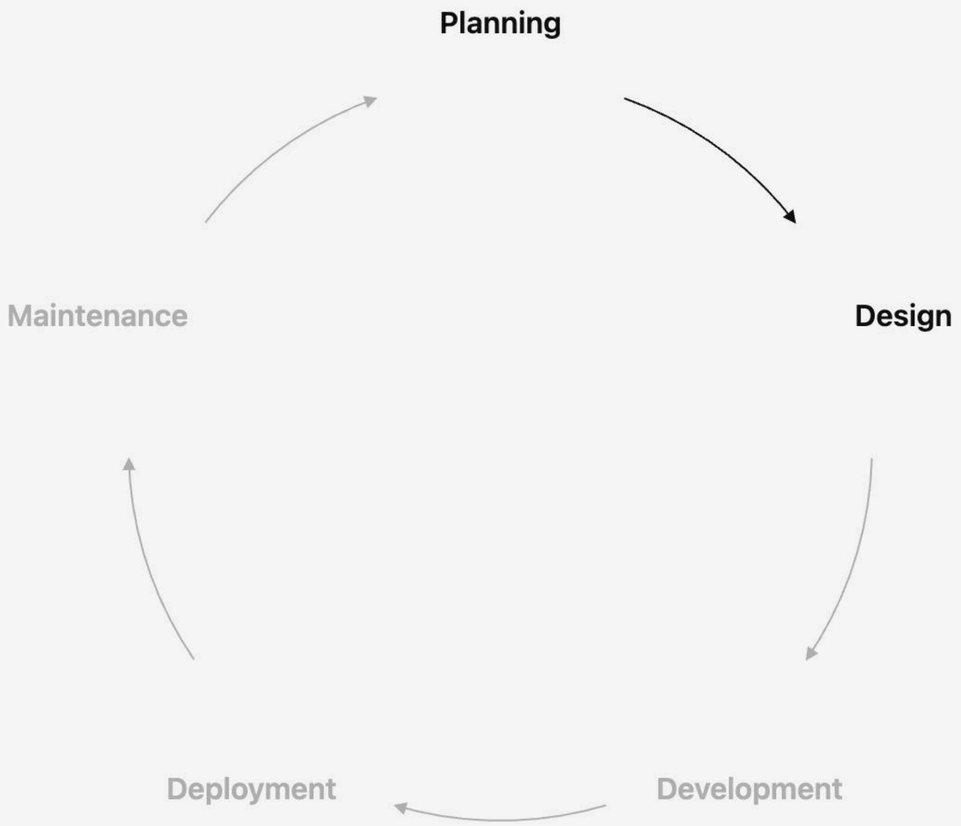
Enterprise grade, startup made

2,000+ enterprise customers
3 months
3 engineers



Always
Be
Carefully _____
Driving
Everything
Forward™

Always
Be
Carefully planning
Driving
Everything
Forward



What do we need to build?

Direct customer feedback

User research

Customer support feedback

Bug reports

Commits To main: 5 Most Impactful Ships Wednesday, April 22nd

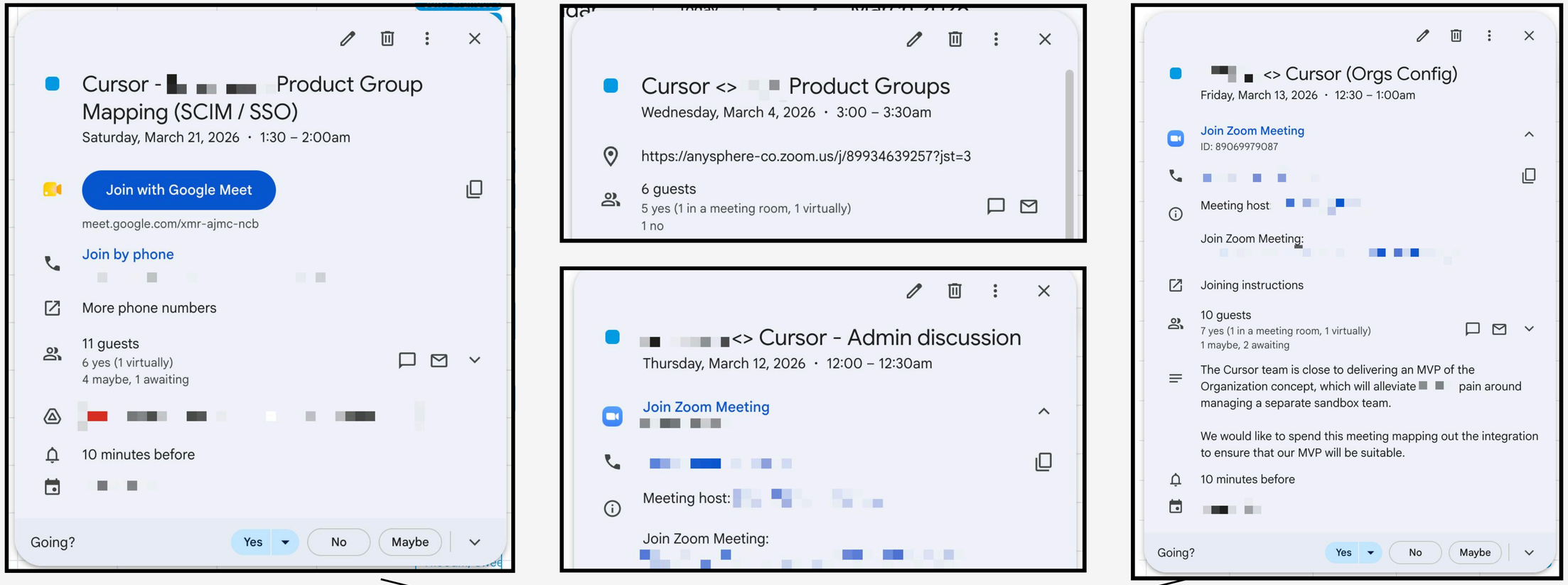
• [New] https://github.com/anysphere/everysphere/pull/103127 — *feat(anytool): organization create-team UX and WorkOS inheritance*
• Why impactful: changes core org/team provisioning semantics (org-level WorkOS inheritance, billing behavior simplification), reducing enterprise onboarding ambiguity and misconfiguration risk.
• [New] https://github.com/anysphere/everysphere/pull/98857 — *Org admin members: show only paid team memberships*
• Why impactful: fixes confusing duplicate membership representations for org admins, improving trust in org member data and admin workflows.
• [New] https://github.com/anysphere/everysphere/pull/103896 — *Aidan/paid wau growth segment tabs*
• Why impactful: expands paid WAU growth segmentation in analytics UX; high leverage for product/revenue decision-making.
• [New] https://github.com/anysphere/everysphere/pull/103891 — *remove v1 fireworks end state and swap to v2*
• Why impactful: infrastructure migration cleanup that reduces dual-path complexity and aligns provider handling on v2.
• [New] https://github.com/anysphere/everysphere/pull/103843 — *vscode: log slow DB close to sentry*
• Why impactful: direct reliability/observability improvement for diagnosing shutdown latency issues in client runtime.

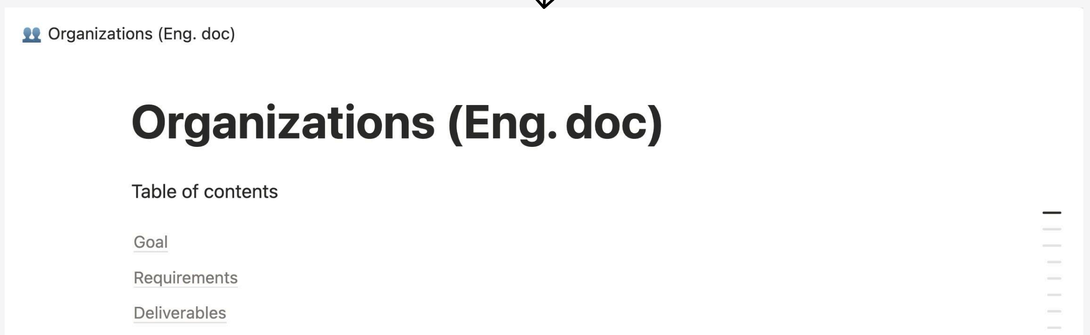
Top Customer Problems & Trends

• [New] Identity lifecycle + access integrity issues are still a top customer pain.
• Examples: SCIM duplicate-account behavior, contractor-to-employee identity transitions, enterprise users appearing as free/missing model picker, login redirect loops.
• [Ongoing] Billing/usage transparency is the most frequent strategic request cluster.
• Examples: Auto-mode model visibility requests, “unknown model” usage rows, dashboard mismatches, invoice line-item limit risk, BYOK cost-accounting confusion.
• [Ongoing] Admin governance controls are a major enterprise trend.
• Examples: per-tool MCP controls, repo-scoped GH app enforcement, CLI disablement, audit-log retention questions, model allow/block granularity.
• [Ongoing] Cloud Agent operability and controls remain high-priority.
• Examples: automations hanging, environment defaults at org scale, VM sizing requests, session/webhook behavior uncertainty.
• [New] From user feedback ingestion (#automatic-user-feedback): recurring reliability/UX regressions.
• Frequent themes: AGENTS/rules adherence after compaction, terminal hangs, MCP long-response freezes, git scanning stuck, chat history loss concerns, billing perception (“double charge”/quota burn).
• [Ongoing] Most important fixes to prioritize from customer signal:
• Reliability regressions in agent/runtime behavior.
• Usage/cost transparency and attribution correctness.
• Enterprise identity/governance controls with clearer adm

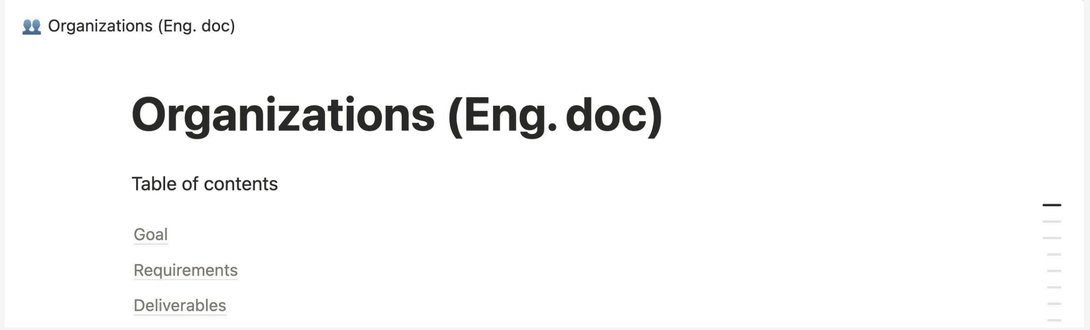
↓ Latest messages

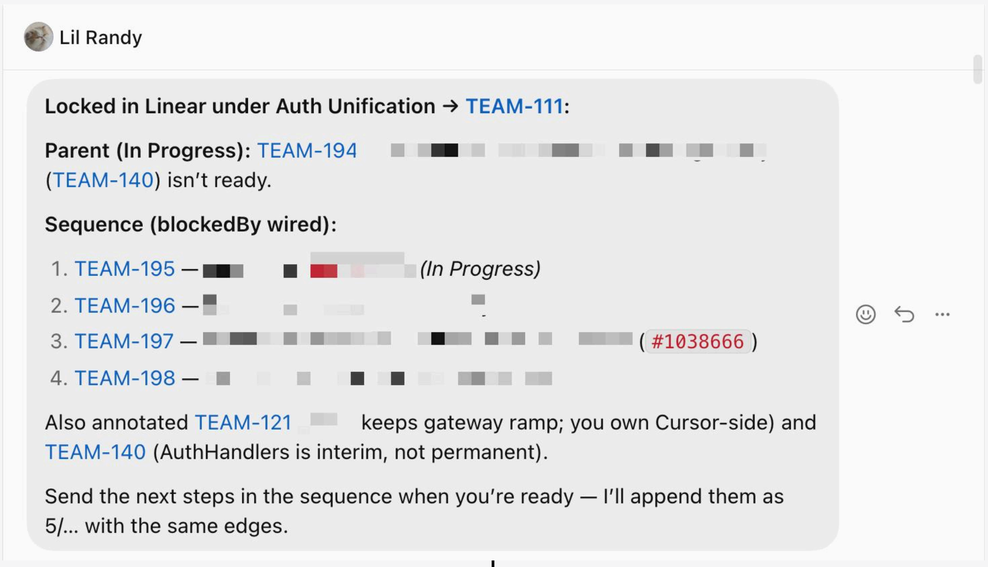
How to plan


Cursor - [redacted] Product Group Mapping (SCIM / SSO)
Saturday, March 21, 2026 · 1:30 – 2:00am
Join with Google Meet
meet.google.com/xmr-ajmc-ncb
Join by phone
More phone numbers
11 guests
6 yes (1 virtually)
4 maybe, 1 awaiting
10 minutes before
Going?
Yes
No
Maybe
Cursor <> [redacted] Product Groups
Wednesday, March 4, 2026 · 3:00 – 3:30am
https://anysphere-co.zoom.us/j/89934639257?jst=3
6 guests
5 yes (1 in a meeting room, 1 virtually)
1 no
[redacted] <> Cursor - Admin discussion
Thursday, March 12, 2026 · 12:00 – 12:30am
Join Zoom Meeting
Meeting host:
Join Zoom Meeting:
[redacted] <> Cursor (Orgs Config)
Friday, March 13, 2026 · 12:30 – 1:00am
Join Zoom Meeting
ID: 89069979087
Meeting host
Join Zoom Meeting:
Joining instructions
10 guests
7 yes (1 in a meeting room, 1 virtually)
1 maybe, 2 awaiting
The Cursor team is close to delivering an MVP of the Organization concept, which will alleviate [redacted] pain around managing a separate sandbox team.
We would like to spend this meeting mapping out the integration to ensure that our MVP will be suitable.
10 minutes before
Going?
Yes
No
Maybe


