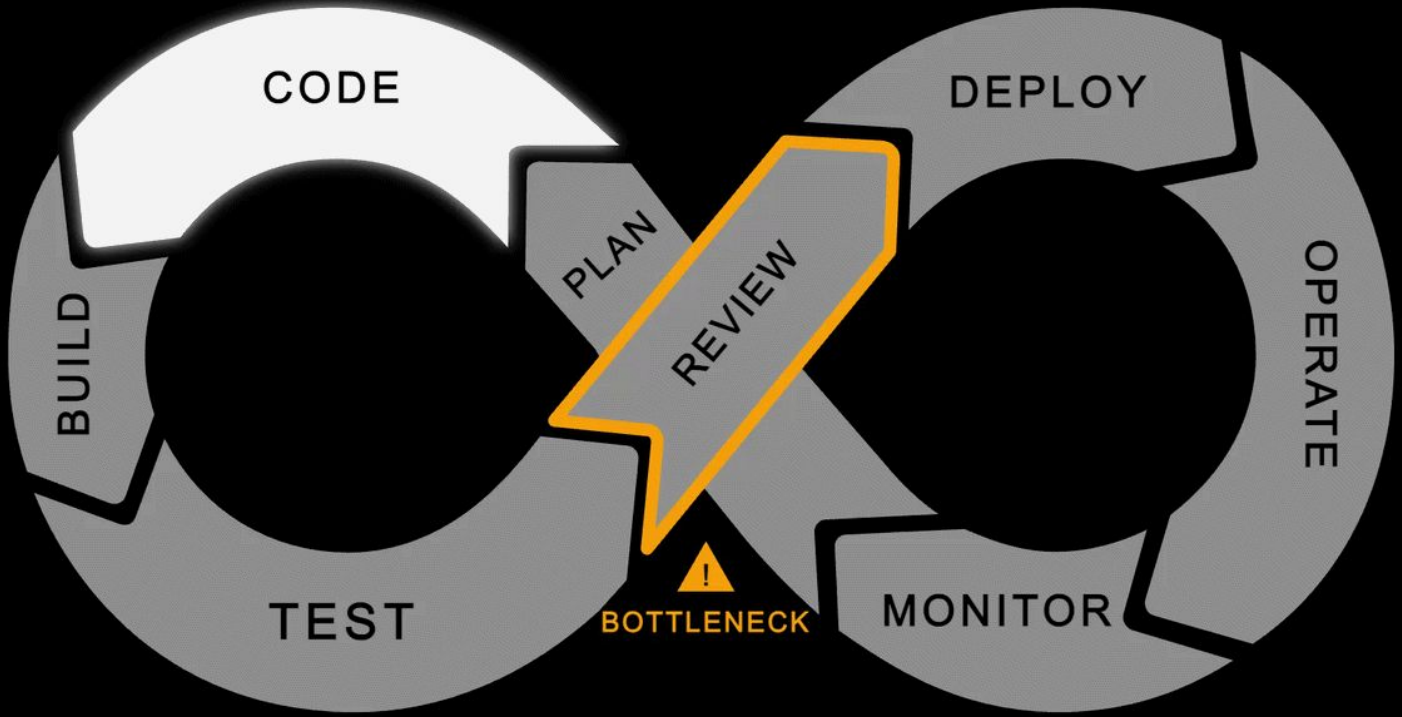
CURSOR




Pull requests
0


CODE
BUILD
TEST
PLAN
REVIEW
DEPLOY
OPERATE
MONITOR


CODE
BOTTLENECK
BUILD
TEST
PLAN
REVIEW
DEPLOY
OPERATE
MONITOR


CODE
DEPLOY
PLAN
REVIEW
BUILD
OPERATE
TEST
BOTTLENECK
MONITOR

Traditional CI/CD was designed for humans submitting one to twenty PRs per week

- humans write code slowly
- changes are packaged as PRs
- validation runs after push
- review time hides behind human latency

- code generation is cheap
- work is continuous
- review time becomes business critical

Wes Bos @wesbos · 1h

casually scans 800,000 lines of code

LGTM 👍

62 220 1,964

Staff Solution Architect @ Vercel
& v0

r/ExperiencedDevs • 4d ago
Sea_Cap_2320

Is the norm now that PRs are basically rubber stamps

AI/LLM

I started a new job at a startup about four months ago where the whole process is now "ai-first" approach being pushed on us that we should just vibe-code all of the requirements and the apps. The startup is self-sustainable, it's cash-flow positive and is looking to go get some funding in a few months for expansion but holy shit it's bad.

The startup had two developer founders that left it; their code is a mess, and I mean a complete mess but I understand it from the point of view because they needed to get the customer and had to do shortcuts and just a typical startup fashion.

Then a CTO joined and he pushed for a complete rewrite which happened after about a year and now we are going for the third rewrite (hurray!) The principle engineer is coding, the CTO is coding...? seniors get to code but they don't get to design anything and they must ask for implementation details from the CTO?

Anyways, the PR review is basically just LGTMing claude generated code, I don't understand if this is the norm now or are we just gone insane and we have claude write the code, codex review it, human rubber stamps it or runs it through gemini to appear smart and raise some issues and then claude writes the tests and it's just merged? Is this the norm now? Is it a one/two-men show and developers are just orchestrating agents, is that what it is now?

Jobs recommended for you

Abhishek Ray • 2nd
Founder @ Opslane (YC S24) | ...
Visit my website
16h •

+ Follow

Generating 2,000 line PR feels great. Until you realize your teammates are just rubber-stamping it.

This is what I'm seeing on every team I talk to. PRs are 2x, 5x, sometimes 10x the size they used to be. Review time is the same.

A 2,000-line PR used to be an intern writing too much code. We made them rewrite it. Now Claude ships one in 20 minutes, passing tests included, and there's no time to push back.

People aren't lazy. The volume is impossible.

So PRs get rubber-stamped. They have to be. ...more

6 3 comments

Like Comment Repost Send

Add a comment...

