THE AI TOOLCHAIN

NO. 026

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED SEPTEMBER 14, 2026 · EVERY WEEKDAY

The daily firehose — everything the toolchain shipped today, already filtered.

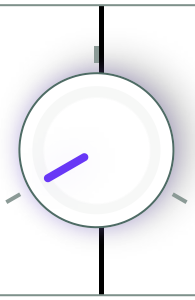
// HOW THIS ISSUE IS MADE

We read every release from the 357 tools on our watchlist at the source — **GitHub and GitLab release notes**, vendor **release pages** and **changelogs**, project **blogs and feeds**, vendor **press releases**, and the **source code** behind the tag. Bug-fix-only releases and non-product newsroom noise are dropped; what's left is summarized down to the new capability, how to try it, and any **screenshots or videos** the release itself published. Every entry links to the sources it was built from.

VIEW

ISSUE VIEW

full issue



FULL ISSUE | EXPAND IN PLACE | PREVIEW PANE

Do you prefer **full issue**? 👍 👎

▸ // CONTENTS SHOW 34 tools

▾ // THE BRIEF HIDE

What stands out across today's releases, grouped by what it lets you do. Every tool named links to its entry below.

Three unrelated things stand out. Braintrust now mines your traces for recurring failure modes instead of making you eyeball runs one by one. Anthropic's `ant apply` syncs agents, skills and memory stores from repo files, so agent config lives in Git. And HyperResearch put an SSRF gate with per-type response size caps in front of its fetchers.

EVALUATE

Find out how your agent fails without reading traces one at a time

Braintrust's Patterns agent clusters failure modes across a trace corpus and Debugger explains a single bad run, which is the step teams currently do by hand after every eval regression. FailproofAI adds transcript ingestion and harness discovery for OpenClaw 2026.9, so past agent runs become a policy-testing corpus rather than dead logs.

Braintrust · FailproofAI

BUILD

Keep agent definitions in the repo instead of clicking them into a console

`ant apply` declaratively syncs agents, environments, skills, memory stores and deployments from files, so an agent's configuration is reviewable and revertible like code, and drift between staging and prod stops being invisible. Cursor's Cloud Agent can now start a project with no pre-existing repo, forward ports for live in-editor previews, and publish to Vercel.

Anthropic · Cursor

GOVERN

Stop a web-fetching agent from being pointed at your own internal network

Any tool that fetches a model-supplied URL is an SSRF hole into cloud metadata and internal services; HyperResearch adds a gate with per-type response size caps and stricter TLS certificate handling, which also bounds the blast radius of a hostile multi-gigabyte response. Docker Sandboxes narrows secret scoping and adds SSH agent forwarding , so an agent building code can install private npm packages without holding a long-lived token.

HyperResearch · Docker Sandboxes

all 171 changes

BUILD

◆ AI Coding Agents

Warp ⓘ 1 RELEASE · 2026-09-14 CHANGELOG ↗ RANK

Warp is an AI terminal that runs commands and helps developers build software.

> Warp's latest changelog centers on team-scoped access across the Oz CLI and environment pickers, first-class support for Grok Build sessions, and extensions to shell widget handoffs and customizable keybindings.

└──▷ WHAT SHIPPED · 5 FEATURES most completely described first ? what's the number?

01 **Team scoping across Oz CLI and environments** NEW 85

Warp extended team-based access control throughout the Oz CLI and environment surfaces: the new `oz model list` command lists a team-specific model catalog, runner listing and name-based updates now support team selection, API key list, create, and expire commands are team-scoped, Oz agent model catalogs now follow the active team selection, and environment lists and interactive pickers follow the selected team, showing both personal environments and environments for the selected window team.

List the model catalog scoped to your team so you only see models your team is entitled to use.

```
$ oz model list
```

— Names multiple scoped commands with a runnable example changelog-20260914-9cd42abb

02 **Shell widget handoff extensions** IMPROVED 80

Warp extends shell widget handoff to fzf's `ctrl-t` file search, alongside the existing `ctrl-r` history-search handoff, and Fish's `ctrl-r` now hands off to PatrickF1/fzf.fish history search rather than only junegunn fzf.

Trigger fzf file search inside a Warp session to quickly locate and insert a file path into your prompt.

> 📍 Press ctrl-t in a Warp terminal session to invoke the fzf file-search widget and select a file path to insert at the cursor.

— Names exact keybindings and tools with a usage example changelog-20260914-9cd42abb

03 **Editable keybindings for Attach file and Create New Window** IMPROVED 65

Warp added `Attach file` to the Command Palette as an action with an editable, unbound-by-default keybinding, and made `Create New Window` an editable keyboard shortcut that can now be remapped or unbound via Settings > Keyboard Shortcuts.

Remap or disable the Create New Window shortcut so it does not conflict with a competing binding in your workflow.

> 📍 In Warp, open Settings › Keyboard Shortcuts, search for 'Create New Window', and assign your preferred key combination or clear the binding entirely.

— Names both shortcuts with navigation steps to remap `changelog-20260914-9cd42abb`

THINNER COVERAGE BELOW

04 Automatic Factory MCP access for Claude Code and Codex NEW

40

Claude Code and Codex runs can now access Factory MCP automatically.

— Thin one-line description with no mechanism detail `changelog-20260914-9cd42abb`

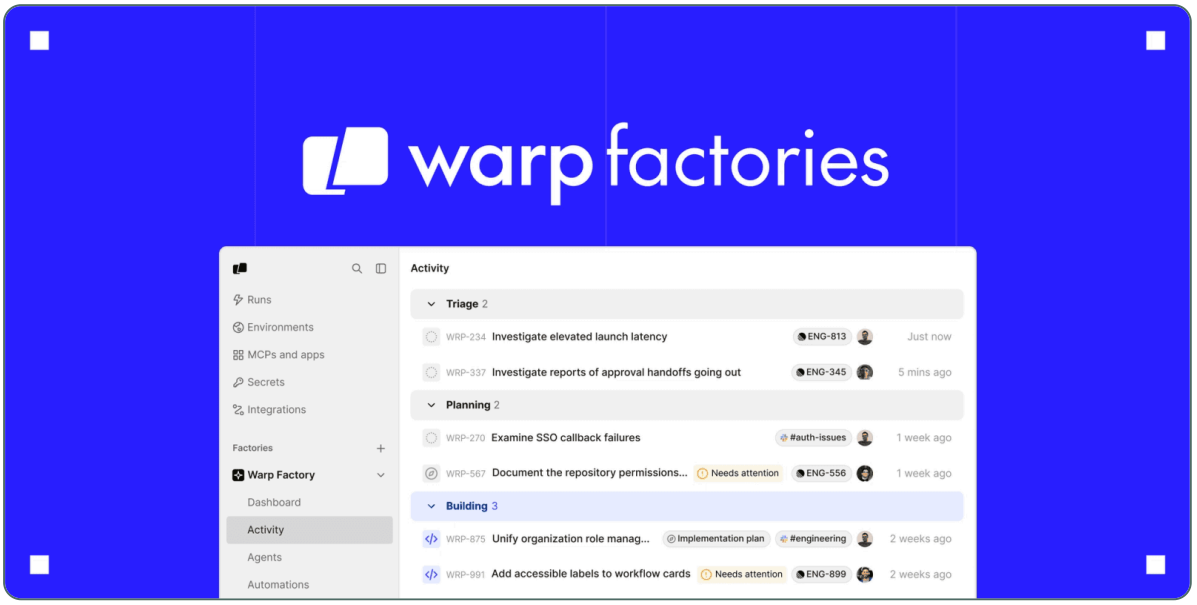
05 Refreshed Entra credentials for Azure DevOps writes IMPROVED

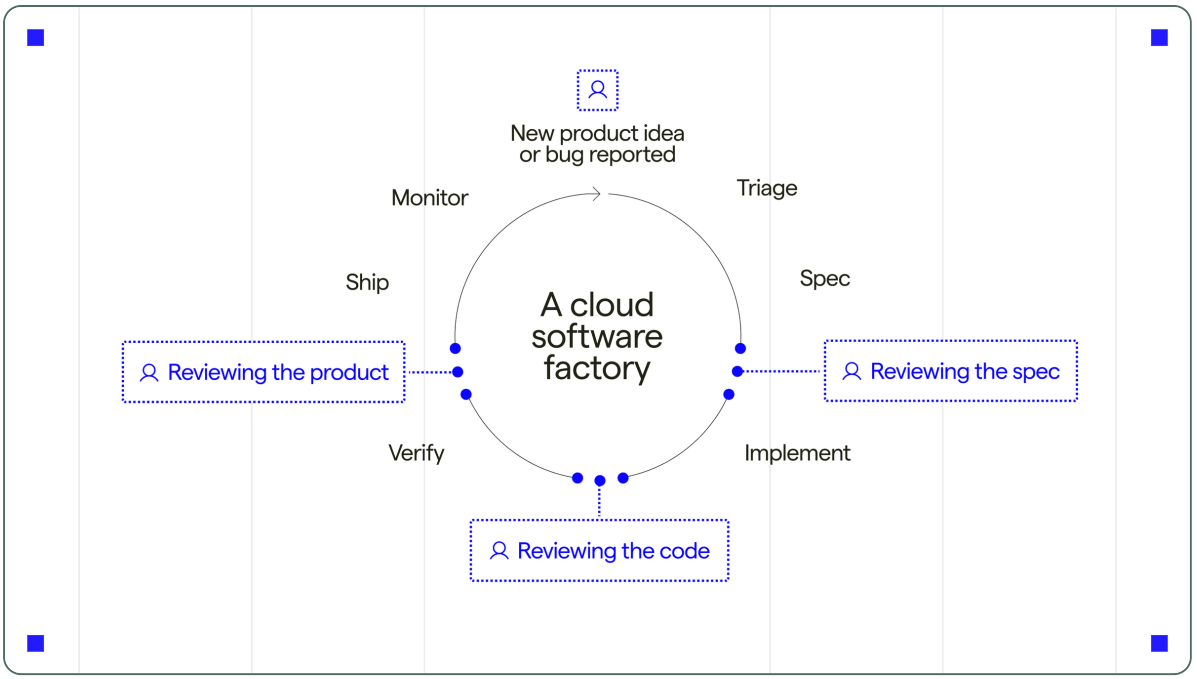
40

Oz agents can use refreshed Entra credentials to write to Azure DevOps.

— Brief note with no configuration or command detail `changelog-20260914-9cd42abb`

└──▷ ALSO FROM THESE RELEASES





[software factories as code]

```yaml
# factory.yaml
schemaVersion: v1alpha1
name: pr-review
repositories:
    owner: acme
    name: web
agentDefaults:
  model: claude-5-fable-high

# agents/foreman/agent.md
agentType: FOREMAN

# agents/review/agent.md
agentType: REVIEW
model: glm-5.2-fireworks

# automations/on-pr/automation.md
agent: review
triggers:
  - provider: github
    event: pull_request_ready
```

WAS THIS USEFUL?