Organizations (Eng. doc)
Organizations (Eng. doc)
Table of contents
Goal
Requirements
Deliverables

How to plan





How to plan

TEAM-195

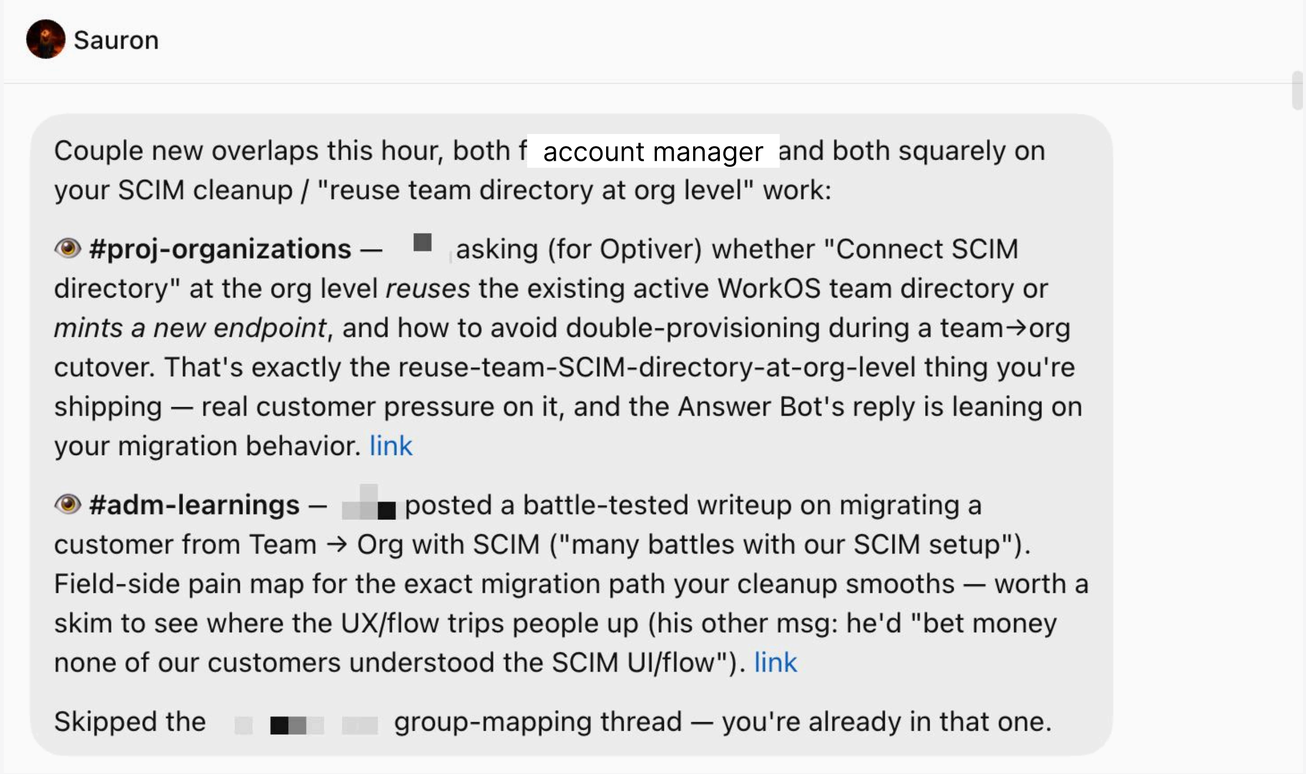
TEAM-196

TEAM-198

What don’t we need to build?

Sauron

keeps me aware of other teams who might be thinking about the same things, so no one has to re-invent the wheel.



Who plans

Ida & family

help bucket things into the same primitives, so we don’t need to constantly rebuild.

Ida

Ida, would it help if you made your own agents, one for researching through slack for adjacent ideas/customer requests, one for breaking down and researching the task, one for triaging? How many agents do you think you'd need?

Please make similar family profile photos for them as you did for yourself. You could make yourself a family or team!

The team's all set and nothing's pending on my end. Scout, Delver, and Filer are created with their sand-blob avatars, and I'm coordinating. Just send your next idea whenever you're ready and I'll put them to work.

Where the three landed:

Scout ✅ — strong Slack signal: customer + other h customers y (anchors), third customer (scoped keys), three a more ir customers (OIDC), two B more / customers (revocation/offboarding).

Fil ✅ — Linear tree created: parent ENT-3443 + 5 subtasks (ENT-3444 → ENT-3448), dependency-chained, mirroring your ENT-3401 cutover pattern.

Del ⏳ — cloud agent finished the research, but hasn't posted its file-cited findings into Linear yet. I nudged it; I'll ping you the moment they land.

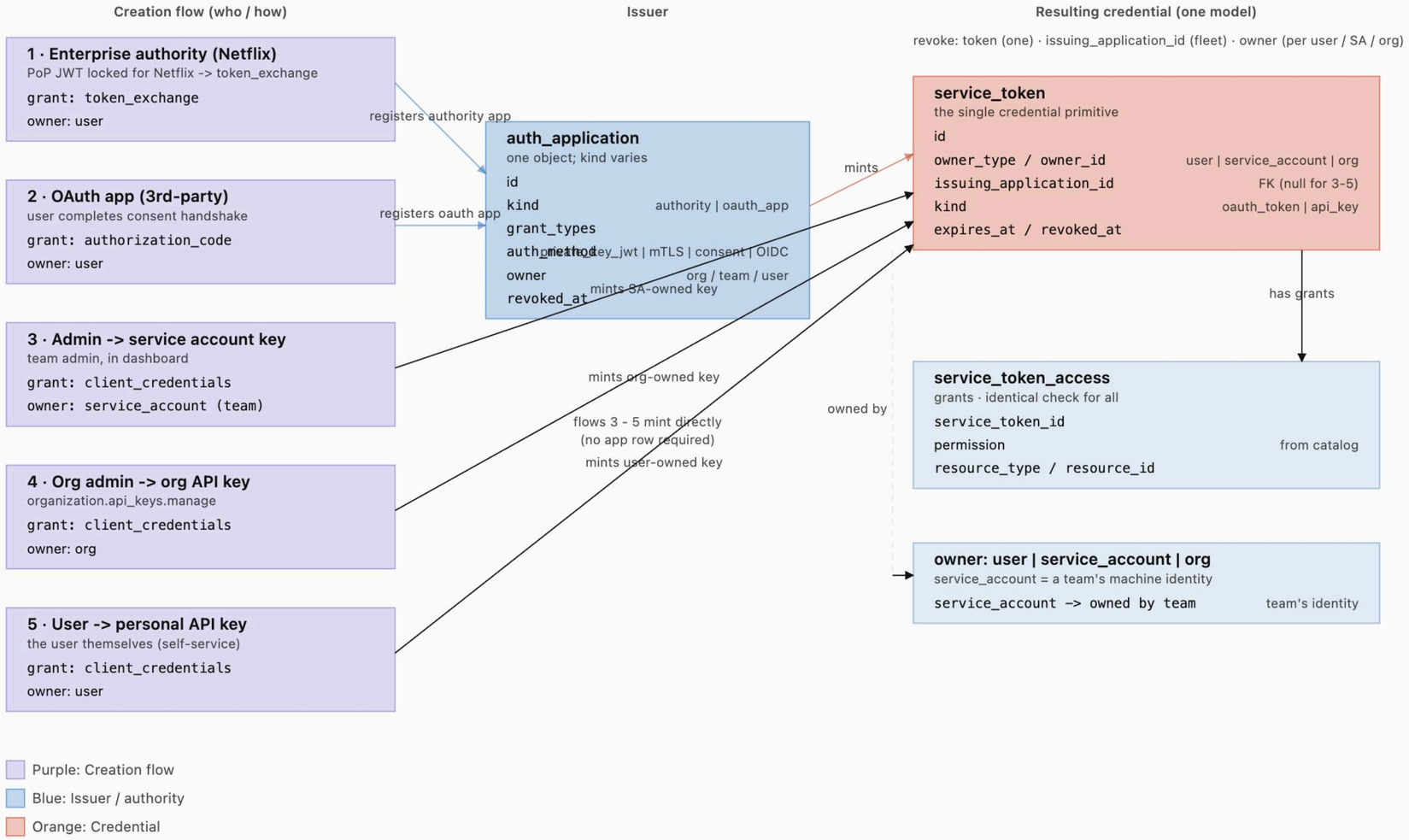
So the Linear issue is created and structured. The only thing outstanding is Del's code-level detail getting written in. Want me to have Del fold Scout's customer signal into ENT-3443 as the "why now" while it's at it?