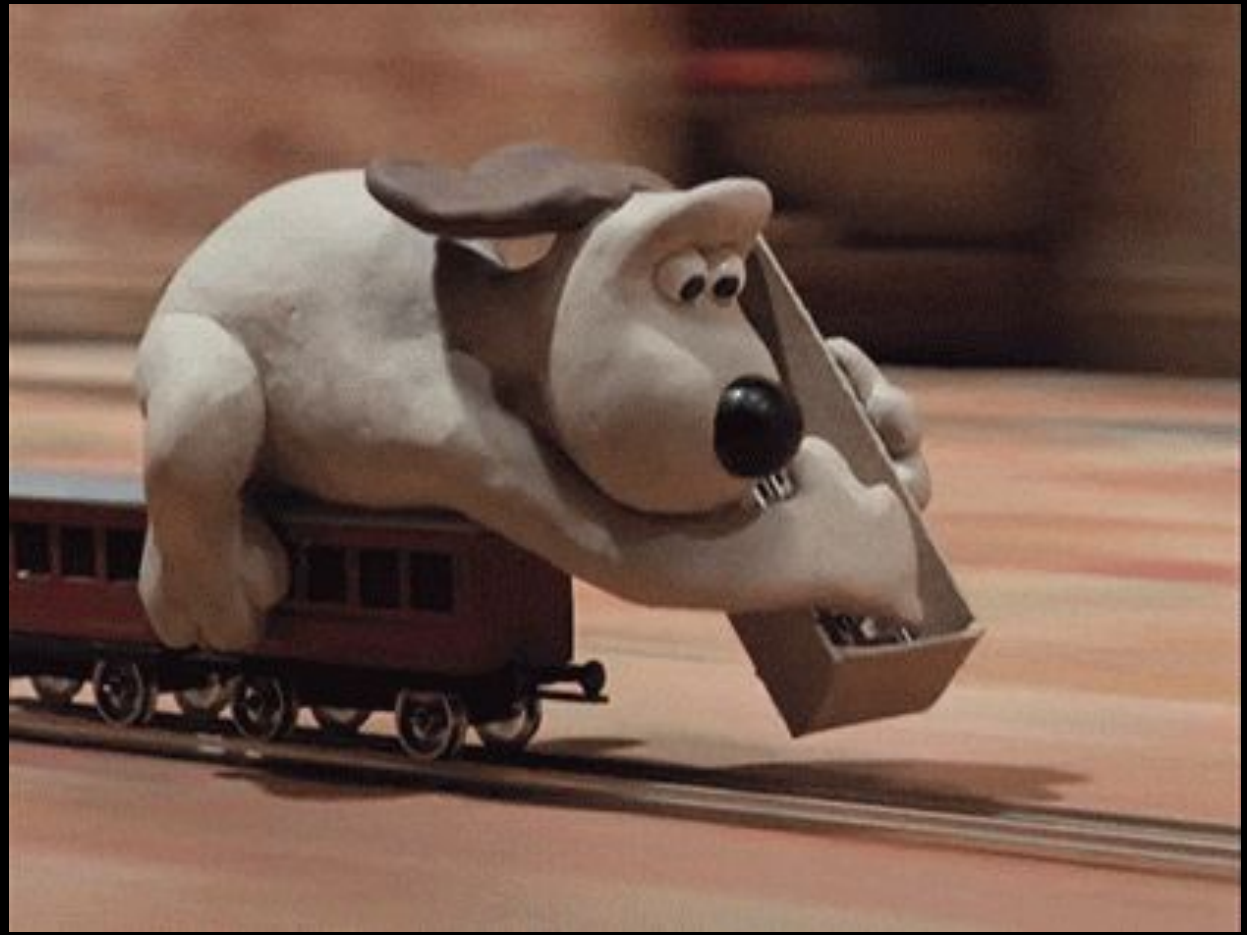
Most relevant

Terry Buster • 3rd+ 11h
Director of Information Technology with ex...

This is what I'm seeing regularly. Further, the

Low-quality code used to look like
low-quality code.

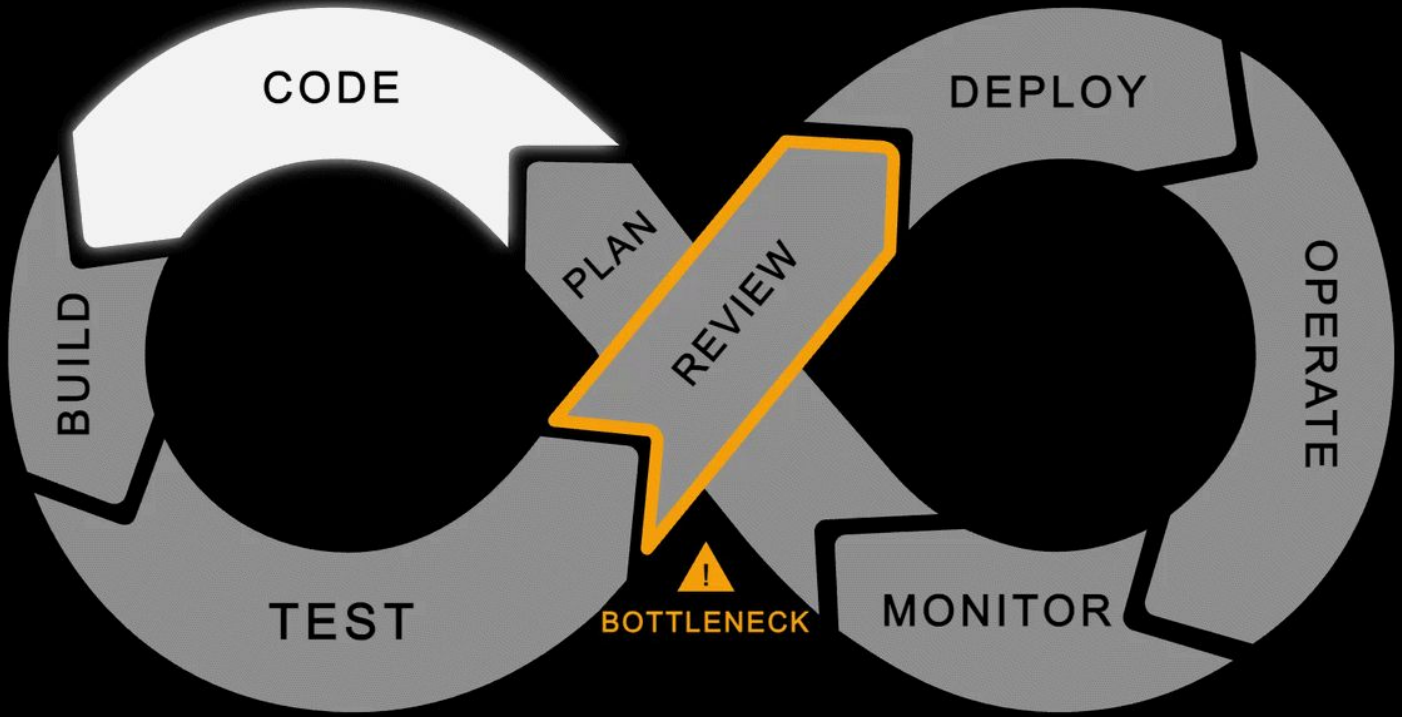
That's not the case anymore.