StackBlitz bolt.new

1 RELEASE · 2026-09-14 | CHANGELOG ↗

RANK

Bolt.new is an AI-powered web development platform that builds websites and applications from prompts.

> bolt.new shipped Bolt Slides for building, presenting and exporting slide decks in-app, direct visual editing in the preview pane that only spends tokens on save, and Bolt Forge for open-source model access on paid plans.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **Bolt Slides presentation builder** NEW 75

Bolt Slides lets users build, customize, share, and present slide decks directly within bolt.new. A 'Speaker view' shows presenter notes, the next slide, and a timer in a separate tab while the audience sees the fullscreen deck, and speaker notes support rich-text formatting (bold, italic, headings, lists). Decks can be exported as JSON via Download > JSON to reuse the deck's content in another Bolt project.

Export a finished Slides deck as JSON to reuse its content in another Bolt project.

> 📍 In your Slides project, open the deck, then select Download › JSON to save the deck file locally.

Present to an audience while keeping your notes and a timer visible only to you.

> 📍 In your Slides project, start the presentation and enable 'Speaker view' — your tab shows notes, the next slide, and a timer; the audience tab shows fullscreen slides.

— Concrete export and speaker-view steps given, no API or config surface

`changelog-20260914-050f2bb3`

THINNER COVERAGE BELOW

02 **Bolt Forge open-source model access** NEW 35

Bolt Forge gives Pro and Lite plan subscribers access to open-source AI models for additional usage beyond standard token limits.

— Names eligible plans and benefit but no usage steps `product docs`

WAS THIS USEFUL? 👍 👎

Vercel v0 ⓘ changelog · Sep 14, 2026 CHANGELOG ↗

The v0 generative UI platform builds web applications from natural-language prompts.

└──▷ **HOW TO FIND IT**

Set the workspace default so all new chats are open to every team member for editing, removing the need to manually share each chat.

📍 Settings → Workspace → Default Chat Visibility → select 'Team can edit'

WAS THIS USEFUL? 👍 👎

Cognition Devin Desktop ⓘ 1 RELEASE · 2026-09-14

CHANGELOG ↗ RANK

Devin Desktop is an agent-native code editor for developing software locally, remotely, and in containerized environments.

> Devin Desktop added SSH-based remote sessions, an inline PR review tab, and batch archiving in the agent inbox, while making ACP always-on and fixing trigger-menu whitespace handling.

└──▷ **WHAT SHIPPED · 6 FEATURES** most completely described first ? what's the number?

01 **Inline PR review tab** NEW 73

Clicking Open Devin Review on a PR card opens a review tab that shows the PR title, switches the sessions sidebar to a PR files/review sidebar, and opens any GitHub link at its diff anchor. `Cmd/Ctrl+F` searches within the diff.

Review a pull request diff inline and jump straight to a specific comment anchor from the PR card.

> 📍 On the PR card, click Open Devin Review — the tab shows the PR title, the sessions sidebar switches to the PR files/review sidebar, and any GitHub link you open lands at its diff anchor. Use `Cmd/Ctrl+F` to search within the diff.

— Names exact UI action and keyboard shortcut for using it. changelog-20260914-6da7d82c

02 **SSH remote session support** NEW 71

From the location selector, choosing the SSH option and entering host details lets Devin open a remote SSH session — for example inside a hardened build server — instead of running locally. A status dot in the sidebar reflects the connection, and the full transcript is restored automatically if the connection drops.

Connect to a remote build server so Devin runs inside your hardened environment rather than locally.

> 📍 In the location selector, choose the SSH option, enter your host details, and select the saved host — Devin will open an SSH session, show a status dot in the sidebar, and restore the full transcript automatically if the connection drops.

— Clear step-by-step mechanism and UI path, no numeric limits given.

changelog-20260914-6da7d82c

THINNER COVERAGE BELOW

03 **Trigger menu whitespace fix** IMPROVED 30

Trigger menus (!) no longer open when followed by whitespace, and stay closed after clicking into the conversation.

— States the fix but no broader context or scope. changelog-20260914-6da7d82c

04 **Reset migration from Windsurf command** NEW 28

The command palette gains a Reset migration from Windsurf action, shown in a screenshot but not otherwise described.

— Only a screenshot alt text; no explanation of behavior. changelog-20260914-6da7d82c

05 **Devin Cloud selector** NEW 19

A new selector for choosing Devin Cloud is shown in the UI, referenced only by a screenshot with no accompanying description.

— No detail beyond a screenshot caption. changelog-20260914-6da7d82c

06 **Adaptive model picker** NEW 19

An adaptive model picker appears in the UI, referenced only by a screenshot with no further description.

— No detail beyond a screenshot caption. changelog-20260914-6da7d82c

└──▷ **BREAKING ON UPGRADE**

! The Enable ACP toggle is removed; ACP is now always on and cannot be disabled.

WAS THIS USEFUL? 👍 👎

Letta Code

1 RELEASE · 2026-09-14 NOTES ↗ RANK

Letta Code is a terminal coding agent with persistent memory and identity across development sessions.

> Letta Code's only update this window makes multi-agent workflows easier to follow by giving new subagents distinct sci-fi names.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **Sci-fi names for new subagents** IMPROVED 28

Newly created subagents are now assigned distinct sci-fi names, making them easier to identify across multi-agent workflows.

— Describes the change but no mechanism or naming scheme detail. v0.32.8

WAS THIS USEFUL? 👍 👎

Cline ⓘ 1 RELEASE · 2026-09-13 NOTES ↗ CODE ↗ RANK

Autonomous coding agent as an SDK, IDE extension, or CLI assistant.

> Cline's desktop app added Crusoe as a new OpenAI-compatible provider, sampled OpenTelemetry tracing for AI SDK calls, and clearer failure messaging for voice input and provider authentication.

└──▷ **WHAT SHIPPED · 4 FEATURES** most completely described first ? what's the number?

01 Sampled OTLP tracing for AI SDK calls NEW

88

Adds `OTEL_TRACES_EXPORTER`, `CLINE_TRACE_SAMPLE_PERCENT`, and `CLINE_TRACE_RECORD_CONTENT` environment variables to enable sampled AI SDK tracing exported via the host's OTLP exporter, for debugging or auditing model interactions through an existing observability stack.

Enable sampled OTLP tracing for Cline AI SDK calls to debug or audit model interactions via your existing observability stack.

```
$ OTEL_TRACES_EXPORTER=otlp
OTEL_EXPORTER_OTLP_ENDPOINT=http://localhost:4318
CLINE_TRACE_SAMPLE_PERCENT=10 CLINE_TRACE_RECORD_CONTENT=true
cline 'Refactor auth module'
```

— Names exact env vars with a runnable example command. desktop-v0.0.27

02 Crusoe provider and API Providers settings rename NEW

60

Adds Crusoe as an OpenAI-compatible provider in API Providers settings, and renames the settings sidebar section from 'Models' to 'API Providers' with a new icon.

— Names provider and UI section but no config detail. desktop-v0.0.27

THINNER COVERAGE BELOW

03 Voice input failure toast instead of redirect IMPROVED

55

Voice input failures now surface as a 'Speech input failed' toast in chat showing the actual error, preserving the draft text and microphone button for retry, instead of redirecting to Settings → Voice.

— Clear before/after behavior, no command or config. desktop-v0.0.27

04 Provider-specific auth failure messaging IMPROVED

55

Auth failures on the Cline provider now show a single actionable 'Sign in to Cline' button; other providers show 'Open model settings'; local-CLI providers retain their CLI link.

— Names exact button labels per provider type, no command. `desktop-v0.0.27`

WAS THIS USEFUL? 👍 👎

mex ⓘ 1 RELEASE · 2026-09-13 CODE ↗ RANK

Mex indexes codebases through a code graph and supplies a repo-local Markdown wiki to AI coding agents.

> Mex v0.8.2 makes browser-based Hub setup the default entry point (a breaking change from terminal-first workflows), adds Relay and Inbox collaboration workflows plus new Hub health/jobs/activity views and a first-run tour, and extends the code graph with a C# extractor and a Next.js App Router route resolver.

└──▷ **WHAT SHIPPED · 8 FEATURES** most completely described first ? what's the number?

01 Browser Hub setup replaces terminal defaults BREAKING

94

Bare `mex setup` now opens a browser-based Hub instead of terminal setup, and bare `mex` opens the Hub rather than the terminal dashboard; use `mex setup --cli` or `mex tui` to retain the previous terminal behavior. The Hub lets users choose integrations, review and commit setup files via Commit setup, and optionally install the `mex` command globally without leaving the Hub. Setup also gained `npx mex-agent@0.8.2 setup --no-open` to print the local Hub link without opening a browser, `--port <n>` to choose a loopback port for SSH/headless environments, `--mode agent-memory` to set up a homelab or long-running infrastructure agent workspace instead of a code repository, and `--dry-run` to preview setup changes without applying them.

Run setup over SSH without triggering a browser open, and bind the Hub to a specific loopback port.

```
$ npx mex-agent@0.8.2 setup --no-open --port 7420
```

Apply the agent-memory template for a homelab or long-running infrastructure agent workspace instead of the default code-repo mode.

```
$ npx mex-agent@0.8.2 setup --mode agent-memory
```

Run terminal-only setup in a headless or CI environment where a browser cannot be opened.

```
$ mex setup --cli
```

Preview the setup steps without making any changes, useful for auditing what mex would configure before committing.

```
$ mex setup --dry-run
```

— Full before/after, every flag named with runnable commands. v0.8.2

02 **Index rebuild for joining existing repos** NEW 80

Adds `npx mex-agent@0.8.2 graph rebuild` and `npx mex-agent@0.8.2 wiki rebuild-index` so teammates joining an existing MEX 0.8 repository can build their own local derived indexes without regenerating shared project memory, plus `npx mex-agent@0.8.2 hub` to open the local Hub afterward.

Join a team repository already using MEX 0.8 by rebuilding local indexes and opening the Hub without regenerating shared project memory.

```
$ npx mex-agent@0.8.2 graph rebuild && npx mex-agent@0.8.2 wiki rebuild-index && npx mex-agent@0.8.2 hub
```

After cloning a repo already using MEX, rebuild local indexes and open the Hub to pick up the team's shared memory without regenerating canonical files.

```
$ npx mex-agent@0.8.2 graph rebuild && npx mex-agent@0.8.2 wiki rebuild-index && npx mex-agent@0.8.2 hub
```

— Names exact commands and gives a runnable join sequence. product docs

03 **C# code graph extraction** NEW 75

Adds a C# language extractor (`CSharpWalker`) covering classes, structs, interfaces, enums, methods (including constructors, destructors, static, and async), properties, fields (`const` mapped to `constant`), parameters, namespaces (block and file-scoped including nesting), `using` imports, attributes as `decorates` , and call/instantiates/extends/implements edges — registered for `.cs` files using `tree-sitter-c-sharp.wasm` from `tree-sitter-c-sharp@0.23.5` .