Who plans

Ida & family  produced doc:

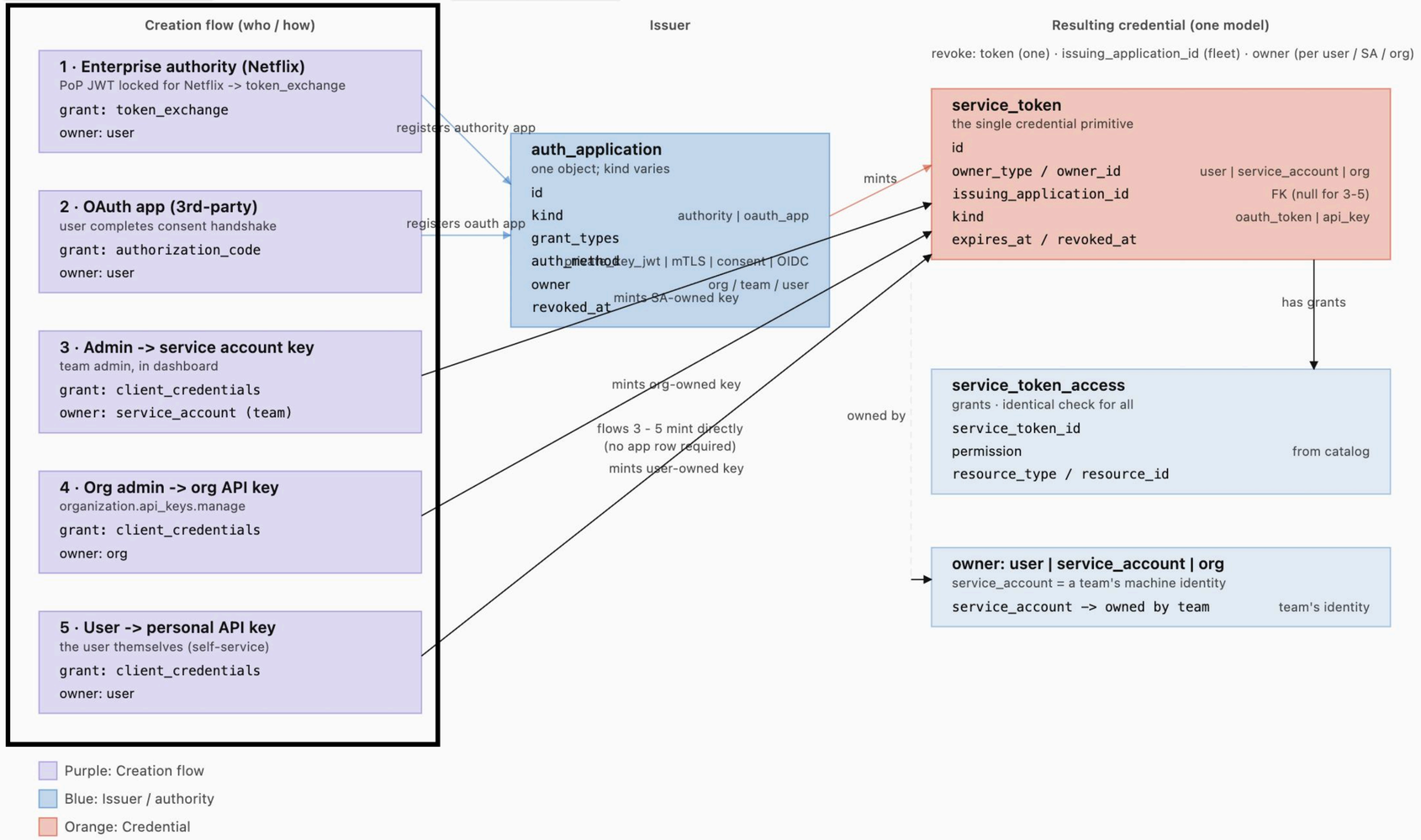
All credential-creation flows, one model

Every credential ([redacted]uthority, 3rd-party OAuth app, team service-account key, org API key, user personal key) lands on the same `service_token` + `service_token_access`. Flows 1 & 2 go through an `auth_application` row; flows 3–5 mint directly. The authZ check is identical for all.



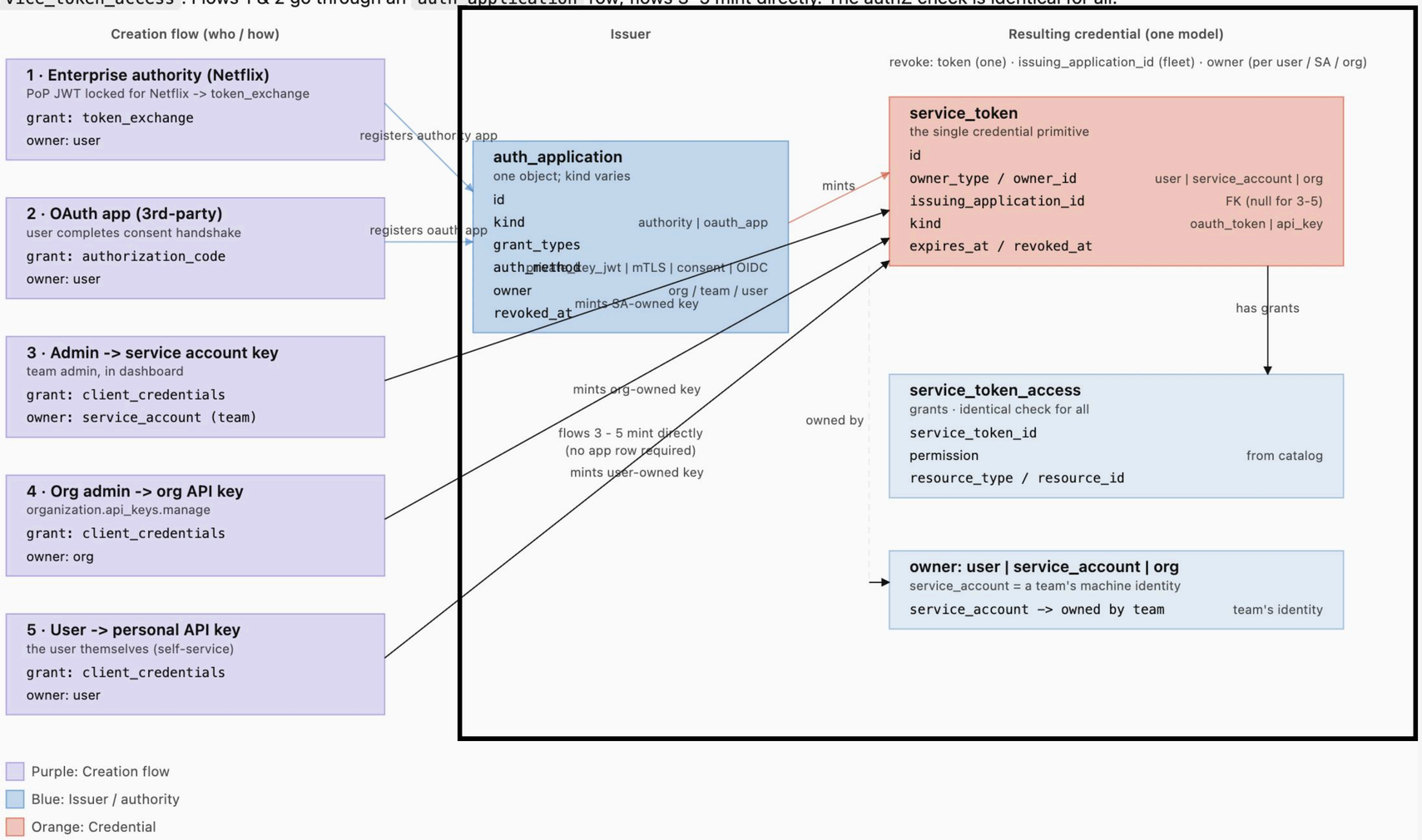
All credential-creation flows, one model

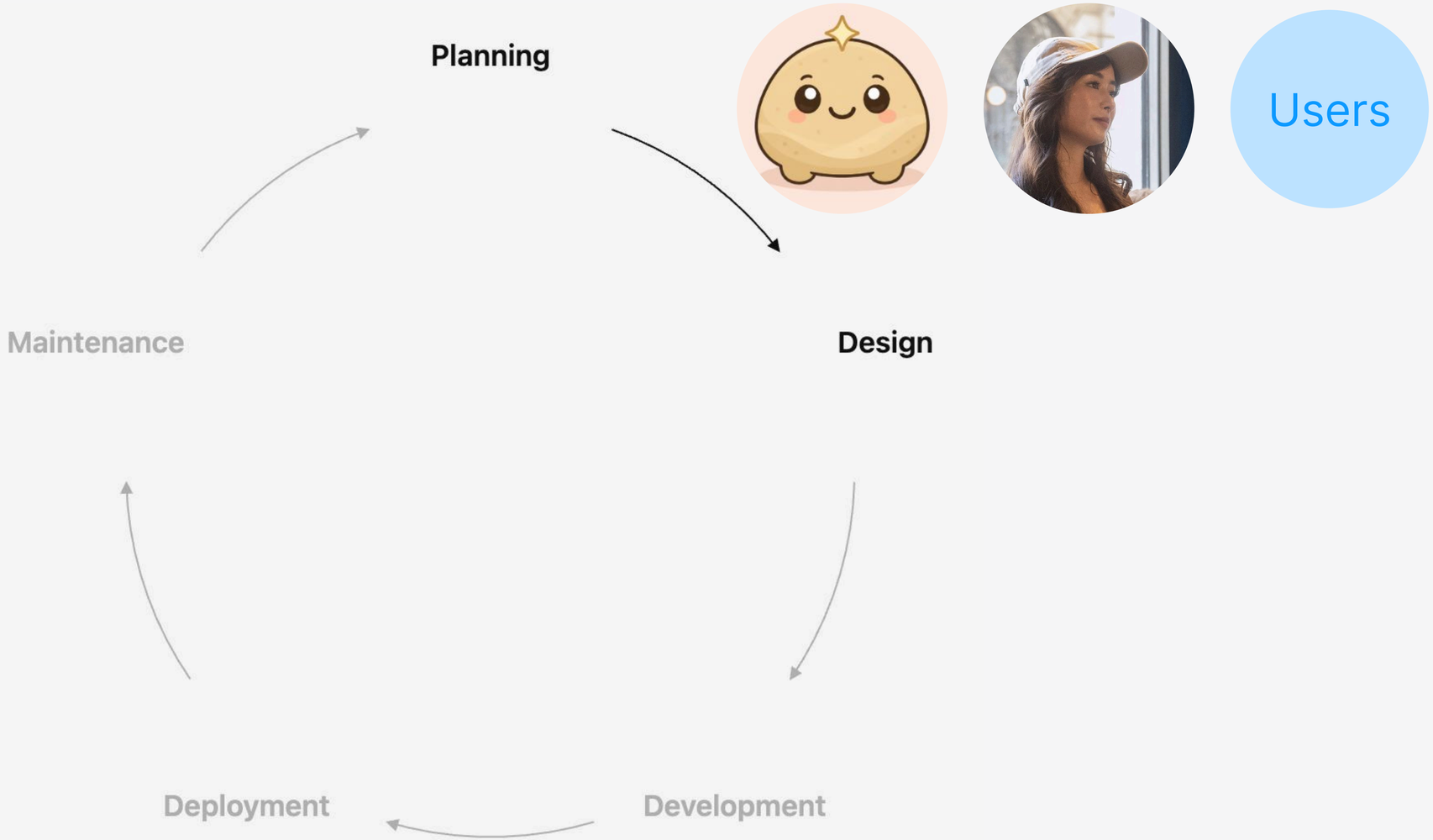
Every credential [redacted]uthority, 3rd-party OAuth app, team service-account key, org API key, user personal key) lands on the same `service_token` + `service_token_access`. Flows 1 & 2 go through an `auth_application` row; flows 3–5 mint directly. The authZ check is identical for all.



All credential-creation flows, one model

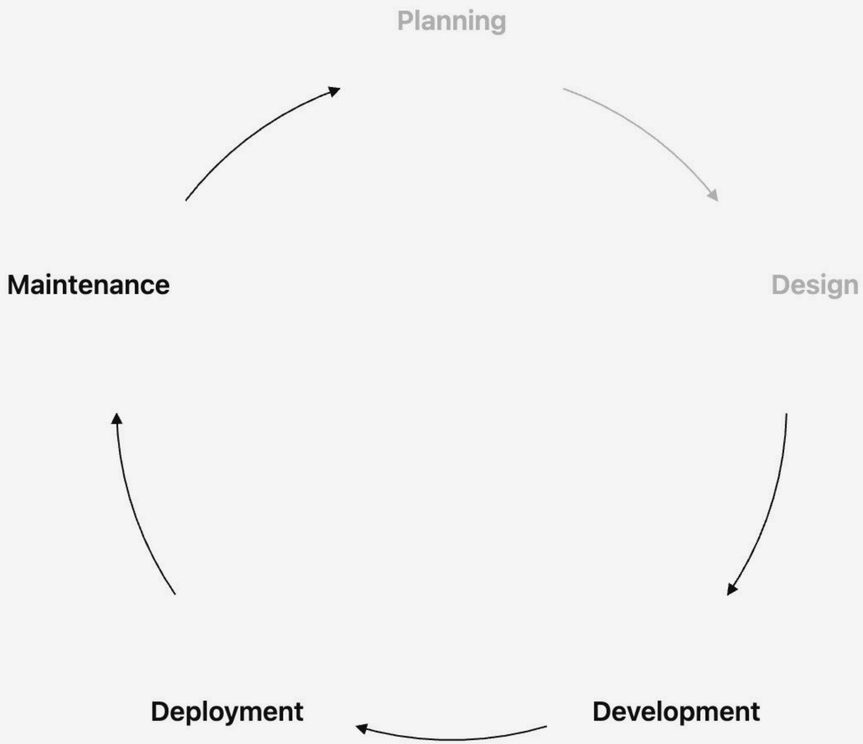
Every credential ([redacted]uthority, 3rd-party OAuth app, team service-account key, org API key, user personal key) lands on the same `service_token` + `service_token_access`. Flows 1 & 2 go through an `auth_application` row; flows 3–5 mint directly. The authZ check is identical for all.