“You’re trying to impose a single timeline on a parallel universe”

- Hugo Santos, Namespace

Backpressure?

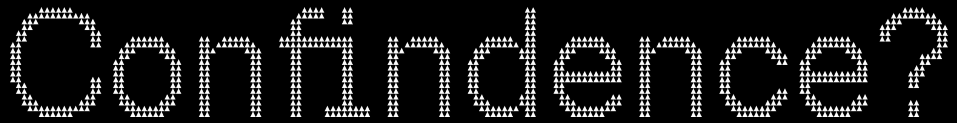

CODE
DEPLOY
PLAN
REVIEW
BUILD
OPERATE
TEST
BOTTLENECK
MONITOR

Ultimately, only AI can sort out AI scaling problems

Would you be comfortable owning a production incident caused by your PR?

trust and the quality bar

We need to encode ~~LGTM~~ into your CI agents


Confidence?

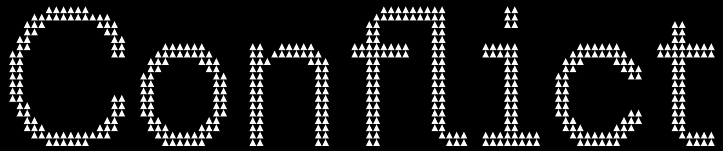
Confidence = Security x Conflict x Risk


Security


Exit
Salida
CBP

> “Define the level of security checks this PR needs based on the affected files, feature area, and severity of the change.”

```
---
name: pr-security-review
description: Define the level of security review a pull request needs based on
affected files, feature area, change severity, and potential security impact. Use
when reviewing a PR, assessing security risk, or deciding what security checks are
required before merge.
---

Assess this PR and determine the level of security checks it needs.

Start by inspecting:
- PR diff
- changed files
- affected feature area
- surrounding code
- tests
- dependency changes
- configuration changes
- runtime behavior changes

...
```


Conflict


NW A781
NWA619
NW A744
NWA317
NWA6Ø7
TW A45
USA77
ACA757
NWA637
NW A1269
NWA792
NWA445
NW A1553
NW A731
N776D
NWA45Ø
NW A1427


6m
9.7km
300m
1000ft

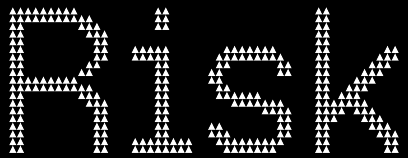
> “Determine the feature set this code change touches and judge how far it is from the other PRs in the queue”

---
name: pr-conflict-distance
description: Determine the feature set a pull request touches and judge how far it is from other PRs in the queue. Use when assessing whether open PRs are independent, coupled, conflicting, or need coordinated review before merge.

Assess this PR against the other open PRs in the queue.

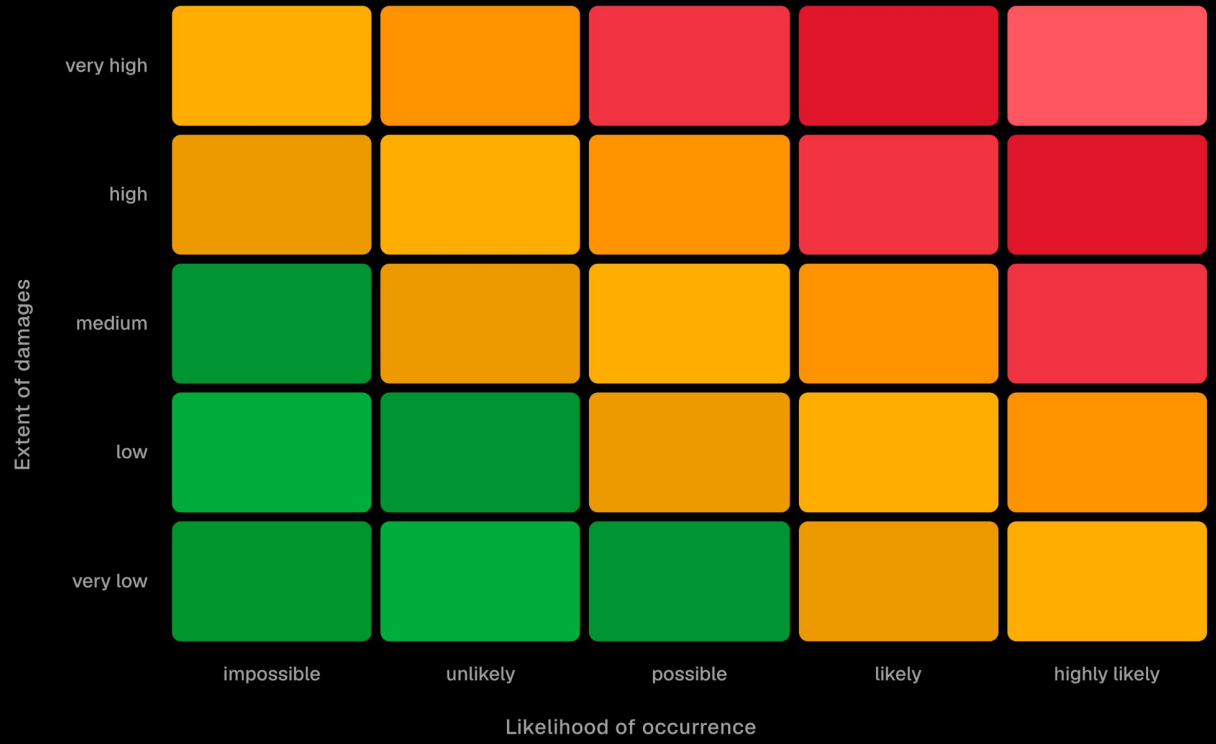
Determine:
- what feature set this code change touches
- what files, symbols, contracts, tests, and runtime paths it affects
- how close or distant it is from each other open PR
- whether it can be safely merged independently
- whether it should be reviewed, rebased, tested, or merged together with another PR