— Exhaustive mechanism detail but automatic, no user-facing command. v0.8.2

04 **Next.js App Router route resolver** NEW 75

Adds a bounded Next.js App Router route resolver turning `app/**/route.ts|js` modules (including `src/app` roots) into route nodes — one per exported HTTP handler (`GET` through `HEAD`) — with URL paths derived from the route file directory, dynamic segments such as `[id]` and catch-alls preserved verbatim, and route groups like (marketing) excluded; Pages Router, layouts, and pages are out of scope.

— Precise scope and behavior named, but no user command to invoke. v0.8.2

---

05 **First-run Hub onboarding tour** NEW 74

Adds a first-run Project Hub tour that spotlights the live sidebar (Search, Project, Teamwork, System, Settings, and Context) once per checkout, with completion recorded in `.mex/local/hub-onboarding.json` so it survives Hub relaunches; the tour can be replayed from Settings.

— Names the config file and replay path, no runnable command. v0.8.2

---

06 **Relay workflow for engineer handoffs** NEW 62

Adds a Relay workflow for engineer-to-engineer handoffs: progress, decisions, blockers, evidence, and next actions are captured in canonical `.mex/relays/**` files, reviewed in the Hub, explicitly published, and then shared via Git.

— Describes mechanism and file path but no exact command. product docs

THINNER COVERAGE BELOW

---

07 **Hub Health, Jobs, and Activity views** NEW 53

Adds Hub Health and Jobs views exposing index status and explicit maintenance tasks, plus an Activity view showing accepted MEX workflow events and recorded project notes.

— Names three views and their contents but no exact navigation given. product docs

---

08 **Inbox workflow for agent-proposed knowledge** NEW 44

Adds an Inbox workflow for agent-proposed knowledge contributions: agents prepare proposals that engineers review and accept before changes are written to the Wiki.

— Explains the review flow but no named surface or command. product docs

└──▷ ALSO FROM THESE RELEASES

Set up MEX for a homelab or long-running infrastructure agent workspace instead of a code repository.

```
$ npx mex-agent@0.8.2 setup --mode agent-memory
```

└──▷ BREAKING ON UPGRADE

! Bare `mex setup` now opens browser setup by default instead of terminal setup; use `mex setup --cli` to get the previous terminal behavior. Bare `mex` now opens the Hub or setup rather than the terminal dashboard; use `mex tui` to get the terminal dashboard.

WAS THIS USEFUL? 👍 👎

Superset

1 RELEASE · 2026-09-13 | NOTES ↗ | RANK

Superset is an agentic IDE to orchestrate 100+ coding agents in parallel. Run any agent with your own subscription.

> Superset desktop v1.29.0 adds first-class Linux support, several new coding-agent integrations, per-file staging in the Changes sidebar, and a batch of mobile app upgrades including Lock Screen agent status, over-the-air updates, and in-app PR review.

└──▷ **WHAT SHIPPED · 16 FEATURES** most completely described first ? what's the number?

01 Pages linking, embedding and mobile reading updates IMPROVED

80

Adds a Pages row in Settings > Links with external/internal and split/tab open modes (SUPER-2315, SUPER-2192); adds an 'Open in Superset' deep link on the published page header (SUPER-2193); adds a Google Fonts stylesheet to the page CSP (SUPER-1995); flags delete requests on their pin and tightens comment chrome in Pages; and enables reading and commenting on pages from the phone.

Configure how Pages links open — externally or internally, in a split pane or a tab — via Settings > Links.

> 📍 In the desktop app, go to Settings › Links, find the Pages row, and choose external/internal and split/tab open mode.

— Names exact settings path, tickets, and CSP change `desktop-v1.29.0`

02 New coding agent integrations NEW

65

Adds Muse Code and Devin CLI as new agents, plus OpenCode sqlite history and OpenCode quota accounts.

— Names four specific agent/account additions, no setup steps `desktop-v1.29.0`

THINNER COVERAGE BELOW

03 Per-file staging in Changes sidebar NEW

55

Adds per-file Stage / Unstage actions in the v2 Changes sidebar.

— Names exact UI location but no further mechanism `desktop-v1.29.0`

04 Lock Screen and Dynamic Island agent status NEW

50

Adds agent status on the Lock Screen and Dynamic Island, keeping the Lock Screen agent card current via APNs pushes.

— Names mechanism (APNs) but no setup steps desktop-v1.29.0

05 **First-class Linux desktop support** NEW 48

Adds first-class Linux window chrome, menu, quit, and packaging support to the Superset desktop app.

— Names components covered but no launch/install steps given desktop-v1.29.0

06 **Named terminal sessions** NEW 45

Enables naming a terminal session on desktop and phone.

— Clear capability, minimal detail on how desktop-v1.29.0

07 **Over-the-air mobile updates** NEW 45

Adds over-the-air updates via EAS Update, signed, with workflow lanes.

— Names EAS Update and signing but no user action desktop-v1.29.0

08 **In-app PR screen from mobile terminal links** IMPROVED 45

Enables terminal PR links to open the in-app pull request screen on mobile.

— Clear navigation path, little mechanism detail desktop-v1.29.0

09 **Net revenue retention tile in admin dashboard** NEW 45

Adds a net revenue retention tile from Stripe Sigma to the admin dashboard.

— Names data source and dashboard location desktop-v1.29.0

10 **Mobile attachment upload via cloud storage** IMPROVED 43

Enables mobile upload of attachments to cloud storage instead of through the relay.

— States before/after mechanism but no config detail desktop-v1.29.0

11 **Sandbox infrastructure moved to Vercel** IMPROVED 35

Moves sandboxes from Blaxel to Vercel.

— Names vendors but no user-facing action or impact desktop-v1.29.0

12 **Realtime nudge channel and relay presence** IMPROVED 35

Adds a realtime nudge channel, eliminating sidebar polling, and adds presence from the relay, eliminating roster polling.

— Explains mechanism but purely internal, no user action desktop-v1.29.0

13 **Usage button visibility setting** NEW — 35

Adds a setting to show the Usage button in the desktop sidebar.

— Names exact toggle location, thin on purpose `desktop-v1.29.0`

14 **Multiple PR author filters** NEW — 30

Adds support for multiple PR author filters in the desktop UI.

— Thin one-line description with no mechanism `desktop-v1.29.0`

15 **Mobile version gate and server notices** NEW — 25

Adds a version gate and server-driven notices to the mobile app.

— Bare description with no mechanism or scope `desktop-v1.29.0`

16 **Automation continues its own agent session** NEW — 20

Lets an automation continue its own agent session.

— Bare one-line description with no mechanism `desktop-v1.29.0`

WAS THIS USEFUL?  

Worktrunk ⓘ 3 RELEASES · 2026-08-27 → 2026-09-08 NOTES ↗ RANK

Worktrunk manages Git worktrees from the command line for parallel AI coding-agent workflows, with hooks, status listings, and LLM-written commits.

> Worktrunk reworked `wt switch --execute` to spawn programs directly via literal argv instead of shell strings, shipped a Pi AI agent integration, and overhauled `wt list`'s JSON output schema and display formatting, alongside several breaking changes to config keys, plugin install commands, and the library API.

└──▷ **WHAT SHIPPED · 16 FEATURES** most completely described first ? what's the number?

01 Argv-based process spawning for `wt switch --execute` BREAKING

95

`wt switch --execute` (`-x`) now names one program with literal argv after `--`, spawning it as a child process instead of passing a shell string; use `-x sh -- -c '…'` to recover shell syntax. Project aliases and hooks can now use `wt switch --execute` too, with the command-approval gate remaining the control on project-defined commands. The rework retires the `WORKTRUNK_DIRECTIVE_EXEC_FILE` and `WORKTRUNK_SHELL` environment variables.

Launch Claude in a new worktree without exposing your shell environment — pass arguments as literal argv instead of a shell string.

```
$ wt switch --create feat -x claude -- --no-telemetry
```

When you need shell syntax inside `--execute` (e.g. piping or variable expansion), delegate explicitly to sh rather than relying on the old shell-string behaviour.

```
$ wt switch feat -x sh -- -c 'npm run build && npm test'
```

— Exact commands and retired env vars named v0.76.0

02 Pi AI agent integration NEW

90

Adds `wt config plugins pi install` to integrate the Pi AI agent, writing an activity hook to `~/.omp/agent/hooks/pre/worktrunk.ts` so Pi sessions show markers in `wt list`. Honors `$PI_CONFIG_DIR`, `$PI_CODING_AGENT_DIR`, `$OMP_PROFILE`, and `$PI_PROFILE`.

Install Pi agent integration so Pi coding sessions surface activity markers in `wt list`.

```
$ wt config plugins pi install
```

— Full install command, hook path, and env vars all named `v0.77.0`

03 **`wt list --format=json` schema and field changes** BREAKING

90

`wt list --format=json` now defaults to schema 2, an envelope with repository metadata and per-item facts, without needing `[list] json-schema` configuration; set `json-schema = 1` to keep the original bare-array format. It also gains a `marker` field reflecting `wt config state marker`. Separately, `wt list --format json` no longer contacts a forge or generates summaries from `[list] columns` by default: CI data now requires `--full`, and summaries additionally require `[list] summary = true` plus a configured generator; schema 1 has lost its config-driven `ci` and `summary` fields.

Pin a repo to the legacy bare-array JSON format if existing tooling parses `wt list --format=json` output directly.

```toml
json-schema = 1
```

Get full CI status per branch in JSON for scripting without triggering forge API calls or summary generation.

```
$ wt list --format json --full
```

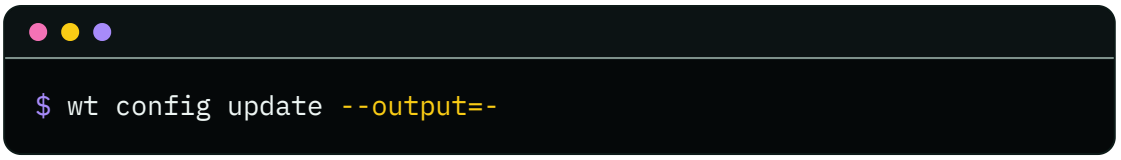
— Schema versions, flags, and config keys fully named `v0.77.0` `v0.75.0`

04 **Output control and key handling in `wt config update`** BREAKING

85

Adds `--output <path>` and `--output=-` flags to `wt config update` to write migration artifacts to a file or stdout instead of applying them in place, and now drops keys that the destination section cannot hold during migration, reporting each removed key. The old `--print` flag has been removed in favor of `--output=-`.