Planning
Design
Development
Deployment
Maintenance
Users

Planning

Always
Be
Carefully _____
Driving
Everything
Forward



Lil Randy

is my project manager, and he keeps me driving everything forward, all of the time.

Who builds

Lil Randy

Morning notes. Auth tiger standup is at 10:00, and a couple of Slack threads are actually waiting on you.

Top of mind

- Auth tiger standup in a few minutes (10:00–10:30).
- [redacted] needs a yes/no in #proj-auth-unification: does unification always require both a Cursor *and* a Grok auth to hit either backend? He said technical decisions hinge on it. thread
- [redacted] systems-design interview at 12:15 (Ashby briefing in the invite).

Today

- 10:00 Auth Tiger Team Standup
- 11:00 Airbnb checkout (Menlo Park)
- 11:15 Core Services Weekly Sync
- 12:15 Systems Design, [redacted] (Settings Sync)

Hanging Slack

- [redacted], #proj-auth-unification (8:04 PT): dual Cursor+Grok auth confirmation, still unanswered. thread
- [redacted] group DM (7:52 PT): still down for the Amsterdam Compile chalk talk Sept 22? dm
- [redacted], #issues-grok-bot (3:13 PT): another customer blocked on a stale `identity_joins` row, asking for a one-shot sweep of leftover rows. thread
- [redacted] asked for a review on Origin #1023707 (CR Gate pointed at you or [redacted]).
- [redacted] (Anysphere DM, late Tue): interested, wants a call if you have people on this side.

Already handled this morning: [redacted]'s "is identitydb public on purpose?" (you: not deliberate)[redacted] i's Aug 21 customer check-in.

PRs

Review that is actually yours: GH #223418 Route WorkOS sign-ins through xAI login (gated), Jorge, updated this morning.

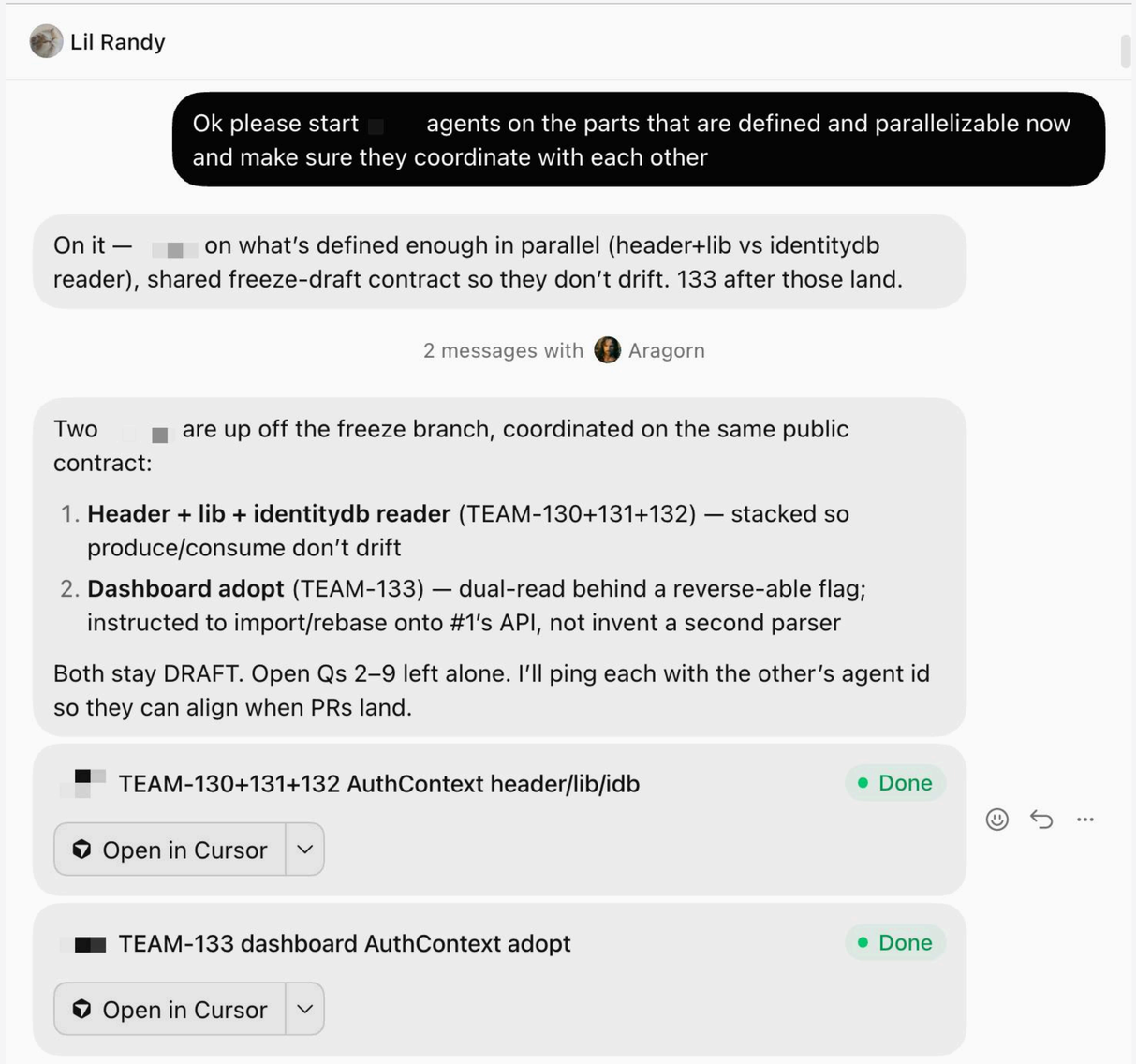
Yours, open, and stale (most idle weeks, PR Freshness red):

- Approved, sitting: GH #167246 org group CRUD, GH #162872 offboarding banners, GH #156893 group type toggle, GH #181689 FSD MCP kill switch
- CI actually failing: GH #153968 in-IDE team switcher, GH #164758 IdP merge cron, GH #160298 SCIM group↔IdP guard, GH #151021 Hoyoverse WorkOS (Security Findings Gate)
- Long-lived drafts: GH #156635 Slack team picker, GH #165337 PlanetScale DR break-glass, plus a few July FSD/zookeeper drafts

Linear (in progress, recently touched or urgent)

Lil Randy

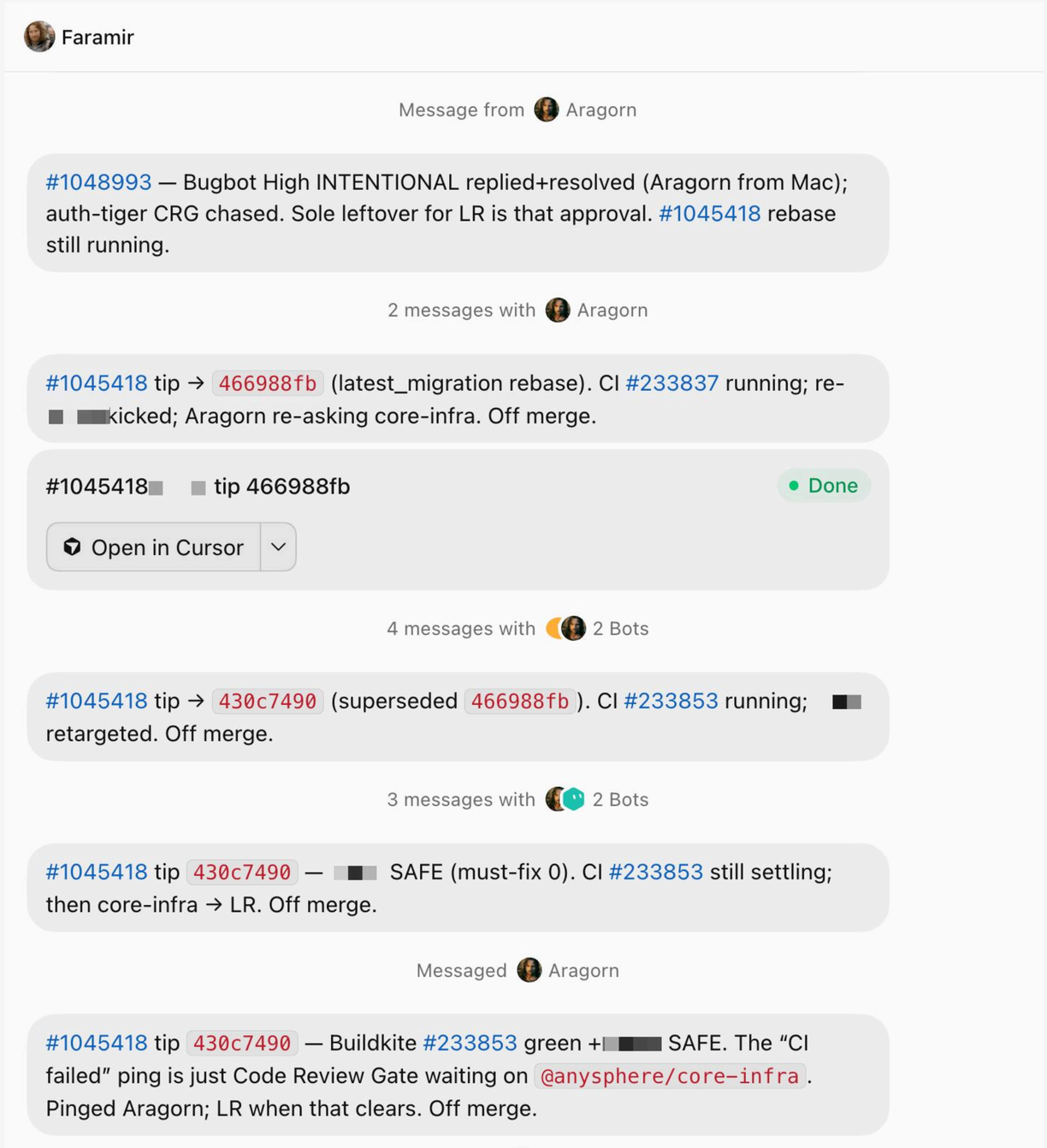
orchestrates greenfield work, and parallelizes work to get things done.