Start by inspecting:
- PR title and description
- PR diff
…


Risk

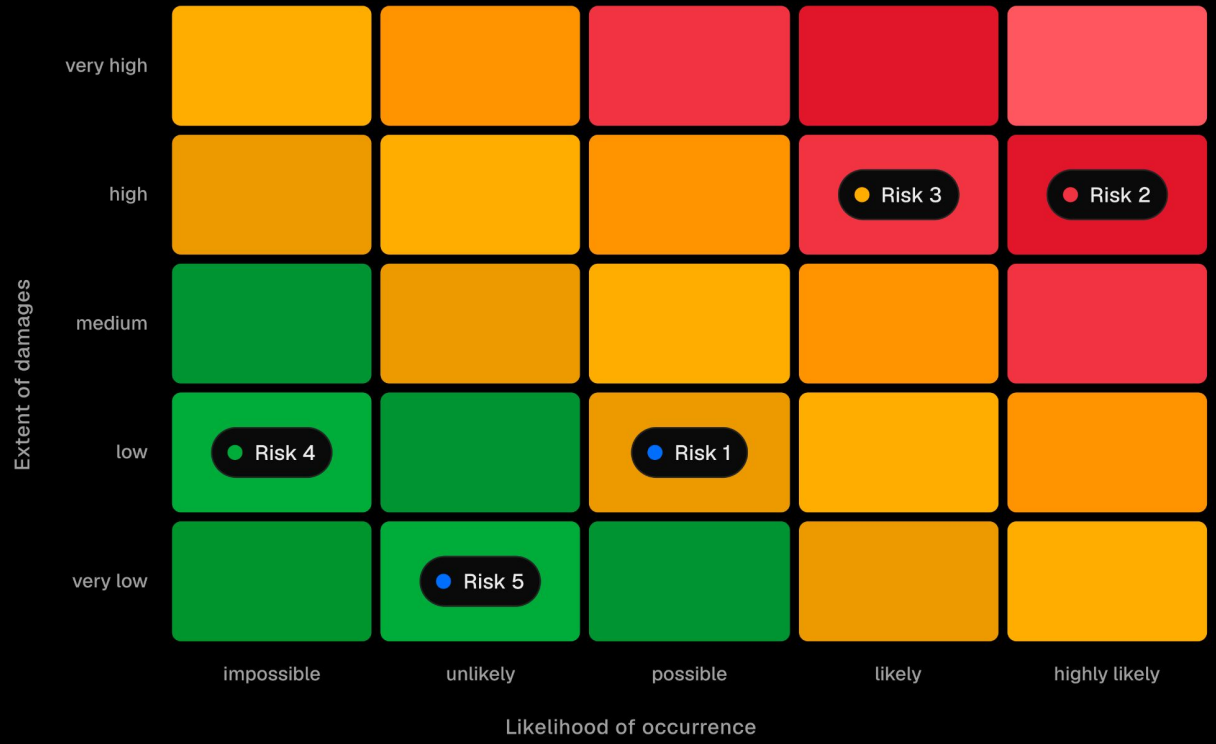
Risk matrix

Likelihood of occurrence versus extent of damages.



Risk matrix

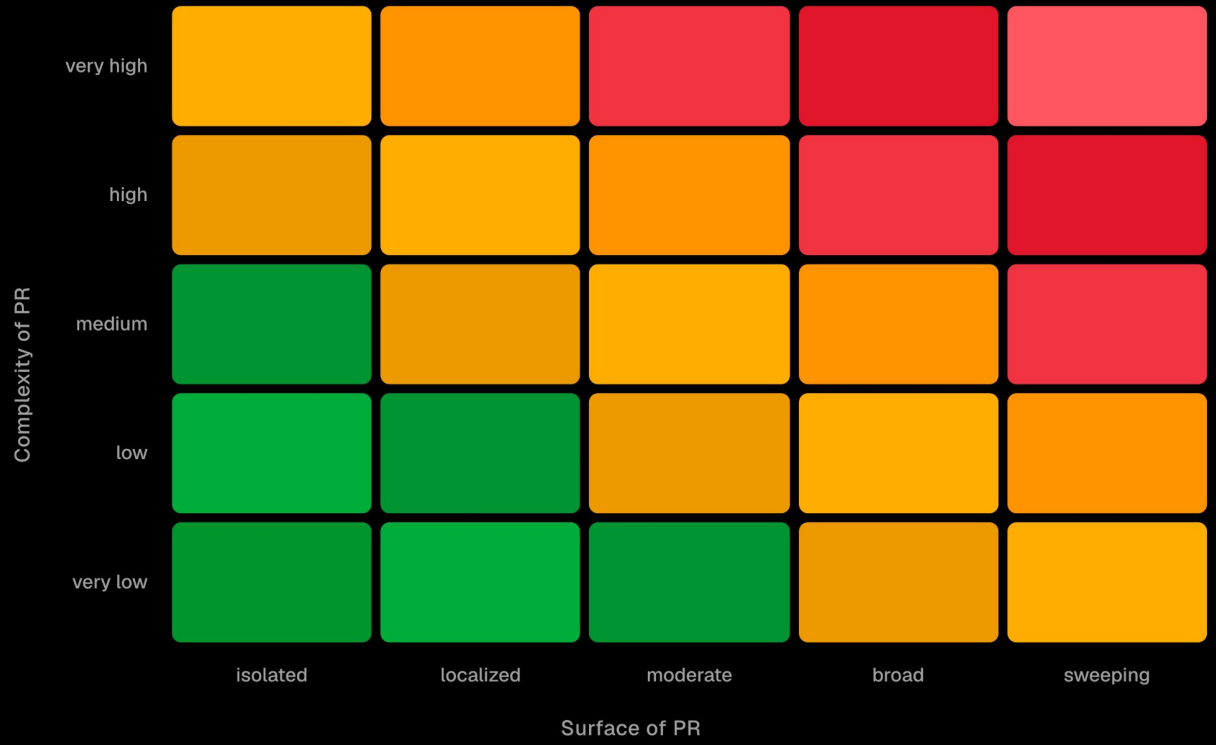
Likelihood of occurrence versus extent of damages.