Preview a config migration as a diff to stdout before applying it, useful in CI or code review.



— Flags and removed --print named, with example command v0.77.0

05 **Removal of commit template-file config keys** BREAKING

85

`commit.generation.template-file` and `squash-template-file` are removed; their values must be moved into `template` or `squash-template` respectively. There is no automatic migration — the old keys now warn as unknown on every load and the prompt reverts to the built-in default. Setting both `template` and `template-file` previously failed the load; it now loads `template` silently.

— Named config keys and migration behavior, no example command v0.76.0

06 **Display and formatting improvements in `wt list`** IMPROVED

80

`wt list` caps the Branch column at 32 characters and elides long names, hides the Remote⇅ column in repos with no remote, and aligns columns by value type. It labels hidden columns by name (e.g. `hidden: Path, Commit`), marks detached worktrees with `⊘`, and leaves prunable row cells blank. In a bare repo with no worktrees it outputs `○ No worktrees` with a hint, lists branches with `--branches`, and emits an envelope with empty `items` under `--format=json`.

— Many named UI details and a flag, but no example commands v0.77.0

07 **Unified diff view in `wt switch` picker** IMPROVED 75

The `wt switch` interactive picker now opens on a single unified diff combining committed, staged, unstaged, and untracked changes; Alt-1 through Alt-8 retain direct access to subsidiary views, and Tab skips empty ones.

— Named keybindings and behavior, navigable but no command v0.75.0

08 **Faster concurrent teardown in `wt step prune`** IMPROVED

70

`wt step prune` speeds up concurrent teardowns by overlapping dirty checks, fsmonitor shutdown, and trash renames — only `git worktree remove` teardowns still serialize — reducing the `prune_e2e/live` benchmark median from 377 ms to 222 ms.

— Concrete mechanism and benchmark numbers, no user action needed v0.76.0

09 **Upstream tracking logic for new branches** IMPROVED 70

New branches now get an upstream only when the branch name matches the remote it starts from, regardless of `branch.autoSetupMerge` — e.g. `--create release --base origin/release` tracks, while `--create feature --base origin/release` does not.

— Mechanism and example scenario given, no runnable example id v0.76.0

10 **Diagnostics and approvals section in `wt config show`** IMPROVED

65

`wt config show` now exits 1 on an unreadable or invalid config, an invalid `[list] columns`, or an invalid `approvals.toml`, and surfaces a new APPROVALS section counting project commands awaiting approval.

— Exit codes and new section named, no example command v0.77.0

11 **Library API surface changes** BREAKING

65

`RemoteRefProvider` and its provider structs are replaced by `ForgeKind` dispatch; `ShellEscapeMode` fish and PowerShell escapes and `WORKTRUNK_SHELL_ENV_VAR` are retired; `BranchRef` replaces `full_ref` / `worktree_path` with `id`; and accessors `Branch::unset_upstream` and `ProjectConfig::ci_platform` are removed.

— Named API symbols but no usage guidance v0.76.0

12 **Install/uninstall symmetry for Codex and Claude plugins**

60

BREAKING

`wt config plugins codex install` now also installs the Codex plugin directly (previously only registered the marketplace), and `codex uninstall` and `claude uninstall` are now full inverses of their install counterparts — `codex uninstall` now removes the installed plugin, and `claude uninstall` now also removes the marketplace.

— Named commands and behavior change, no example v0.77.0

THINNER COVERAGE BELOW

13 **Reflinked-copy reporting in `wt step copy-ignored`** NEW

50

`wt step copy-ignored` now reports whether each copy was reflinked or a full copy in its text summary, and `--format=json` gains matching `reflinked` and `written` fields.

— Named JSON fields but no example or flag shown v0.77.0

14 **Automation interfaces graduate to stable** IMPROVED

50

`wt step eval`, `wt step for-each`, `wt step prune`, LLM branch summaries, `wt config state vars`, and `commit.generation.template-append` graduate from experimental to stable.

— Lists graduated commands but no usage detail `v0.76.0`

15 **Git 2.43 minimum version requirement** BREAKING 40

Git 2.43 is now the minimum supported version; running Git-dependent commands on an older Git version exits with an upgrade message, though `wt config shell` remains available.

— States requirement, minimal actionable detail `v0.75.0`

16 **Cosmetic header change in `wt config approvals add`** IMPROVED 30

`wt config approvals add` now opens with a cyan `Review 1 command for repo:` header instead of a yellow warning; the execution-time approval gate itself is unchanged.

— Minor cosmetic change, thinly described `v0.76.0`

└──▷ **BREAKING ON UPGRADE**

! `wt list --format=json` defaults to schema 2 (envelope with repository metadata); callers expecting the bare-array schema 1 format must set `json-schema = 1` in `[list]` config.

! `wt config show` now exits 1 on an unreadable/invalid config or invalid `[list] columns` or `approvals.toml`; it previously always exited 0.

! `wt config update --print` has been removed; use `--output=-` to write to stdout instead.

! `wt config plugins codex uninstall` now also removes an installed plugin (previously left the plugin installed); `wt config plugins claude uninstall` now also removes the marketplace.

! Templates that reference `{{ branch }}`, `{{ base }}`, or `{{ target }}` in a detached worktree now error instead of silently resolving to `HEAD`.

! The `-x` / `--execute` flag on `wt switch` now takes a program name with literal argv after `--` instead of a shell string; existing `-x` invocations that passed a shell string will break.

! `WORKTRUNK_DIRECTIVE_EXEC_FILE` and `WORKTRUNK_SHELL` environment variables are retired by the `--execute` rework and will no longer be read.

! The config keys `commit.generation.template-file` and `squash-template-file` are removed; their values must be moved into `template` or `squash-template` respectively. There is no automatic migration — the old keys now warn as unknown on every load and the prompt reverts to the built-in default. Setting both `template` and `template-file` previously failed the load; it now loads `template` silently.

! Library API: `RemoteRefProvider` and its provider structs are replaced by `ForgeKind` dispatch; `ShellEscapeMode` fish and PowerShell escapes and `WORKTRUNK_SHELL_ENV_VAR`

are retired; `BranchRef` replaces `full_ref` / `worktree_path` with `id`; and accessors `Branch::unset_upstream` and `ProjectConfig::ci_platform` are removed.

! Git 2.43 is now the minimum supported version; running Git-dependent commands on an older Git version exits with an upgrade message (`wt config shell` remains available).

! `wt list --format json` schema 1 loses the config-driven `ci` and `summary` fields; CI data now requires `--full`, and summaries require `[list] summary = true` plus a configured generator.

Amazon Kiro ⓘ 1 RELEASE · 2026-09-14 CHANGELOG ↗ RANK

Kiro is an agentic development environment that uses spec-driven workflows to plan, build, and maintain software.

> Kiro's latest release centers on governance and lifecycle controls — administrators can now push fleet-wide security policies without a restart, app manifests gain new declarative fields, and Crew Members moves behind a preview flag — alongside a large set of breaking changes touching sandboxing, knowledge management, speech-to-text, snapshots, and permission grants.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **New config keys for agent run and turn limits** NEW 75

Kiro adds `agent.chat_turn_timeout_secs` to extend a chat turn up to four hours, `agent.subagent_timeout_secs` to extend subagent runs up to three hours, and `agent.subagent_max_turns` to raise the turn count up to the documented tool-call ceiling.

– Exact config keys and limit values given changelog-20260914-ab873c74

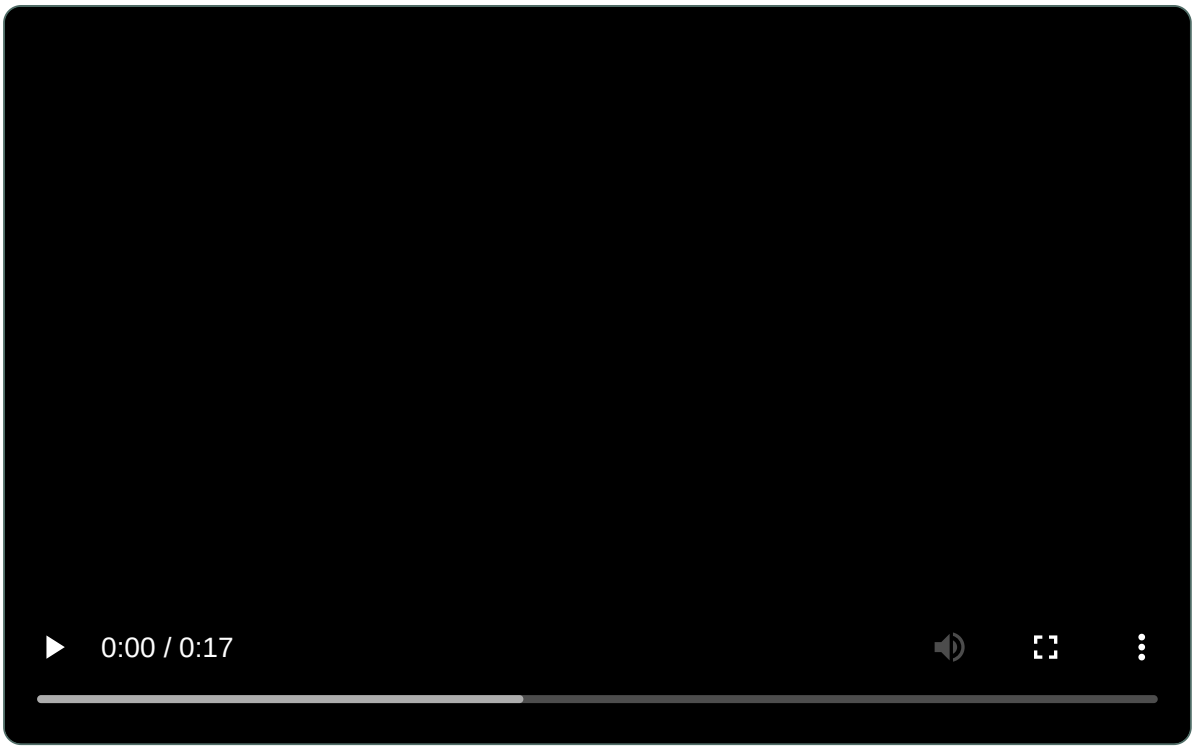
THINNER COVERAGE BELOW

02 **Kiro CLI bundle-path override removed** BREAKING 45

Kiro CLI bundle-path environment overrides from earlier releases are no longer read; Kiro Crew now uses Kiro CLI's published relay.