Who builds

Faramir

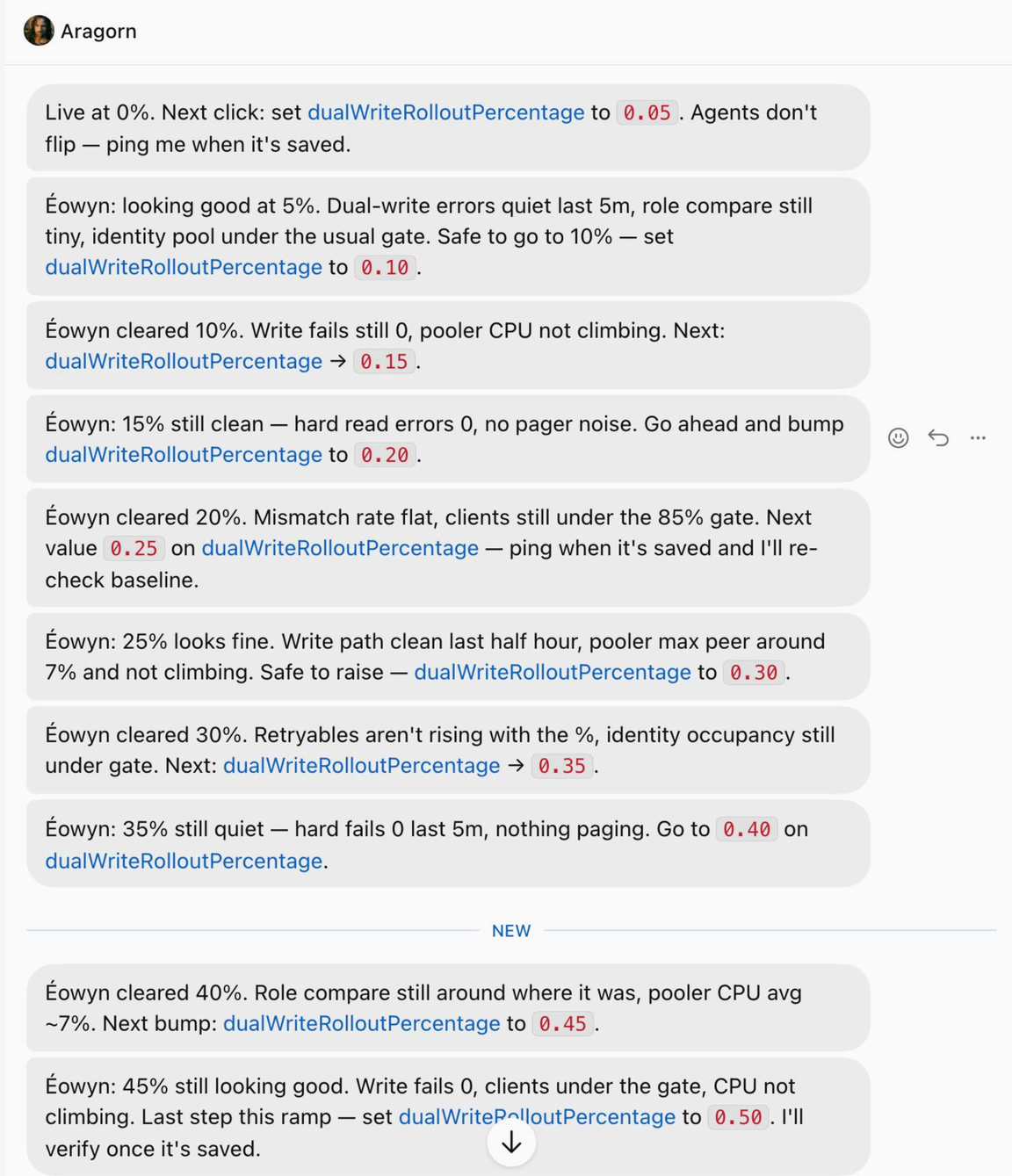
drives PRs to review and merge-ready.



Who builds

Aragorn

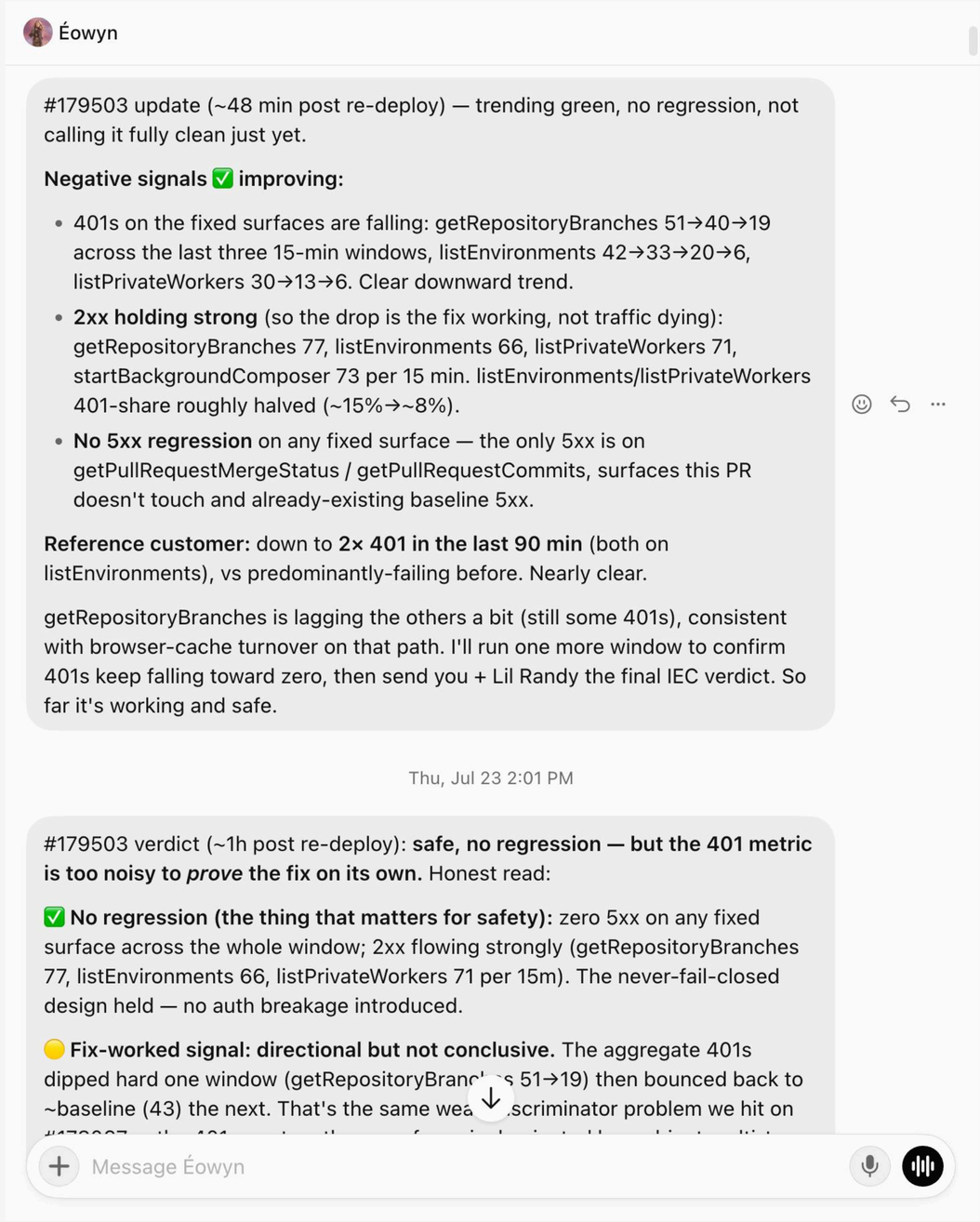
walks me through complex migrations and delicate brownfield work.



Who builds

Éowyn

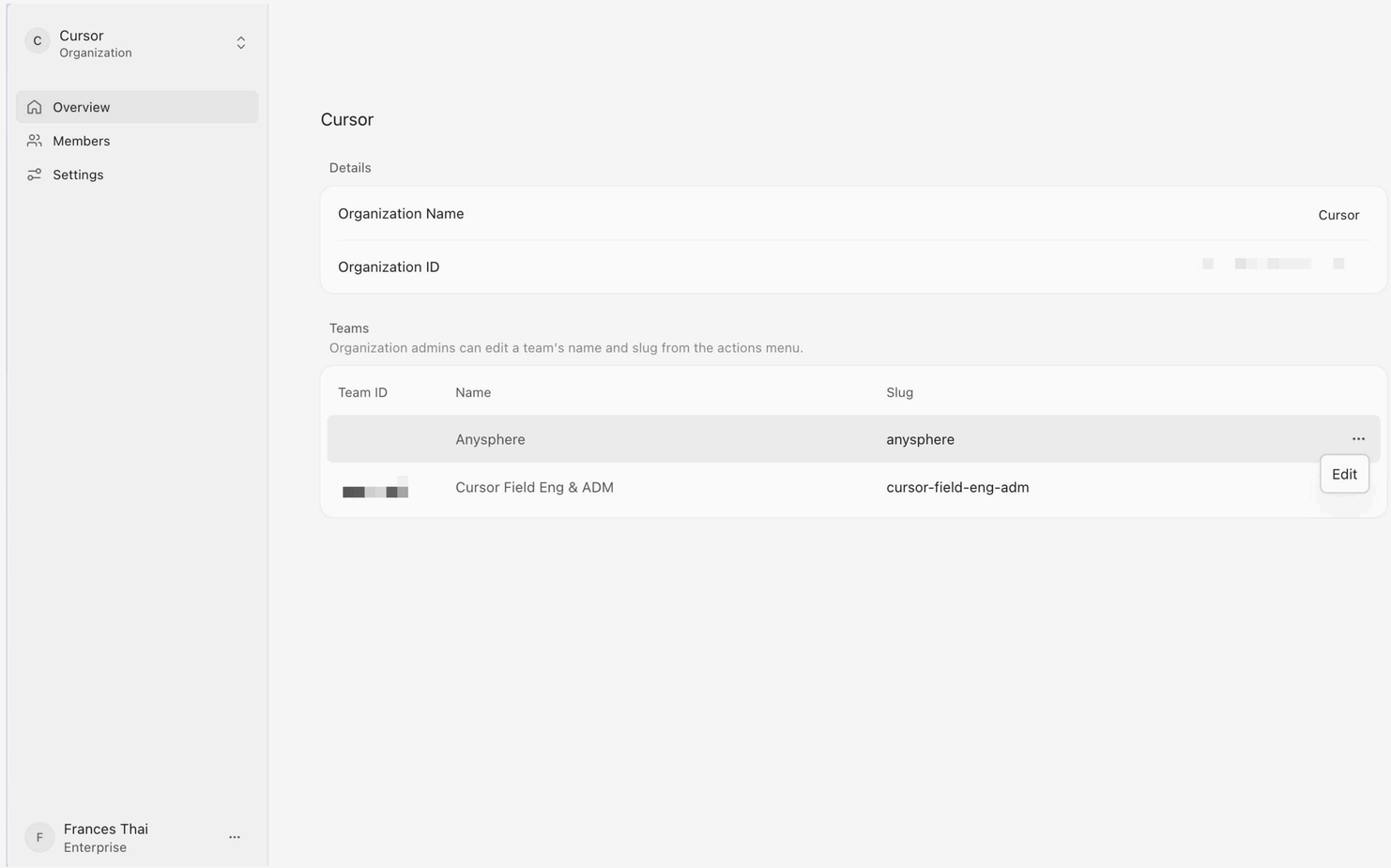
keeps watch over changes: when they deploy, if we should rollback, if the changes look good.



Who builds

Dogfood early

The fastest iteration loop is the one you test on yourself.



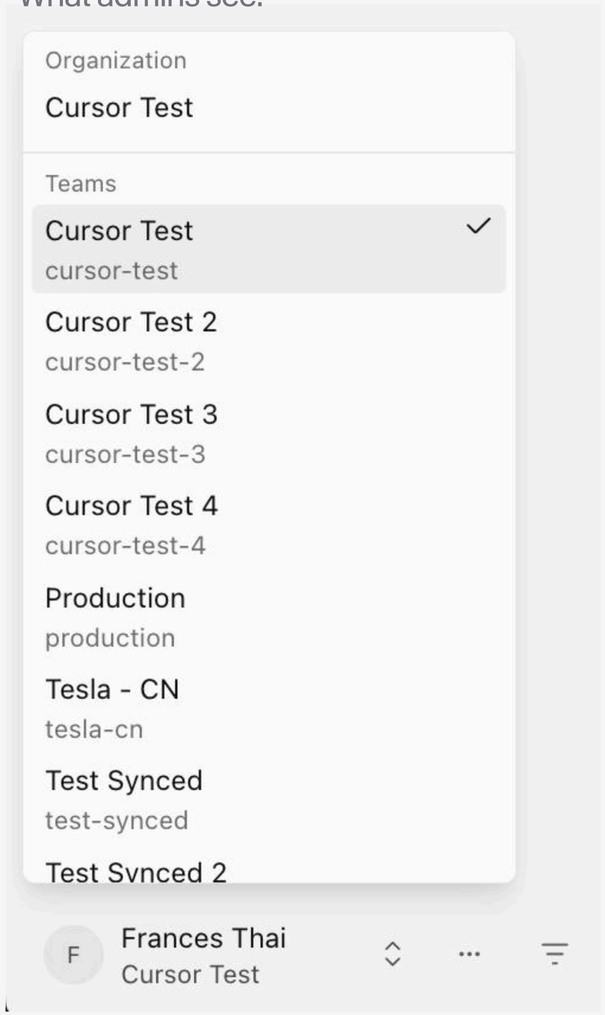
How to build

Don’t stress users

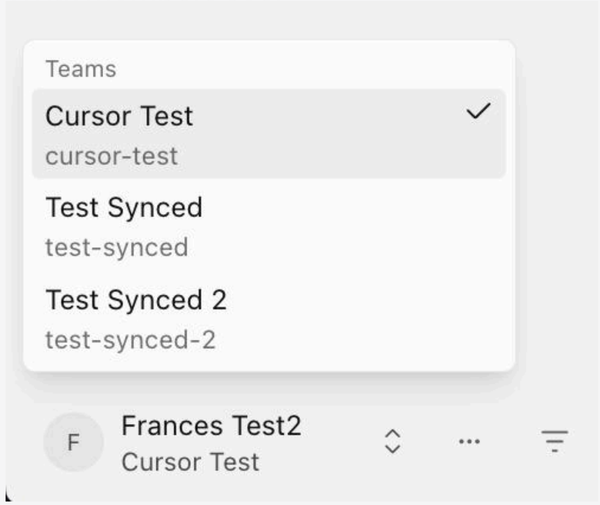
Incrementally ship to different user segments.
Nobody sees what they don’t need to see.
Only admins see admin features.