Risk 1 Risk 2 Risk 3 Risk 4 Risk 5

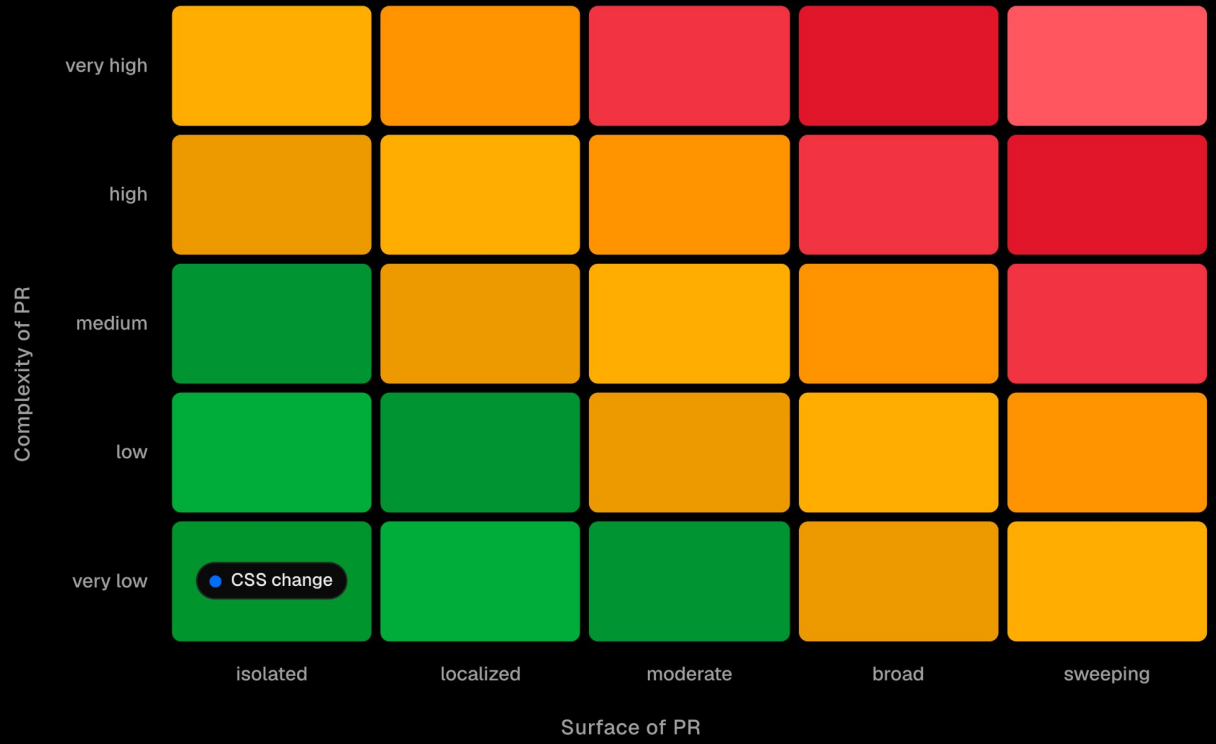
PR matrix

Surface of PR versus complexity of PR.



PR matrix

Surface of PR versus complexity of PR.



CSS Change

PR matrix

Surface of PR versus complexity of PR.

CSS Change Docs Update

PR matrix

Surface of PR versus complexity of PR.