– Names removed override generically, no env var name given changelog-20260914-ab873c74

└──▷ **ALSO FROM THESE RELEASES**



└──▷ BREAKING ON UPGRADE

- ! Python 3.12 is now the minimum required version; hosts on Python 3.10 or 3.11 must upgrade before installing or updating Kiro Crew 0.6.0.
- ! `agent.sandbox` in `config.json` must not be `off` if you rely on the command gate — credential-path protection no longer uses command-text inspection.
- ! Unattended auto-run plans now stop after two hours by default; raise `orchestrator.max_plan_duration_seconds` in `config.json` to restore longer runs.
- ! Automatic knowledge folders are removed; folders must be explicitly added and confirmed before appearing in the Library.
- ! Developing an app UI outside its install root via the dashboard or API is no longer permitted; use `kirocrew app dev <name> --confirm-out-of-install-root` instead.
- ! The `whisper`, `mlx`, `parakeet`, and `faster` speech-to-text providers are retired; saved settings fall back to the new single `local` provider, which downloads the 148 MB `base` model on first use (a stored `turbo` setting still works but downloads 1.6 GB).
- ! `kirocrew snapshot --to s3://...`, `--aws-profile`, and the `s3://` fetch path are removed; use the AWS Control app for cloud backup instead.
- ! App trust is now tied to the repository you approved; legacy grants that cannot be matched to a repository will prompt for re-consent.
- ! Commands that previously 'looked like' a help or version check no longer receive implicit approval, and durable always-allow grants no longer cover structured non-shell tools.
- ! Kiro CLI bundle-path environment overrides from earlier releases are no longer read; Kiro Crew now uses Kiro CLI's published relay.
- ! The standalone Knowledge page is removed from the sidebar; content moves to `Agent Capabilities > Knowledge & instructions`, and old `/knowledge` links redirect there.
- ! The standalone Auto-Triage Pipeline app and its saved repository are retired; its boards move into Issue Radar as a fourth board.

! Disconnecting a connection revokes its local OAuth grant; reconnecting requires sign-in again unless another agent or scope still shares the endpoint.

! A misspelled `sandbox` key in a governance security policy now fails validation, and a malformed `publish` section denies publishing instead of silently dropping the restriction.

! Crew Members is now a preview opt-in; enable `Settings > Developer > Feature Previews > Crew Members and Crew Mode` to restore the Crew Members entry.

OpenAI Codex CLI

1 RELEASE · 2026-09-14 NOTES ↗ CODE ↗

RANK

OpenAI Codex CLI runs an agent in the terminal that reads, changes, and tests code in local repositories.

> OpenAI Codex CLI's alpha release adds trusted enterprise MCP authentication (`EmaAuth`, `codex mcp login`), turns on worktrees and voice conversations by default, expands the agent command center with model grouping, token usage and read-only history views, and removes the `thread/rollback` API.

└──▷ **WHAT SHIPPED · 8 FEATURES** most completely described first ? what's the number?

01 Trusted enterprise auth for MCP servers NEW

85

Adds trusted enterprise MCP auth configuration (`EmaAuth`) managed through Codex account sign-in, with `codex mcp login` blocking manual OAuth flows for enterprise-managed servers.

— Names config key and command but no setup walkthrough. `rust-v0.155.0-alpha.4`

02 Agent command center enhancements IMPROVED

60

Adds model grouping to the agent command center, shows task tokens and usage estimates in the command center, opens tasks managed elsewhere as read-only history, returns to the command center after session cancellation or deletion instead of exiting the TUI, and renders Markdown in agent overview task details.

— Lists five concrete UI changes but no config surface. `rust-v0.155.0-alpha.4`

THINNER COVERAGE BELOW

03 Voice conversations default with Windows bundling IMPROVED

45

Enables TUI voice conversations by default and bundles native voice runtimes in Windows releases.

— Describes default enablement and platform bundling, no flags named.

`rust-v0.155.0-alpha.4`

04 TUI streaming and theming refinements IMPROVED

45

Previews streaming prose in the TUI before a newline arrives, adds consistent theme-based thread colors across the TUI, and delays automatic recaps while separating next actions in a more compact layout.

— Describes UI polish without named config or flag. `rust-v0.155.0-alpha.4`

05 **Managed network policy in Windows MXC sandbox** NEW 40

Adds managed network policy support to the Windows MXC sandbox.

— Names sandbox but omits policy configuration specifics. `rust-v0.155.0-alpha.4`

---

06 **Removal of thread/rollback API** BREAKING 40

The `thread/rollback` API has been removed.

— Names removed endpoint but gives no migration guidance. `rust-v0.155.0-alpha.4`

---

07 **Worktrees enabled by default** IMPROVED 35

Enables worktrees by default for task isolation.

— States default change but no config or flag detail. `rust-v0.155.0-alpha.4`

---

08 **MCP server capability status reporting** NEW 25

Exposes advertised MCP server capabilities in status responses.

— Bare description with no mechanism or endpoint named. `rust-v0.155.0-alpha.4`

└──▷ **BREAKING ON UPGRADE**

! The `thread/rollback` API has been removed.

WAS THIS USEFUL? 👍 👎

AGENT

◆ AI Agent Frameworks

CopilotKit OpenBot ⓘ 1 RELEASE · 2026-09-13 NOTES ↗ RANK

OpenBot runs self-hosted AI coworkers that each get their own browser, files, and gated tools, with every action decided and recorded.

> OpenBot v0.0.10 introduces generative UI so Bots can render A2UI interfaces, tables, and forms, and hardens the platform with strict 400 validation across API write endpoints, a fixed permission bypass on component decision-functions, and more reliable message and token handling.

└──▷ WHAT SHIPPED · 6 FEATURES most completely described first ? what's the number?

01 Stricter validation across API write endpoints IMPROVED

85

Adds 400-returning validation across several endpoints: `POST /api/plugins/skills` requires `slug`, `title`, and `instructions` as non-empty strings and `summary` absent or a string, naming the bad field before any store write; `GET /channels` and `GET /api/admin/people` reject non-integer or malformed `?limit=` values instead of silently coercing them; `DELETE /api/plugins/grants` refuses a blank or whitespace-only `ref` or Bot with no delete and no audit row; the four computer gesture endpoints (click, type, key, scroll) refuse payloads missing required fields, such as a click with no coordinates or a key with no key, rather than sending them as a no-op; and validation is extended to the agent tool-call endpoint's name and arguments, policy dry-run and audit-event list limits, a component's publication flag, sandboxed-component fields, and catalogue entries.

Register a skill via the API — the endpoint now enforces that slug, title, and instructions are non-empty strings and summary is absent or a string, returning a 400 with the bad field named before any write.

```
$ curl -X POST https://<host>/api/plugins/skills \
  -H 'Content-Type: application/json' \
  -d '{"slug":"summarise-pr","title":"Summarise
PR","instructions":"Summarise the pull request and list action
items.","summary":"PR summary skill"}'
```

— Names every endpoint and field, with a runnable curl example. v0.0.10

02 Generative UI with A2UI interfaces, tables, forms NEW

80

Bots can now render A2UI interfaces, sortable comparison tables, and submission forms, with LangGraph receiving the component schemas needed to draw these interfaces. This is enabled by default; set `OPENBOT_GENERATIVE_UI` to `false` or `0` to disable it.

Disable generative UI in a deployment where you do not want Bots rendering A2UI interfaces, tables, or forms.

```env
OPENBOT_GENERATIVE_UI=false
```

— Names the env var and exact disable values, with a config example. v0.0.10

03 **Permission bypass fix on decision-functions field** IMPROVED

60

Closes a permission bypass on the component decision-functions field: a payload like `{"functions":[123,null,{}]}` no longer coerces to an empty list and returns `allowed`; the verdict is now based on the functions actually named, or a 400.

— Clear before/after and payload shape, but no runnable example given. v0.0.10

04 **Numeric environment variable fallback validation** IMPROVED

60

Fractional or out-of-range values for numeric environment variables such as `PORT`, `COMPUTER_MAX_BROWSERS`, and timeout settings now take the documented fallback instead of binding or computing with a broken value; `PORT` additionally enforces the 1–65535 range.

— Names specific env vars and the enforced range, no example command shown. v0.0.10

THINNER COVERAGE BELOW

05 **Supervisor swaps computer on token mismatch** IMPROVED

55

The supervisor now replaces a computer whose `COMPUTER_TOKEN` does not match the one currently in use, preserving profile and workspace volumes so the Bot resumes with its logins and files intact.

— Names the token and preserved state but is an automatic behavior, not user-triggered. v0.0.10

06 **Refused tool calls surfaced to LangGraph model** IMPROVED

40

The LangGraph Bot now reports refused tool calls with the HTTP status and deployment reason rather than telling its model the tool returned nothing.

— States the change but no concrete status codes or reasons given. v0.0.10

WAS THIS USEFUL? 👍 👎

HyperResearch

1 RELEASE · 2026-09-11 NOTES ↗ RANK

Agent-driven research knowledge base. Agents collect, search, and synthesize web research into a persistent, searchable wiki.

> HyperResearch v0.11.0 ships a major security hardening pass — an SSRF gate with per-type response size caps and stricter TLS certificate handling — alongside a new eight-source scholarly discovery command (`hpr scholar search`) and Parallel Search as a fifth, keyless web provider.

└──▷ **WHAT SHIPPED · 6 FEATURES** most completely described first ? what's the number?

01 **Scholarly discovery via hpr scholar search** NEW 100

Adds `hpr scholar search` and `hpr scholar sources` subcommands that query OpenAlex, Crossref, CORE, DOAB, `ClinicalTrials.gov`, SEC EDGAR, and FRED through a single cache-first, rate-limited client layer and return one deduplicated list. Records are deduplicated by DOI first, then normalized title within ±1 year, surfacing the highest citation count and longest abstract in `also_in` when multiple providers confirm the same work. Scholarly specialist records carry a `work_type` tag (trials, filings, economic series) so the pipeline never treats a 10-K as a paper, and when CORE cannot confirm a full-text version, `oa_source` is set to `core` with `oa_version` left unset, surfacing an explicit 'version unrecorded' banner instead of implying version of record.

Search all eight scholarly sources at once and inspect what is wired and available before running a full research session.

```
$ hpr scholar sources
hpr scholar search 'zero-trust network architecture' --limit 10
```

— Runnable command plus full dedup and tagging mechanism. v0.11.0

02 **SSRF protection gate for fetches** NEW 90

A new SSRF gate (`web/safe_http.py`) blocks fetches to private, loopback, link-local, multicast, CGNAT (RFC 6598), and unspecified address space across the builtin provider, crawl4ai entry points, and PDF and image downloads. The `[fetch] allow_private_hosts` config key (hostnames or CIDRs) acts as an escape hatch to permit fetches to self-hosted mirrors and intranet sources that would otherwise be refused.