...and we don’t use the word “migration”.

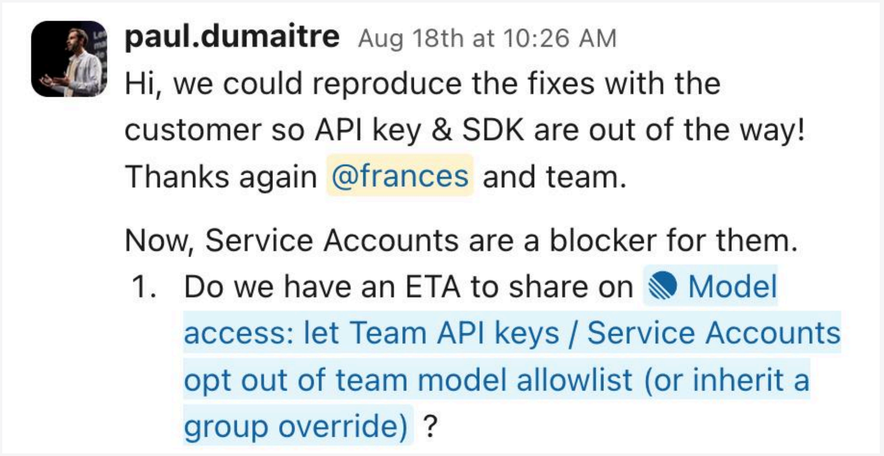
What admins see:



What members see:



How to build


paul.dumaitre Aug 18th at 10:26 AM
Hi, we could reproduce the fixes with the customer so API key & SDK are out of the way! Thanks again @frances and team.
Now, Service Accounts are a blocker for them.
1. Do we have an ETA to share on Model access: let Team API keys / Service Accounts opt out of team model allowlist (or inherit a group override) ?
Frances Thai Aug 21st at 4:51 PM
@Cursor Use [redacted] model to plan how service accounts could neatly be added as a principal into organization groups. Make sure to not invent any new settings or primitives; we merely want to fit service accounts into the shape that users take on in Organization-level Cursor groups. Build the UX to add/remove service accounts from Cursor groups and also neatly plug into the existing group setting reconciliation path that happens for model allowlist enforcement. Take videos and screenshots as evidence of the different UX flows. (edited)

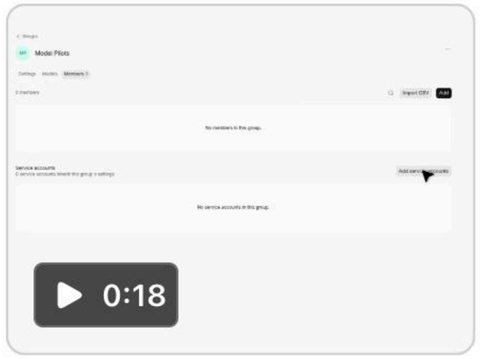
Frances Thai Aug 21st at 4:51 PM

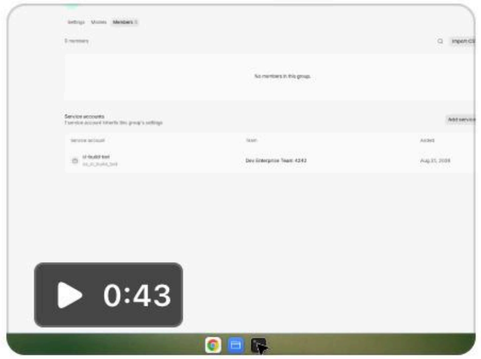
@Cursor Use [redacted] model to plan how service accounts could neatly be added as a principal into organization groups. Make sure to not invent any new settings or primitives; we merely want to fit service accounts into the shape that users take on in Organization-level Cursor groups. Build the UX to add/remove service accounts from Cursor groups and also neatly plug into the existing group setting reconciliation path that happens for model allowlist enforcement. Take videos and screenshots as evidence of the different UX flows. (edited)

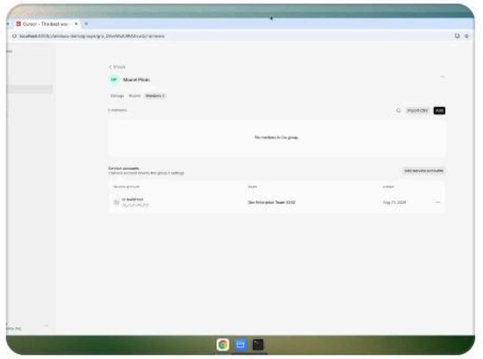
Cursor AGENT Aug 21st at 6:50 PM

Done and working end to end locally — PR #1017918 (draft, no timeline promises, this is the starting point).

3 files | Download all

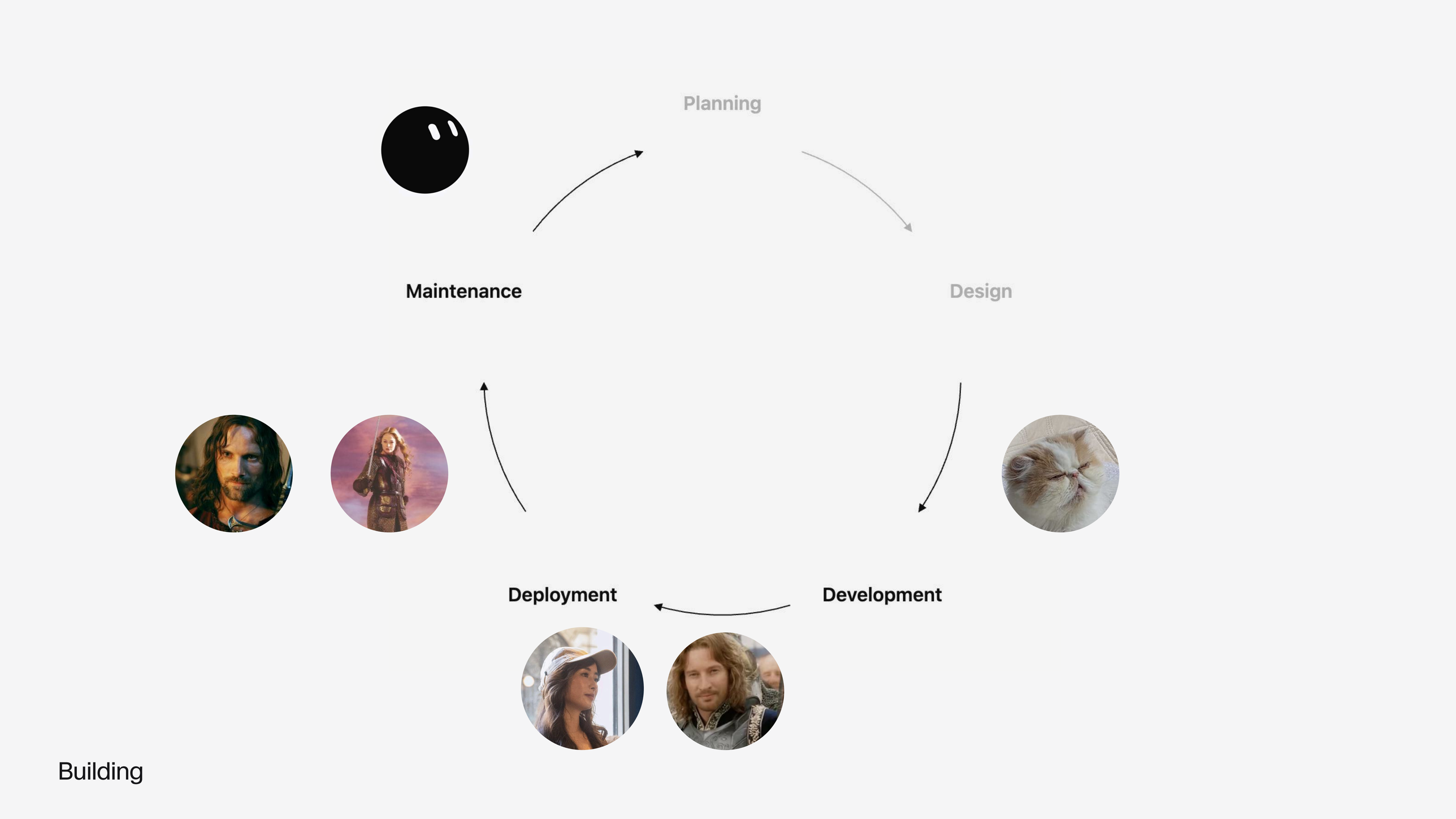




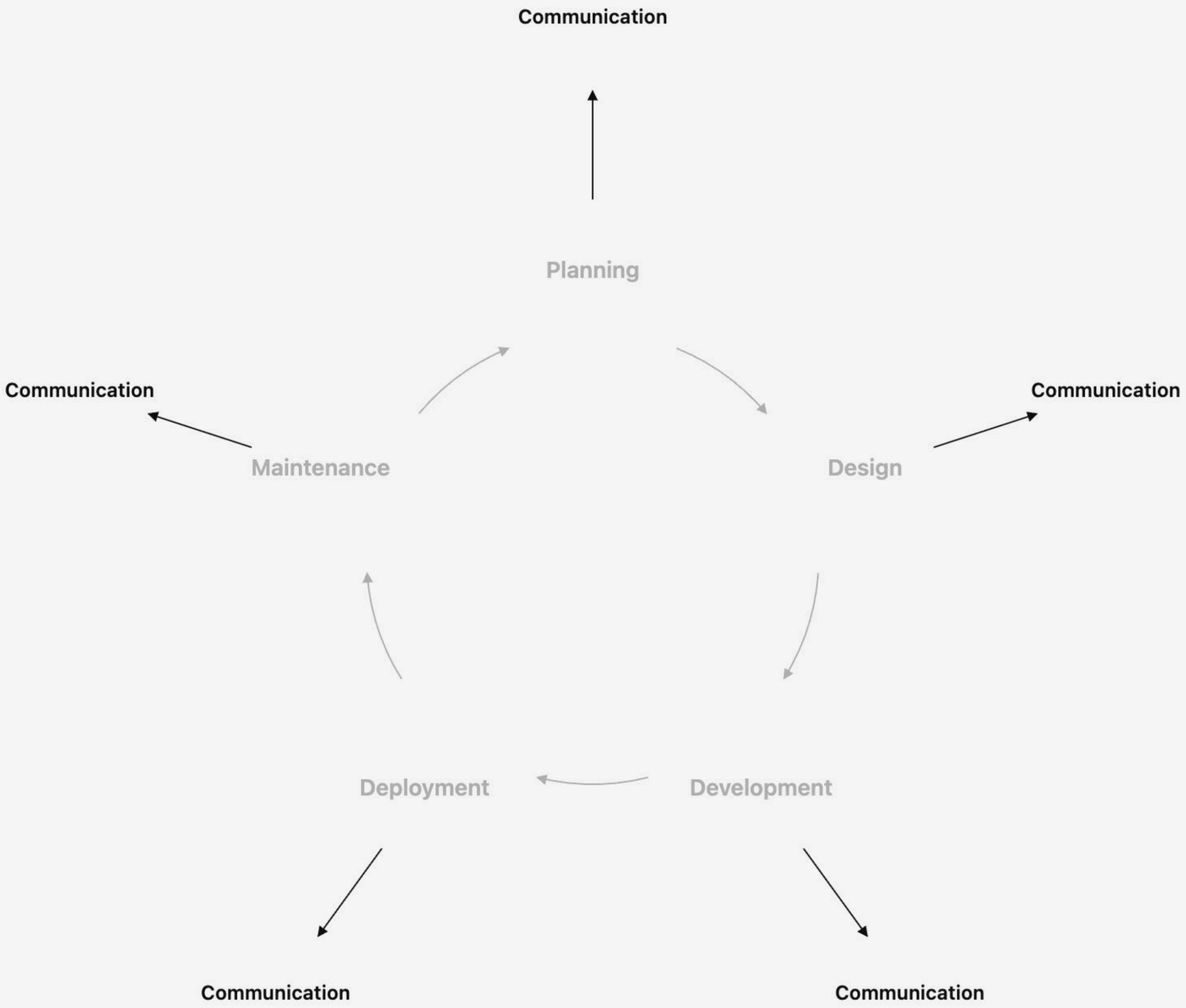


That's everything for this pass: PR #1017918 is up as a draft with the full plan in the description, and the demo videos/screenshot above show the roster UX and the allowlist reconciliation working against a live local stack. I'm watching CI and review comments on the branch and will pick up anything that comes back.