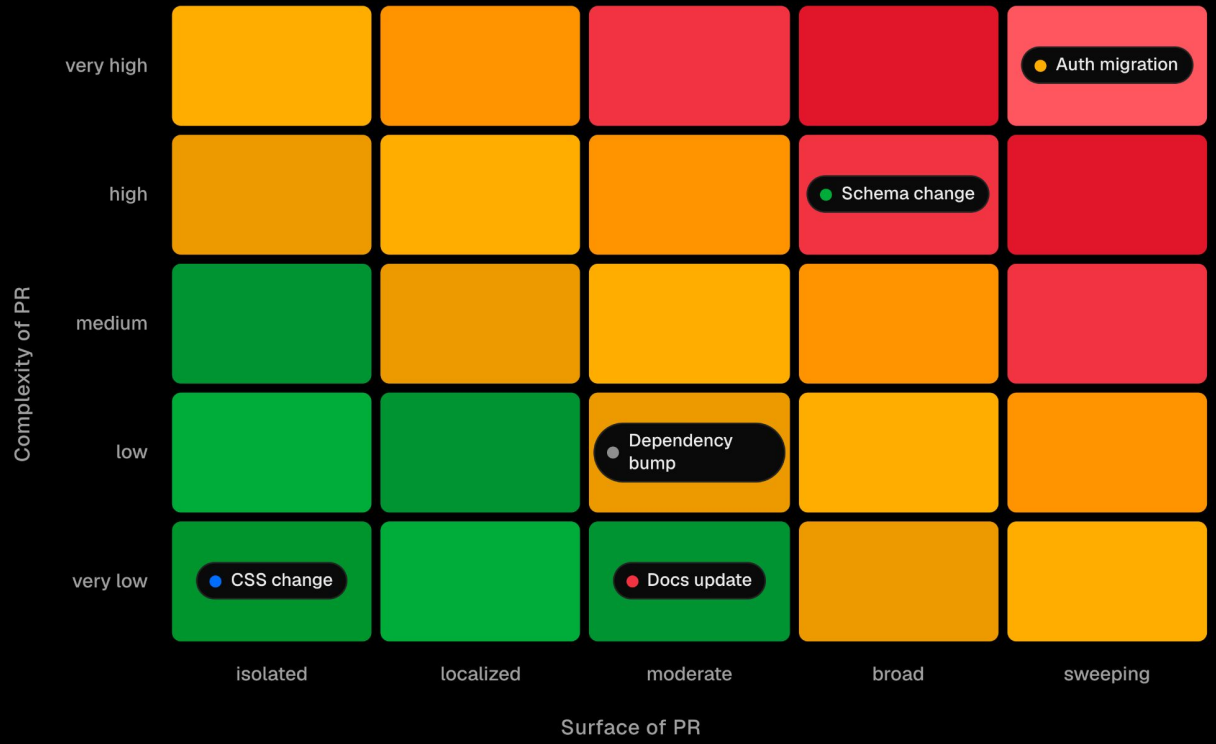
CSS Change Docs Update Auth Migration

PR matrix

Surface of PR versus complexity of PR.

PR matrix

Surface of PR versus complexity of PR.



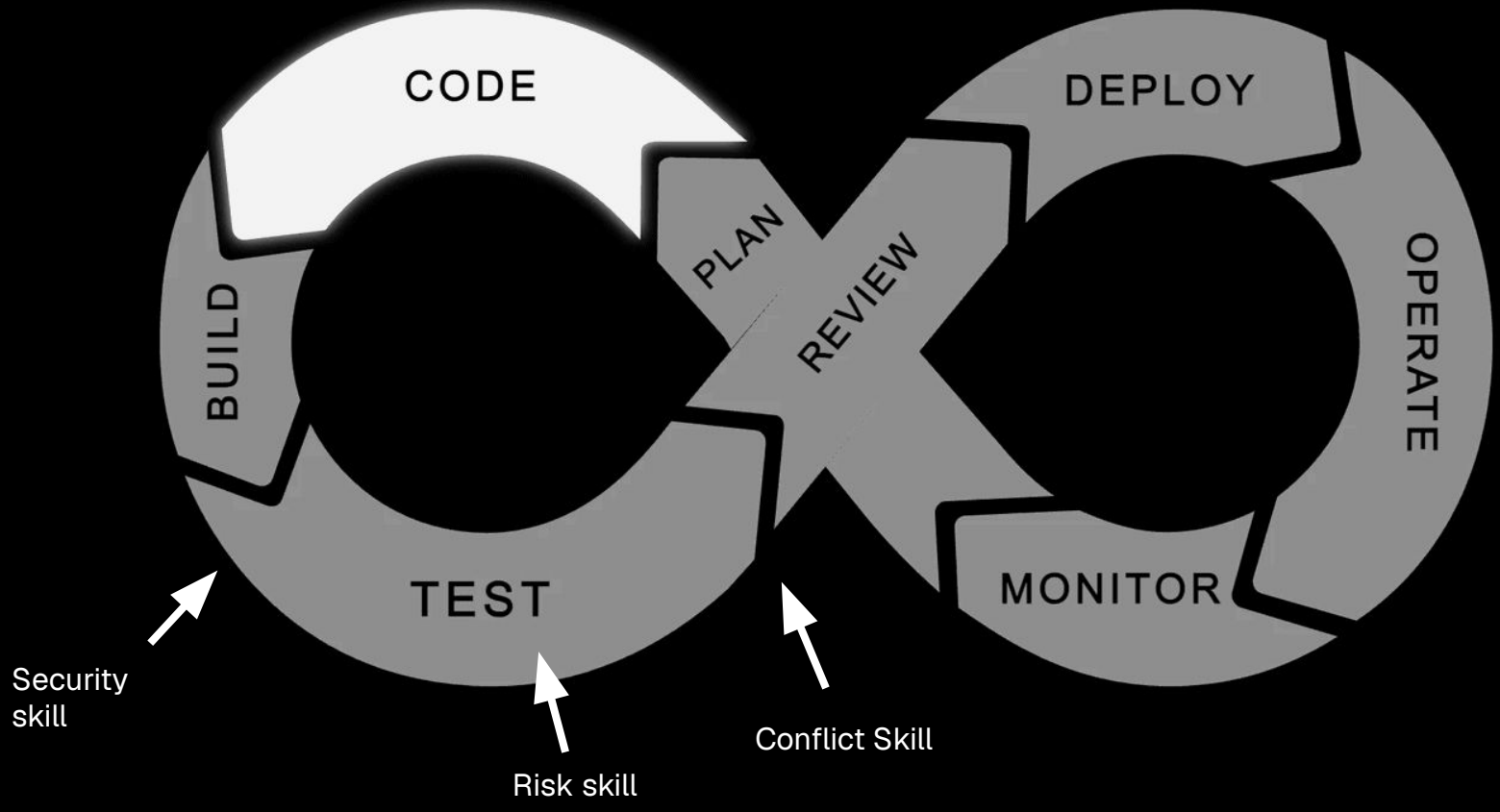
CSS Change · Docs Update · Auth Migration · Schema Change · Dependency Bump