Allow fetches to an internal documentation mirror and cap HTML responses to 5 MiB to prevent memory exhaustion from large intranet pages.

```toml
[fetch]
allow_private_hosts = ["docs.internal.corp", "192.168.10.0/24"]
max_html_bytes = 5242880
max_pdf_bytes = 26214400
max_image_bytes = 2097152
```

— Names exact config key, file, and blocked address classes. v0.11.0

---

03 **Per-type response size caps for fetches** NEW 80

Adds `[fetch]` config keys `max_html_bytes` (default 10 MiB), `max_pdf_bytes` (default 25 MiB), and `max_image_bytes` (default 2 MiB) to enforce response-size caps that are streamed mid-body, preventing memory exhaustion from oversized responses.

Allow fetches to an internal documentation mirror and cap HTML responses to 5 MiB to prevent memory exhaustion from large intranet pages.


```toml
[fetch]
allow_private_hosts = ["docs.internal.corp", "192.168.10.0/24"]
max_html_bytes = 5242880
max_pdf_bytes = 26214400
max_image_bytes = 2097152
```


— Exact config keys with default values and a config example. v0.11.0

---

04 **Authentication env vars for scholarly providers** NEW 80

Adds `HYPERRESEARCH_CONTACT_EMAIL` to replace the `mailto=research@example.com` placeholder, unlocking polite-pool access on OpenAlex and Crossref and enabling SEC EDGAR (which rejects requests without a contact address). Adds `FRED_API_KEY` to authenticate against FRED, with requests bypassing the SQLite cache entirely to avoid writing the key in plaintext. Adds `CORE_API_KEY` to activate CORE as a third open-access full-text resolver after Unpaywall and Europe PMC.

— Three exact env vars named with what each unlocks, no usage example. v0.11.0

---

05 **Parallel Search as fifth web provider** NEW 70

Adds `[web] provider = "parallel"` to enable Parallel Search as a fifth web provider — the first search-capable provider requiring no API key.

Enable keyless web search via Parallel Search when you have no Exa or Tavily API key.

```toml
[web]
provider = "parallel"
```

— Exact config value given, but no mechanism detail beyond that. v0.11.0

THINNER COVERAGE BELOW

06 **Distinct TLS certificate failure handling** NEW 55

Uses `CertVerificationError` as a distinct exception type for TLS certificate failures on PDF fetches, preventing automatic unverified-TLS retries that a MITM could force.

— Names the exception type but gives no config or command to act on. v0.11.0

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ AI Model & Data Infrastructure

Fireworks AI ⓘ changelog · Sep 14, 2026 CHANGELOG ↗

Fireworks AI provides hosted inference, model fine-tuning, and deployment APIs for open-weight models.

└──▷ TRY IT

Tag a deployment with an environment label using the new `firectl` subcommand, replacing a bare key that would now be rejected by the REST API.

```
$ firectl deployment tag set <deployment-id> environment=production
```

List all customer-managed tags on a deployment to verify the `custom/` namespace is in use after migrating from bare keys.

```
$ firectl deployment tag list <deployment-id>
```

└──▷ BREAKING ON UPGRADE

! Customer-managed annotation keys in REST API writes must now begin with `custom/` (e.g. `custom/environment`); bare keys such as `environment` now return HTTP `PERMISSION_DENIED`.

! `GetDeployment` and `ListDeployments` now return only `custom/*` annotation entries for regular account users; any reads relying on bare-key annotations will silently receive no data for those keys.

WAS THIS USEFUL? 👍 👎

Perplexity API

2 RELEASES · seen 2026-09-14 CHANGELOG ↗ RANK

Perplexity API Platform - build with the Router, Agent, Search, and Embeddings APIs. Real-time, web-wide research and Q&A capabilities for your products.

> Perplexity API added support for the Gemini 3.8 Flash model at promotional pricing, OAuth-based sign-in for its remote MCP server, and custom MCP connector registration for the Agent API.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 **Gemini 3.8 Flash model support** NEW 80

Adds `google/gemini-3.8-flash` as a supported model, usable via both the chat completions endpoint (`https://api.perplexity.ai/chat/completions`) and the Agent API endpoint (`https://api.perplexity.ai/agent`), with promotional pricing offering low cached-input token costs during the launch window.

Query the new Gemini 3.8 Flash model via the Perplexity API during the promotional pricing window to take advantage of low cached-input token costs.

```
$ curl https://api.perplexity.ai/chat/completions \
  -H 'Authorization: Bearer <your-api-key>' \
  -H 'Content-Type: application/json' \
  -d '{
    "model": "google/gemini-3.8-flash",
    "messages": [{"role": "user", "content": "Analyze this log
file for anomalies."}]
  }'
```

Use Gemini 3.8 Flash for cost-sensitive agentic tasks that need fast reasoning at promotional token rates.

```
$ curl https://api.perplexity.ai/agent \
  -H 'Authorization: Bearer <your-api-key>' \
  -H 'Content-Type: application/json' \
  -d '{
    "model": "google/gemini-3.8-flash",
    "messages": [{"role": "user", "content": "Summarize the
latest threat intel reports."}]
  }'
```

— runnable curl commands given but no pricing figures `changelog-20260914-5fd6bf3c` `snapshot-20260914`

02 **OAuth sign-in for remote MCP server** NEW 80

Connect an OAuth-capable MCP client (e.g. Claude Code, Cursor, or VS Code) to the remote Perplexity MCP server at `https://api.perplexity.ai/mcp` without managing API keys — sign in with your Perplexity account and select which API organization to bill during the OAuth flow.

Connect an OAuth-capable MCP client to the remote Perplexity MCP Server without managing API keys — sign in with your Perplexity account to select the billing org.

📍 In your MCP client (e.g. Claude Code, Cursor, or VS Code), add the Perplexity MCP server URL and select OAuth as the auth method. Sign in with your Perplexity account and choose which API organization to bill when prompted. No API key needs to be pasted or stored.

Connect the Perplexity MCP server to an OAuth-capable MCP client without managing an API key — billing org is chosen at sign-in.

📍 1. In your MCP client (e.g. Claude Code, Cursor, or VS Code), add a new remote MCP server. 2. Set the server URL to: `https://api.perplexity.ai/mcp` 3. Choose 'Sign in with Perplexity' when prompted for authentication. 4. Select the API organization to bill during the OAuth flow.

— exact server URL and sign-in steps, no token/scope detail `changelog-20260914-5fd6bf3c` `snapshot-20260914`

03 **Custom MCP connector registration in Agent API** NEW

75

Adds `type: "connector"` support in Agent API requests, letting you reference a registered remote MCP server by its generated connector ID instead of passing credentials per-request. Custom MCP connectors are registered via the Project connectors page, which stores server credentials in Perplexity and supports API-key or no-auth authentication over Streamable HTTP or SSE transport.

— names field and UI path but no example request shown `snapshot-20260914`

WAS THIS USEFUL? 👍 👎

Together AI

2 RELEASES · seen 2026-09-14 | CHANGELOG ↗ | RANK

Run, train, and serve open-source AI models on Together AI.

> Together AI added preemptible GPU node support with graceful shutdown handling for dedicated clusters, custom LoRA adapter serving, and a new DeepSeek-V4.1-Flash serverless model, while removing an RL training-session API and a CLI flag.

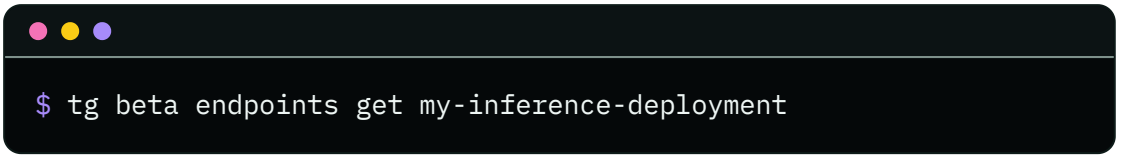
└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 Preemptible GPU nodes with graceful shutdown NEW

72

Dedicated clusters can now run interruptible workloads on discounted preemptible GPU nodes. When a preemptible node is reclaimed, Together AI emits a `TogetherPreemptionNotified` Kubernetes event and sends SIGTERM with up to 300 seconds of grace, enabling checkpoint-and-exit workflows.

Set a preemptible GPU target when creating a cluster via the CLI to reduce GPU costs using spare capacity.



– Names event and grace period but CLI example doesn't clearly match. snapshot-20260914 changelog-20260914-dc5432c3

02 DeepSeek-V4.1-Flash added to serverless NEW

63

`deepseek-ai/DeepSeek-V4.1-Flash` is now available on serverless with a 1,000,000-token context window, FP8 quantization, function calling, and structured outputs, priced at $0.30/$1.20 per 1M input/output tokens.

– Full model specs and pricing given but no usage example. changelog-20260914-dc5432c3

THINNER COVERAGE BELOW

03 Custom LoRA adapter serving on dedicated clusters NEW

35

Dedicated clusters can serve custom LoRA adapters uploaded from Hugging Face or S3.

– Thin description with no config or command example. snapshot-20260914

└──▷ **BREAKING ON UPGRADE**

! Removed endpoint GET `/rl` /training-sessions/{session_id} `/operations` /forward/{operation_id}

! Removed endpoint POST `/rl` /training-sessions/{session_id} `/operations` /forward

! The `--scale-to-zero-window` flag has been removed from the CLI.

WAS THIS USEFUL? 👍 👎

Nebius AI Cloud ⓘ 1 RELEASE · 2026-09-14 BLOG ↗ RANK

Nebius AI Cloud provides GPU compute, managed Kubernetes, distributed training infrastructure, and serverless AI workloads.

> Nebius AI Cloud's single release this window centers on two new Object Storage previews—bucket inventory reporting and filesystem buckets—alongside expanded compute options including a new London region, an NVIDIA RTX PRO 6000 VM platform, and a new CPU preset.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 **Bucket inventory in Object Storage** NEW 66

A preview feature, available in all regions, that lets you schedule daily or weekly reports listing bucket objects in `CSV` or Apache Parquet format, delivered to a destination bucket, for auditing, analytics, or compliance workflows.

— Names formats and schedule options but no API/CLI syntax given

September 7–13, 2026 (2026-09-14)

02 **New London region, GPU platform, and CPU preset** NEW 65

Adds the `uk-south2` (United Kingdom, London) public region; the `gpu-rtx6000-a` virtual machine platform with NVIDIA RTX PRO 6000 GPUs and Intel Granite Rapids CPUs, available in `uk-south2`; and the `2vcpu-8gb` preset for the `cpu-d3` (Non-GPU AMD EPYC Genoa) platform, available in all regions.

— Names exact region, platform, and preset identifiers September 7–13, 2026 (2026-09-14)

THINNER COVERAGE BELOW

03 **Filesystem buckets in Object Storage** NEW 48

A preview feature that lets you attach an Object Storage bucket to an existing shared filesystem so the same data can be read and written over both an S3-compatible API and the filesystem interface.