Building



Building


Communication
Planning
Communication
Design
Communication
Development
Communication
Deployment
Communication
Maintenance

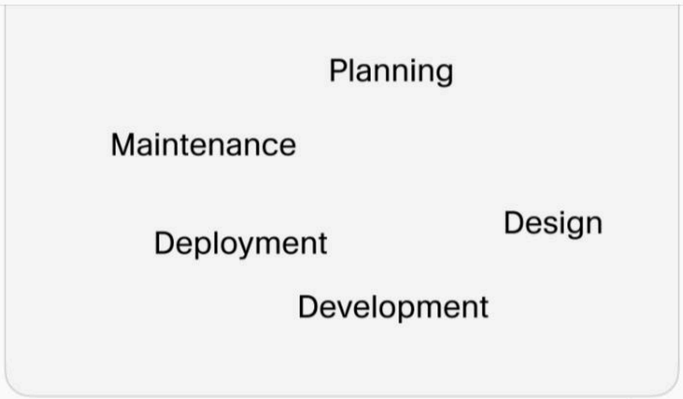
Always
Be
Carefully communicating
Driving
Everything
Forward

Arwen

creates graphics and diagrams for better communication.

Who communicates

Arwen



Hi arwen. could you make me a nice, evenly spaced SDLC circle with these labels, and arrows from one to the other in a clockwise manner? minimalistic is gerat with the same background color please

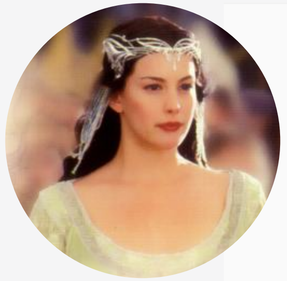
NEW

Making a clean SDLC circle canvas from those five labels, evenly spaced with clockwise arrows on the same light gray.

Done. Evenly spaced clockwise loop on `#F5F5F5`: Planning → Design → Development → Deployment → Maintenance.

Path: `/Users/frances/.cursor/projects/Users-frances-everysphere/canvases/sdlc-circle.canvas.tsx`

Open it in Cursor to see the circle. Want any tweaks (label size, ring radius, title wording)?



Proposal · Access + refresh

Proposal (not locked). After `token_exchange`, Cursor returns a short-lived user-owned access token plus a Cursor-issued refresh token. This is the user-owned credential Cursor CLI holds. Separate from the locked Option A authority app auth and from enterprise key or cert rotation. TTL numbers are TBD. Cursor owns access default and max; [redacted] may request a shorter access lifetime.

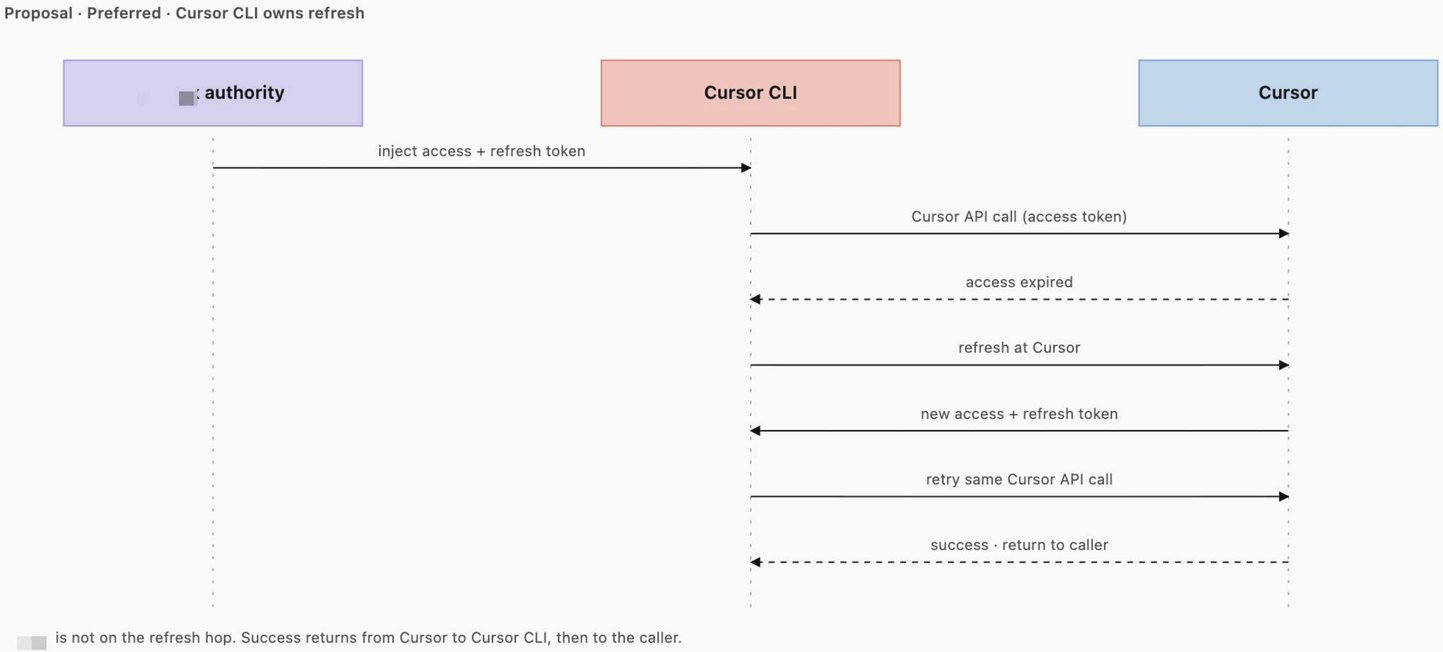
Proposal · Who builds what

Cursor owns Cursor CLI refresh-then-retry and the Cursor refresh endpoint (and any other Cursor-owned client that holds this mint).

[redacted] owns bootstrap and remint (`token_exchange`) and how access + refresh get into the process (token file, env and restart, or sidecar). [redacted] does not build a refresh client on top of Cursor CLI.

Proposal · Preferred sequence · Cursor CLI owns refresh

[redacted] injects the access token and refresh token into Cursor CLI (file or env per the Cursor contract). Cursor CLI calls the Cursor API. If the access token is expired, Cursor CLI refreshes at Cursor, stores the new access token and refresh token, and retries the same call. [redacted]'s not on the refresh hop.

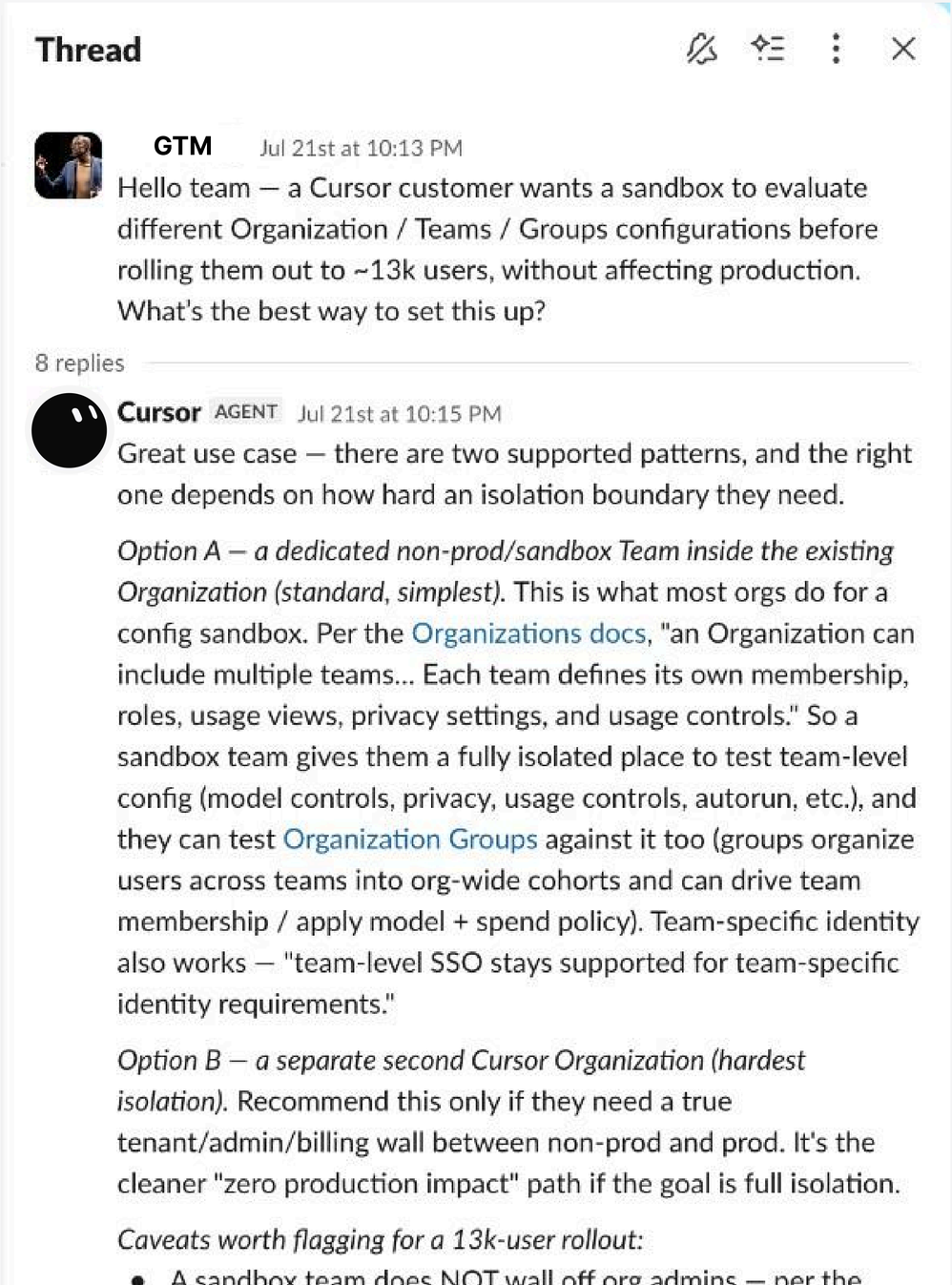


Who communicates

Intra-team communication

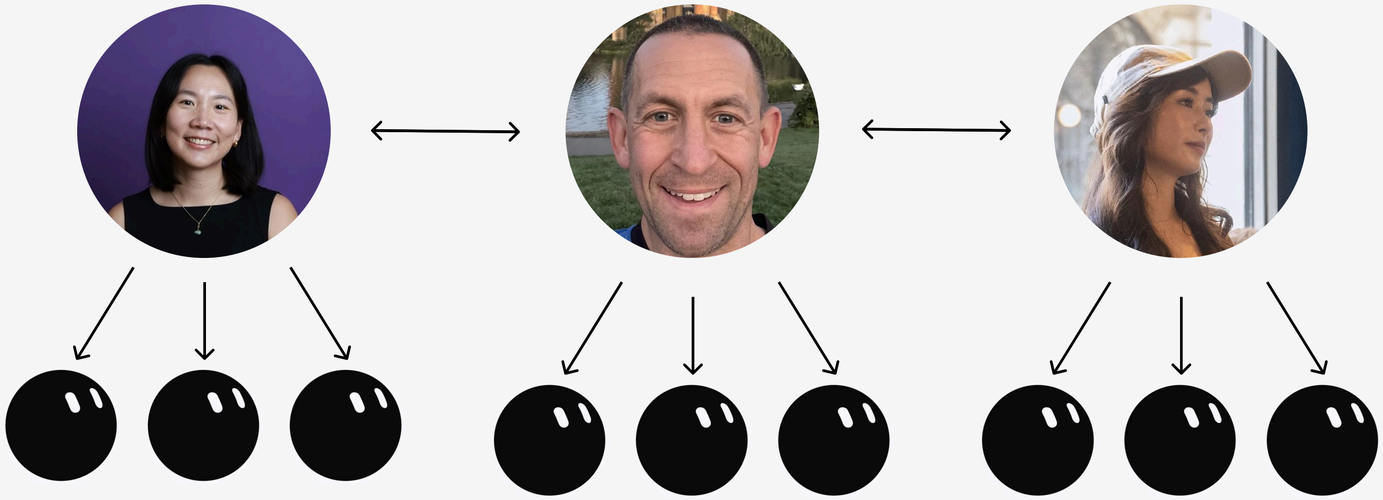
Agent answers questions, updates its own knowledge stores, and repeats. Keeps internal stakeholders up-to-date.