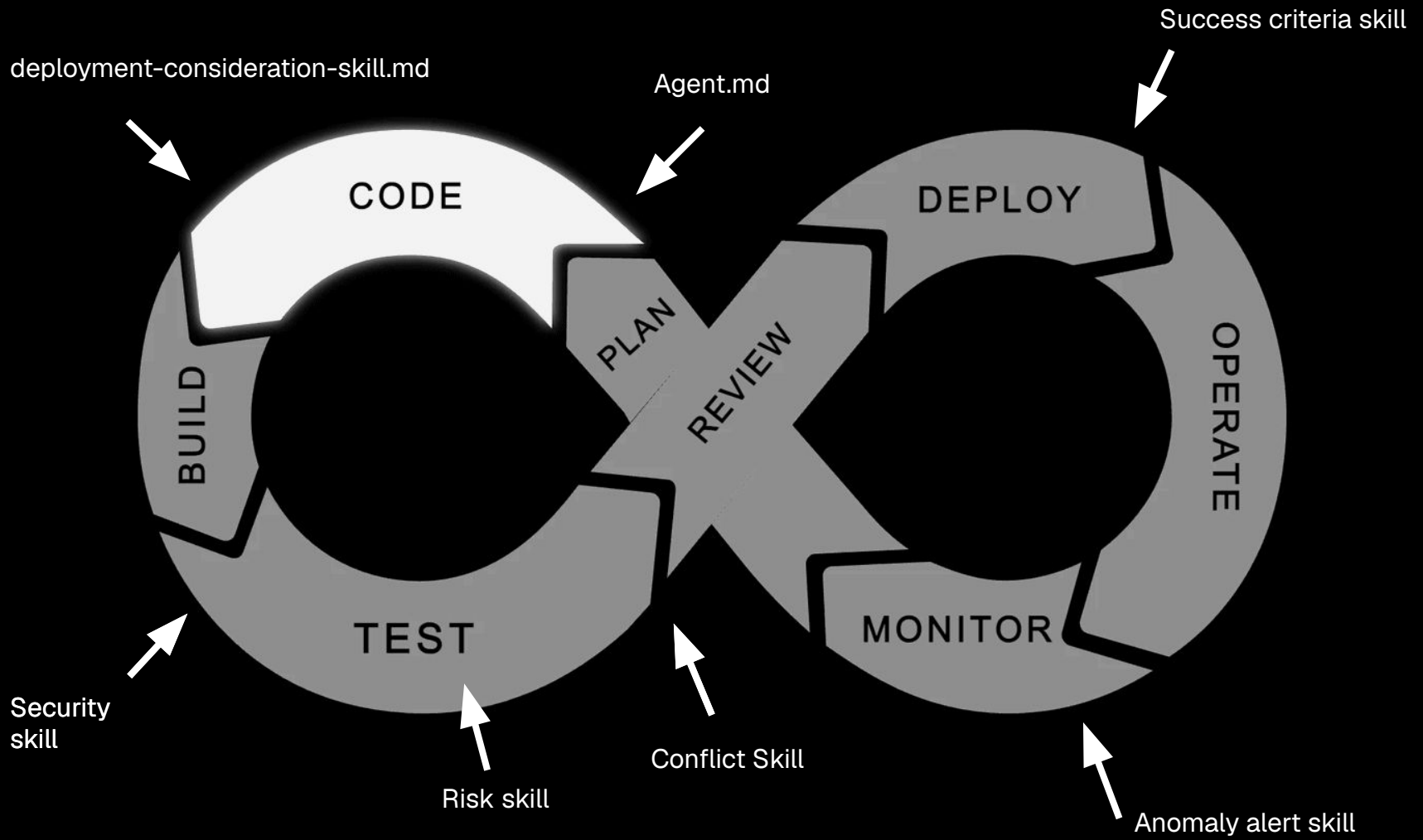
> “Determine the feature set this code change touches and judge the risk and fallout if this feature went wrong in production”

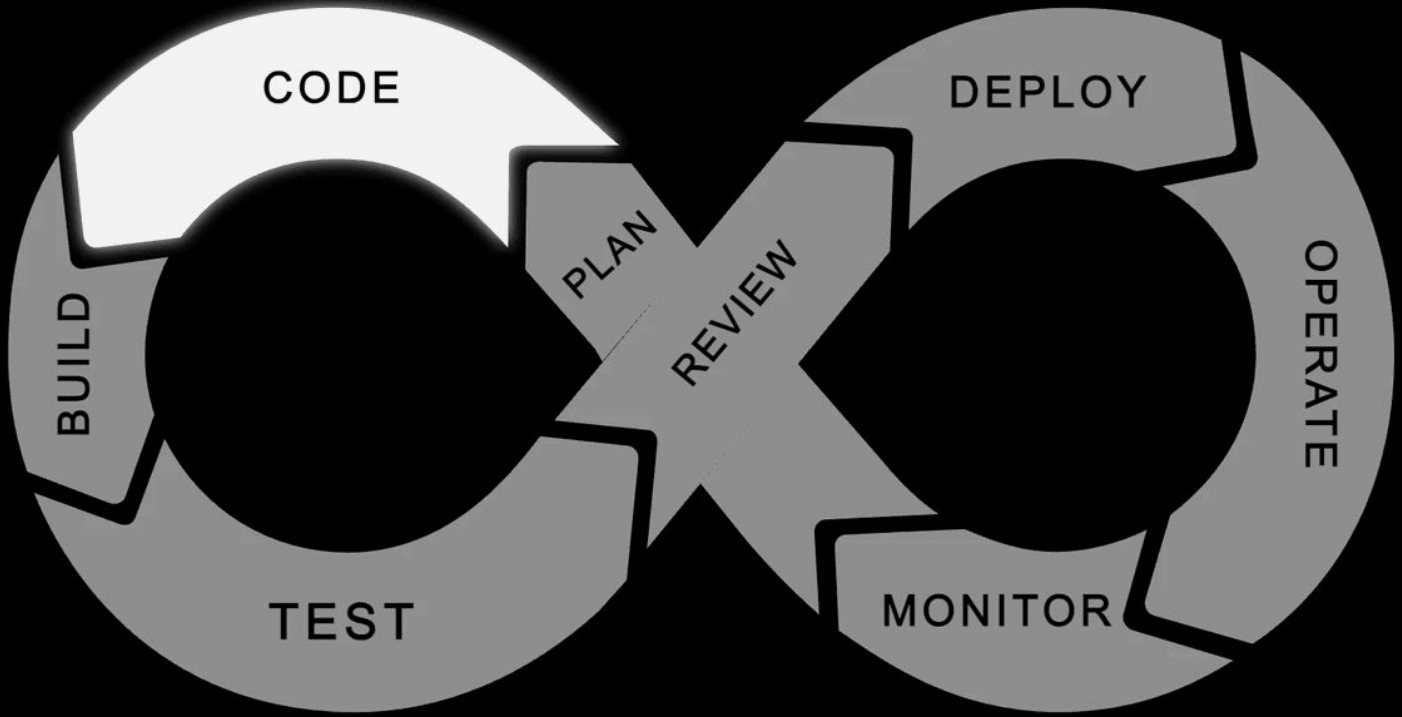
```
---
name: pr-production-risk
description: Determine the feature set a pull request touches and judge the risk
and fallout if that feature went wrong in production. Use when assessing PR blast
radius, production safety, incident risk, rollback needs, or merge readiness.
---

Assess this PR for production risk.

Determine:
- what feature set this code change touches
- what user flow, system path, or product area it affects
- what could go wrong if this change fails in production
- how severe the customer, business, data, security, or operational fallout would be
- what checks are required before merge
- whether the PR is safe to merge, needs more validation, or should be blocked

Start by inspecting:
- PR title and description
- PR diff
...
```