— Describes mechanism but no exact command or config key September 7–13, 2026 (2026-09-14)

WAS THIS USEFUL? 👍 👎

HeyGen HyperFrames

1 RELEASE · 2026-09-14 NOTES ↗ RANK

HyperFrames is an open-source HTML-to-video renderer that runs in AI-agent workflows.

> HyperFrames added an `npx hyperframes init` scaffolding command that generates a starter project with a centered title on a dark frame, using faster local capture and `looks` quality encoding by default.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 Project scaffolding via `npx hyperframes init` NEW

75

The new `npx hyperframes init` command scaffolds a centered title on a dark frame, using the faster local capture path and defaulting encode quality to `looks`.

— Runnable command named but no further mechanism detail given. v0.8.38

WAS THIS USEFUL? 👍 👎

OpenRLHF ⓘ 1 RELEASE · 2026-09-14 NOTES ↗ RANK

OpenRLHF trains language models with reinforcement learning and preference optimization using Ray, vLLM, and DeepSpeed.

> OpenRLHF v0.11.2 adds FlashREINFORCE, a critic-free asynchronous RL algorithm selectable as a PPO advantage estimator.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **FlashREINFORCE advantage estimator** NEW 88

A new `--algo.advantage.estimator flash_reinforce` option enables FlashREINFORCE, a critic-free, single-rollout asynchronous RL algorithm that uses a binary-KL trust region on vLLM logprobs with sample-mean aggregation and divergence-gated importance-sampling correction, removing the critic model to reduce GPU memory pressure.

Run a FlashREINFORCE async agent training job instead of PPO to eliminate the critic model and reduce GPU memory pressure.

```
$ bash examples/scripts/train_flash_reinforce_ray_agent_async.sh
--algo.advantage.estimator flash_reinforce
```

— Names exact flag, mechanism and runnable script command v0.11.2

WAS THIS USEFUL? 👍 👎

Thinking Machines Lab Tinker ⓘ 1 RELEASE · 2026-09-09

NOTES ↗ RANK

Tinker provides a managed API for training and sampling language models while users control their training algorithms and data.

> Tinker's SDK 0.27.2 release adds a way to cleanly terminate training sessions and release resources.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **ServiceClient.close() for session termination** NEW 80

Adds ServiceClient.close() to finalize a training session with a status of `'success'`, `'errored'`, or `'interrupted'`, plus an optional detail string. It stops heartbeats and releases HTTP and telemetry resources, and returns a future that can be resolved via .result() or awaited.

— Names exact method, status values, and return behavior. SDK 0.27.2

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

vMLX ⓘ 1 RELEASE · 2026-09-14 NOTES ↗ RANK

The vMLX server runs compressed MLX models on Apple Silicon with disk caching, paged memory, continuous batching, and hybrid SSM scheduling.

> vMLX v1.6.58 sharpens remote API session handling, adjusts default speculative-decoding behaviour for Qwen Flash-Next, and tightens error handling in the Ollama gateway and prefill pipeline.

└──▷ **WHAT SHIPPED · 6 FEATURES** most completely described first ? what's the number?

01 Qwen Flash-Next MTP default and ceiling behaviour IMPROVED

64

New Qwen Flash-Next desktop sessions now start with MTP Off by default; existing saved MTP choices and other model families' defaults are preserved. When MTP is enabled, the selected depth acts as a ceiling, with switching decisions accounting for warmup and probing cost.

— Names model, setting, and switching mechanism v1.6.58

THINNER COVERAGE BELOW

02 Remote reasoning controls and tool-call ID handling IMPROVED

53

Remote Chat Completions and Responses sessions now send protocol-specific reasoning controls and preserve tool-call IDs even when providers stream the ID separately from the function name and arguments.

— Explains mechanism but no config surface named v1.6.58

03 Ollama gateway typed errors and control preservation IMPROVED

53

The Ollama gateway now preserves media controls, reasoning effort, and notices, and returns typed errors; malformed tool argument objects are rejected without invented replacement arguments.

— Names several preserved fields and error behaviour v1.6.58

04 Model-declared context and output limits IMPROVED

48

Context and output admission now use the loaded model's declared limits, without a family-wide 256K clamp.

— Clear before/after but no config key given v1.6.58

05 **Prefill cancellation at worker-safe boundaries** IMPROVED

40

Prefill now supports cancellation at worker-safe boundaries and reports completed work promptly.

— States mechanism but no interface to trigger it v1.6.58

06 **Remote throughput reporting as request-window rate** IMPROVED

28

Remote throughput is now labelled as an observed request-window rate, with counts accumulated across tool continuations.

— Thin description, no way to act on it v1.6.58

WAS THIS USEFUL? 👍 👎

mlxtop

1 RELEASE · 2026-09-11 NOTES ↗ RANK

The mlxtop terminal UI monitors local LLMs on Apple Silicon, showing loaded models, memory use, GPU activity, and generation speed.

> mlxtop's v1.0.0 debut ships a full `top -style` terminal monitor for local LLMs on Apple Silicon, combining a one-shot CLI snapshot, three interactive TUI views, real-time oMLX integration, and process detection across six local-LLM backends.

└──▷ **WHAT SHIPPED · 6 FEATURES** most completely described first ? what's the number?

01 **One-shot text report via --once** NEW 80

Adds `mlxtop --once` to print a one-shot text report of memory, paging, GPU, and model information — usable in a terminal or over SSH — then exit.

Get a quick snapshot of memory, GPU, and model state over SSH — no interactive TUI needed.

```
$ mlxtop --once
```

— Exact flag with a runnable command example. v1.0.0

02 **oMLX integration for real-time generation metrics** NEW 80

Integrates with oMLX to expose prompt and generation speed, request queues, and cache activity in real time. Connection settings are read from `~/.config/omlx-coding/server.env`, defaulting to `127.0.0.1:8080`.

— Named config file path and default address. v1.0.0

03 **Overview, MLX Top and Journal TUI views** NEW 65

Adds three keyboard-navigable views: Overview (press `1`) showing model status, response speed, memory use, and GPU activity for locally running LLMs; MLX Top (press `2`) for inspecting running model processes and their resource use; and Journal (press `3`) tracking request activity, resource pressure, and recovery events for the current session.

— Named keybindings and views, but only UI navigation, not a command. v1.0.0

04 **Local diagnostic logging** NEW 65

Writes diagnostic logs locally to `~/Library/Logs/mlxtop/mlxtop.log`, capturing counters and model/provider names while excluding prompts, outputs, and API keys.

— Exact log path and what it excludes, no further mechanism. v1.0.0

THINNER COVERAGE BELOW

05 **Process detection across local LLM backends** NEW 50

Supports process-level detection for MLX-LM, Ollama, llama.cpp, LM Studio, KoboldCpp, and LocalAI alongside system memory and GPU readings.

— Names six backends but no usage detail. v1.0.0

06 **Sampling pause and refresh interval controls** NEW 50

Provides interactive controls including `p` / `Space` to pause/resume sampling and `+` / `-` to change the refresh interval.

— Exact keybindings given, minimal further explanation. v1.0.0

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

Pinecone ⓘ 2 RELEASES · seen 2026-09-14 CHANGELOG ↗ RANK

Pinecone is a managed vector database that stores and queries embeddings for AI applications.

> Pinecone's biggest addition this window is dedicated read nodes indexes with recall/latency tuning parameters, alongside a breaking 2026-07 API version that introduces schema-based Documents API, a new service account secret rotation endpoint, a documentation index endpoint, and a 10x increase to the Enterprise namespace limit.

└──▷ WHAT SHIPPED · 4 FEATURES most completely described first ? what's the number?

01 Dedicated read nodes indexes with tuning parameters NEW

95

Adds dedicated read nodes indexes, covering node types, shards, replicas, and index fullness, with workflows to create from scratch or from a backup, migrate existing on-demand or pod-based indexes, size and load-test, scale by adding replicas or shards, and manage node type changes, pausing, or conversion back to on-demand. Also adds `scan_factor` and `max_candidates` query parameters on these indexes to trade recall for lower latency and higher throughput.

Lower query latency on a high-traffic dedicated read nodes index by reducing scan depth, accepting a small recall tradeoff.

```
$ curl -X POST 'https://<index-host>/query' \
  -H 'Api-Key: <PINECONE_API_KEY>' \
  -H 'Content-Type: application/json' \
  -d '{"vector": [0.1, 0.2, 0.3], "topK": 10, "scan_factor": 0.5,
"max_candidates": 200}'
```

— full workflow description plus a runnable query example with named params

snapshot-20260914

02 New 2026-07 API version with Documents API BREAKING

65

Pinecone published API version `2026-07` for both the Inference and Admin APIs (`inference_2026-07.oas.yaml` and `admin_2026-07.oas.yaml`). This version introduces schema-based indexes via the Documents API; existing code must be reviewed and migrated as described in the 'Adopt the Documents API' guide.

— names spec files and migration guide but no code sample `snapshot-20260914`

THINNER COVERAGE BELOW

03 **Serverless index namespace limit increased to 1M** IMPROVED

55

Enterprise plan now supports up to `1,000,000` namespaces per serverless index, enabling large-scale multitenancy; higher limits are available on request via Support.

— specific numeric limit but no command or config key `changelog-20260914-b1036644`

04 **Documentation index endpoint /llms.txt** NEW

50

Adds `/llms.txt` documentation index endpoint for discovering all available Pinecone documentation pages.

— endpoint named but no further usage detail `snapshot-20260914`

└──▷ **BREAKING ON UPGRADE**

! API version `2026-07` introduces schema-based indexes via the Documents API; existing code must be reviewed and migrated as described in the 'Adopt the Documents API' guide.

WAS THIS USEFUL? 👍 👎

Volcengine OpenViking

1 RELEASE · 2026-09-14 NOTES ↗

RANK

OpenViking is a context database that manages agent memory, knowledge retrieval, and skills.

> OpenViking v0.4.20 reworks Compile into a persisted, pollable task API — breaking the old `/bot` /v1/compile proxy — while adding server-side Git-based skill imports, restricted ACL inheritance for shared directories, and a wide set of new config surfaces spanning memory extraction defaults, Feishu import, plugin filtering, and an external parser bridge into the Understanding API.

└──▷ **WHAT SHIPPED · 14 FEATURES** most completely described first ? what's the number?