How to communicate



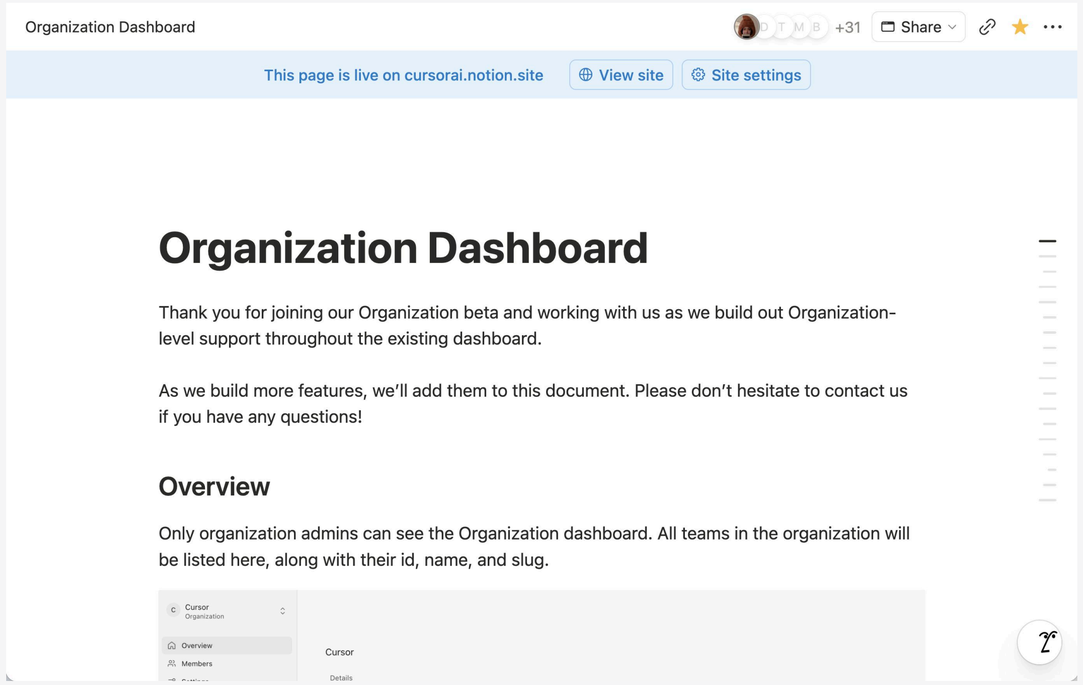
Inter-team communication

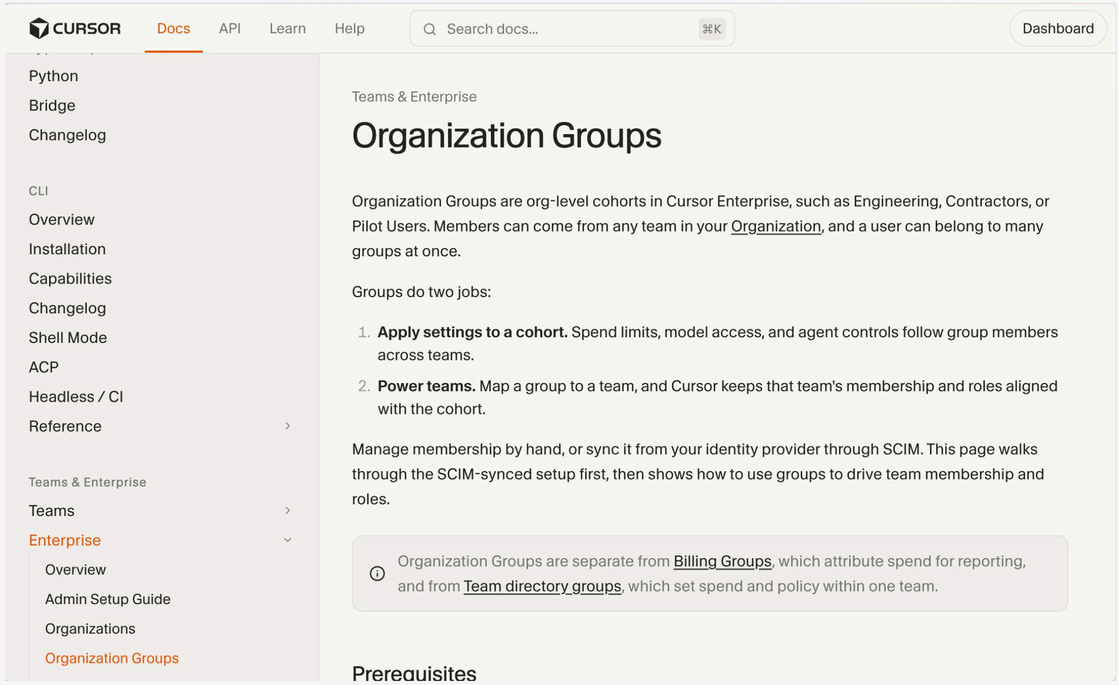
Regular updates and syncs keep teammates up-to-date with each other.



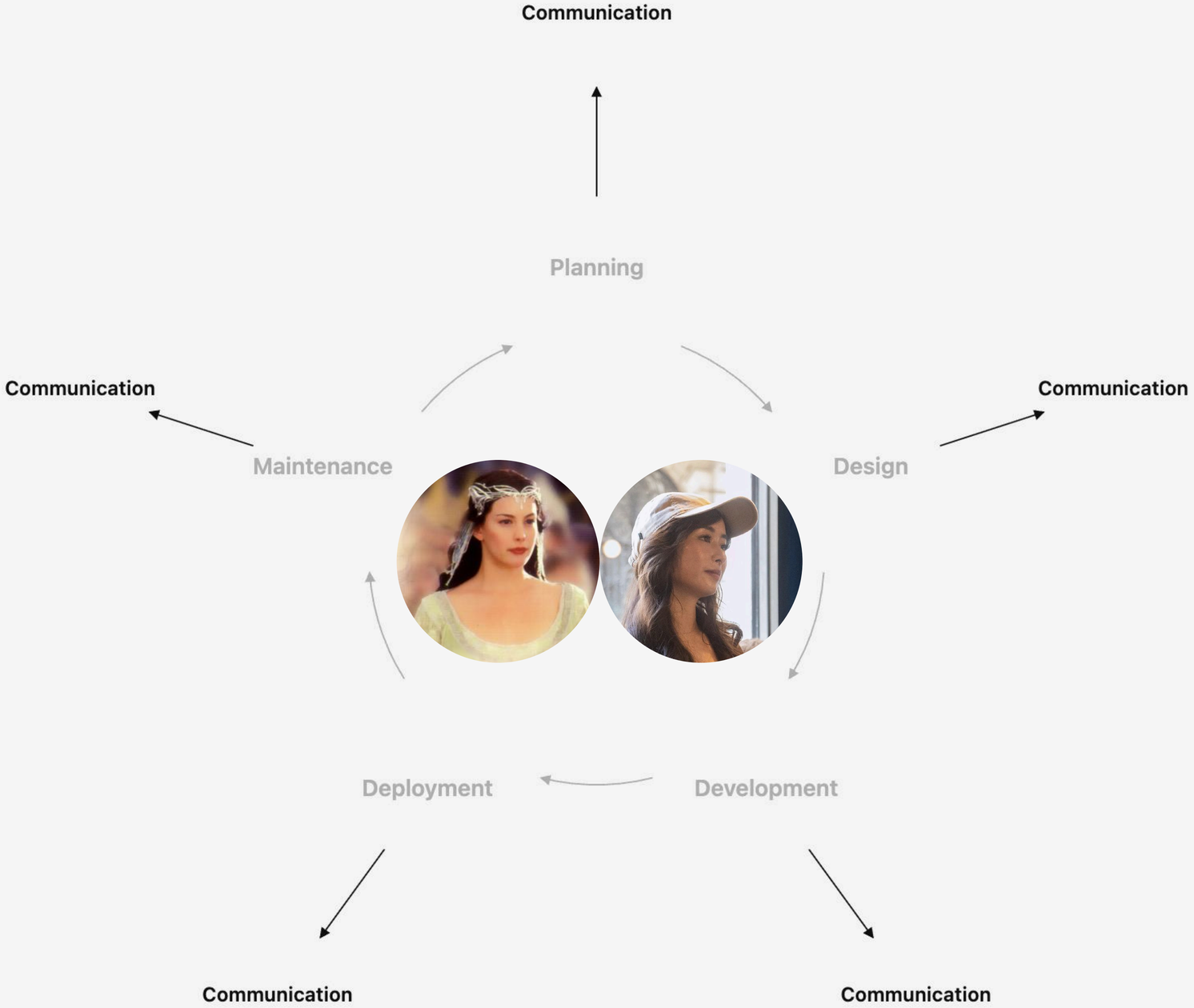
External communication

Agents help us keep beta and public docs up-to-date.





How to communicate



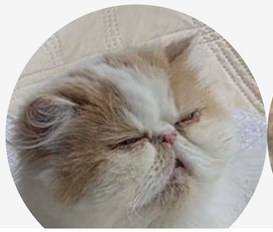
Communicating

What does it take to do a lot with very little?

It takes a factory

Ideator

Migrator



Orchestrator

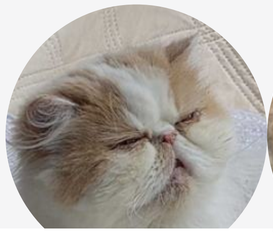
Driver

Sentry

It takes a ~~factory~~ village

Ideator

Migrator



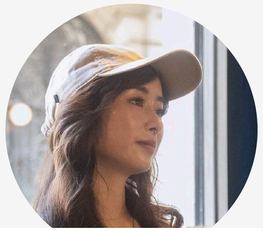
Orchestrator

Driver

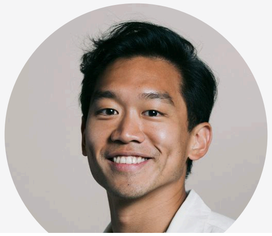
Sentry

Shipper

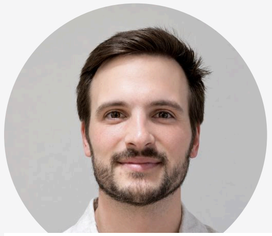
Sculptor



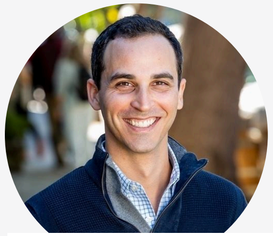
Architect



Artist



Tactician



Visionary

Thank you!