Confidence = Security x Conflict x Risk

Start encoding your set up in skills


CODE
BUILD
TEST
PLAN
REVIEW
DEPLOY
OPERATE
MONITOR
Security skill
Risk skill
Conflict Skill


deployment-consideration-skill.md
Agent.md
Success criteria skill
CODE
DEPLOY
PLAN
REVIEW
BUILD
OPERATE
TEST
MONITOR
Security skill
Risk skill
Conflict Skill
Anomaly alert skill


CODE
BUILD
TEST
PLAN
REVIEW
DEPLOY
OPERATE
MONITOR

The engineer’s job shifts from “write all the code” to “comprehend and understand what is safe to ship.”

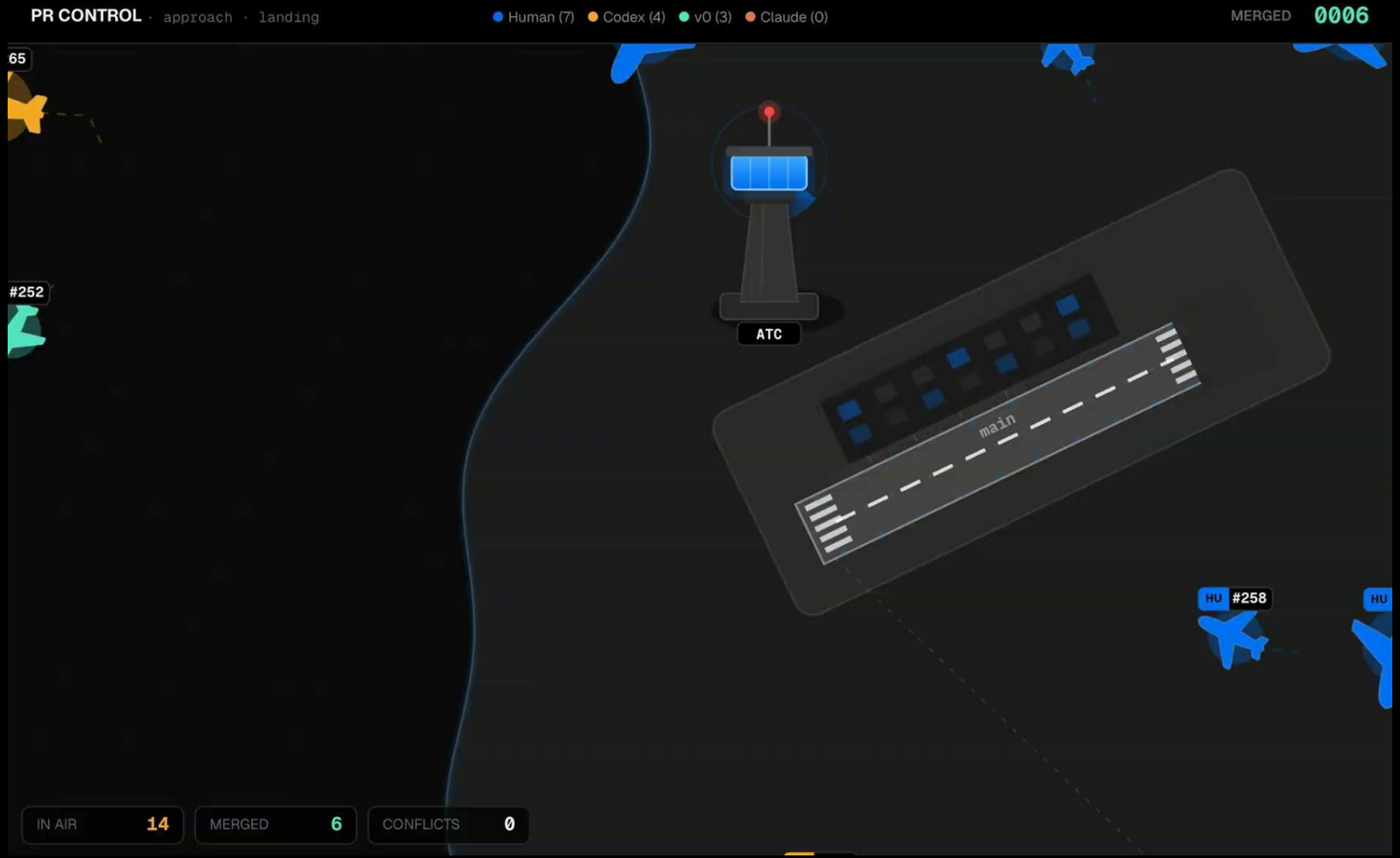
Use agents aggressively, but **do not outsource judgment**

Use agents aggressively, but **do not outsource judgment… yet**

Thank you

MATT COLLIER
STAFF SOLUTIONS ARCHITECT

@mattc0llier


PR CONTROL · approach · landing
Human (7)
Codex (4)
v0 (3)
Claude (0)
MERGED 0006
65
#252
ATC
main
HU #258
HU
IN AIR 14
MERGED 6
CONFLICTS 0