01 Persisted Compile task API and runtime config

BREAKING

100

Adds `POST /api/v1/compile`, which now persists, queues, and tracks Compile tasks with a returned `task_id`, replacing the old `/bot/v1/compile` proxy; task status and cancellation move to `/api/v1/tasks/{task_id}` and `/api/v1/tasks/{task_id}/cancel`, with `include_events=true` returning `result.execution_events`. New `ov.conf` keys `compile_api.base_url`, `compile_api.http_timeout_seconds`, `compile_api.poll_interval_ms`, `queue_workers.external_task.max_concurrent`, and `compile_api.gateway_token` (sent as `X-Gateway-Token`) configure an independent Compile Runtime, with Python, TypeScript `client.compile` and Go `Client.Compile` SDK methods added. Breaking: the `/bot/v1/compile` create/query/cancel endpoints and the request body's `runtime_timeout_seconds` field are removed, `ov compile` drops `--wait` / `--timeout` / `--runtime-timeout` in favor of polling the task API, and `--reason` / `reason` is renamed to `--instruction` / `instruction` (Go: `CompileOptions.Instruction`).

Poll a Compile task for its full execution event log after submitting with `ov compile` or `POST /api/v1/compile`.

```
$ curl -sS 'http://localhost:1933/api/v1/tasks/TASK_ID?include_events=true' -H 'X-API-Key: your-key'
```

Connect an independent Compile Runtime and tune concurrency and polling — add this block to `ov.conf` and restart the server.

```json
{
  "compile_api": {
    "base_url": "http://runtime.example:8080",
    "http_timeout_seconds": 10,
    "poll_interval_ms": 30000
  },
  "queue_workers": {
    "external_task": {"max_concurrent": 10}
  }
}
```

— Full endpoint, config, and breaking-change detail given `v0.4.20`

02 **Skill import selection via API and CLI** NEW 85

`ov skills add` and `ov add-skill` gain a `--list` flag to preview available skills in a collection and a `--skill` flag to select specific skills for import. Server-side, `POST /api/v1/skills` now accepts a Git URL, a `skills` selection list, and a `list_only` preview mode, and records the source repo, ref, and subdirectory to support a future `ov skills update`.

— Names CLI flags and endpoint but omits future update mechanics `v0.4.20`

03 **External parser bridge into Understanding API** NEW 85

Adds a `parser_api` block (`enable`, `host`, `api_key`, `extensions`, `http_timeout_seconds`) to `ov.conf` to bridge an external parsing service, such as LlamaParse v2, into the Understanding API, with an example bridge service provided at `examples/llamaparse-understanding-bridge`.

— Config keys and example path given, integration mechanism unclear `v0.4.20`

04 **Restricted ACL inheritance for shared directories** NEW 85

Adds `acl_mode: 'restricted'` to resource directory ACL configuration to block a node from inheriting parent-directory permissions while preserving direct grants, plus a `--acl-mode` flag on `ov acl set` for choosing `restricted` or `inherit` on shared resource directories.

Restrict a shared project subdirectory so only directly-granted users have access, blocking inherited permissions from the parent team directory.

```
$ ov acl set viking://resources/team/project-a --acl-mode
restricted --entry user:bob=read
```

— Config key, flag, and runnable command all given v0.4.20

05 **Plugin filtering and retention config** NEW 85

Adds `recallQueryFilters` and `captureFilters` under `plugin.claude_code` (and `plugin.codex`) in `~/.openviking/ovcli.conf` for ordered regex filtering of recall queries and captured content, and `commitRetentionMode: 'turn_budget'` to the OpenClaw plugin config to retain recent context by user turns and token budget on auto-commit.

— Config keys and file path named, mechanism partly explained v0.4.20

06 **Session context reset on commit** NEW 75

Adds a `reset_context` field to the `POST /api/v1/sessions/{session_id}/commit` request body to clear injected session context without affecting the Session ID, original history, or long-term memory.

— Endpoint and field named, scope of effect explained v0.4.20

07 **Line-range reads in MCP read** NEW 75

Adds `offset` and `limit` parameters to the MCP `read` operation for line-range reads, e.g. `offset: 100`, `limit: 40`, with `limit: -1` meaning no line limit.

— Parameters and example values given v0.4.20

08 **Memory extraction default format changed to Python DSL** BREAKING 75

`memory.extraction_output_format` now defaults to `python`, a restricted DSL, instead of `json`; JSON output remains available by explicitly setting `{"memory": {"extraction_output_format":"json"}}`.

— Config key and migration override given, no behavioural detail on DSL v0.4.20

09 **Server thread pool and identity flush tuning** NEW 70

Adds `server.executor_threads` to `ov.conf` to control the asyncio default thread pool size per service process, and `server.trusted_identity_flush_interval_seconds` (default 300 seconds) to control the async bulk identity registration flush interval in trusted mode.

— Two config keys named with defaults, no usage example v0.4.20

10 **Feishu import improvements** IMPROVED 70

Adds `args.feishu_recursive: true` to Feishu import to recursively import Wiki subtrees, and a `domain` configuration field on the Feishu channel config to support Lark (`https://open.larksuite.com`) alongside the default Feishu domain.

— Two config fields named, no example shown `v0.4.20`

11 **Memory maintenance and batch extraction thresholds** NEW

70

Adds `memory.maintenance_review_tokens` (default 1,000) as a threshold for triggering memory maintenance review prompts when reading large memory sets, and per-batch long-term memory extraction for long sessions driven by `pending_token_threshold` and `message_count_threshold` on the saved `auto_commit_policy`.

— Threshold keys named with default, no example usage `v0.4.20`

12 **Experimental pi context management example** NEW

65

Provides a pi experimental context management example at `examples/pi-experimental-context-management` enabling agents to archive the current context window and open a new one via `new_context` and `history`.

— Example path and function names given, limited explanation `v0.4.20`

THINNER COVERAGE BELOW

13 **Studio user memory strategy selection** NEW

55

Adds memory strategy selection (General, User Personal Memory, Agent Experience Memory, or Custom) on user creation and edit in Studio.

— UI options named, no navigation path given `v0.4.20`

14 **Agent Experience module in Studio** NEW

50

Adds an Agent Experience module to Studio for browsing experience assets, viewing related execution traces and result distributions, and tracing provenance.

— Describes capability but no concrete UI path or API `v0.4.20`

└──▷ **BREAKING ON UPGRADE**

! The `/bot/v1/compile` create, query, and cancel endpoints are removed; creation now uses `POST /api/v1/compile`, and query/cancel use `/api/v1/tasks/{task_id}` and `/api/v1/tasks/{task_id}/cancel`.

! `ov compile` no longer accepts `--wait`, `--timeout`, or `--runtime-timeout`; scripts must save the returned `task_id` and poll via the task API instead.

! The top-level `runtime_timeout_seconds` field is removed from the Compile HTTP request body.

! `ov compile` (and Python/TypeScript SDK options) no longer accept `--reason` / `reason`; the field is now `--instruction` / `instruction` (Go: `CompileOptions.Instruction`).

! `memory.extraction_output_format` defaults to `python` instead of `json`; deployments whose prompts or evaluations depend on JSON output must explicitly set `{"memory": {"extraction_output_format":"json"}}`.

WAS THIS USEFUL? 👍 👎

timgordontg engrim ⓘ 1 RELEASE · 2026-09-13 NOTES ↗ RANK

The Universal Cross-Model Episodic Memory Standard. Local-first, project-scoped SQLite engine for Google Antigravity, Claude Code, Cursor, Codex CLI, and OpenCode.

> engrim v1.4.5 adds a built-in health-check command, an enterprise server CLI, and hardens cross-platform hook reliability with automatic path detection and fallback logic.

└──▷ **WHAT SHIPPED · 4 FEATURES** most completely described first ? what's the number?

01 **engrim doctor health audit command** NEW 90

`engrim doctor` audits SQLite WAL integrity, active memories, semantic embedding models, and all configured agent hooks. It supports `--fix` for automatic self-repair and `--json` for machine-readable output.

— Names command, both flags, and exact audit scope. v1.4.5

02 **Self-healing lifecycle hook fallbacks** IMPROVED 69

Adds self-healing lifecycle hook fallbacks for Antigravity, Claude Code, and Codex using a portable PATH fallback pattern (`<bin> || engrim ... || true`) to prevent silent failures across Windows, Linux, and macOS.

— Exact fallback pattern named but no direct user action. v1.4.5

THINNER COVERAGE BELOW

03 **engrim ent enterprise server command** NEW 58

`engrim ent` lets local CLI users connect to and inspect directives in Engrim Enterprise In-VPC servers.

— Runnable command named, but connection details unspecified. v1.4.5

04 **Cross-OS hook path sanitization** NEW 42

Adds `_hook_bin` cross-OS path sanitization that automatically detects and resolves cross-platform executable path mismatches.

— Internal mechanism named but no user-facing usage detail. v1.4.5

WAS THIS USEFUL? 👍 👎

Docling

1 RELEASE · 2026-09-14 NOTES ↗ RANK

Docling parses PDFs, Office files, and HTML into structured documents with layout, tables, and reading order preserved for downstream RAG pipelines.

> Docling v2.127.0 broadens its input coverage with new MHTML and RTF backends, adds an AWS region field to S3-hosted document references, and standardizes OCR language codes to BCP-47 .

└──▷ WHAT SHIPPED · 4 FEATURES most completely described first ? what's the number?

01 New input formats: MHTML and RTF NEW

65

Docling adds an MHTML input backend, enabling parsing of web archive (`.mhtml`) files, and adds RTF input support via a LibreOffice-based backend, extending the set of office formats it can convert. Both can be converted the same way as other formats, e.g. via DocumentConverter().convert("saved_page.mhtml") or convert("report.rtf").

Convert a saved web page archive to Markdown — useful for preserving and parsing web content offline.

python

```python
from docling.document_converter import DocumentConverter

converter = DocumentConverter()
result = converter.convert("saved_page.mhtml")
print(result.document.export_to_markdown())
```

Convert an RTF document (e.g., a legacy report) to structured Markdown via the LibreOffice backend.

python

```python
from docling.document_converter import DocumentConverter

converter = DocumentConverter()
result = converter.convert("report.rtf")
print(result.document.export_to_markdown())
```

— Names both new formats and shows runnable conversion code v2.127.0

THINNER COVERAGE BELOW

02 AWS region field on S3Coordinates IMPROVED

55

Adds a `region` field to S3Coordinates for specifying the AWS region when referencing S3-hosted documents.

— Names exact field but gives no usage example v2.127.0

03 **BCP-47 OCR language canonicalization** IMPROVED 40

Canonicalizes OCR language identifiers across all OCR engines according to the BCP-47 standard, standardizing how language codes are interpreted regardless of which OCR engine is used.

— Explains behavior change but no code or config example v2.127.0

04 **Local figure embedding for JATS articles** IMPROVED 25

Embeds local figure images when parsing JATS articles.

— Bare one-line description with no mechanism or example v2.127.0

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

LangChain LangSmith ⓘ 3 RELEASES · 2026-08-20 → 2026-09-14

CHANGELOG ↗ RANK

LangSmith provides tracing, evaluation, and deployment tools for LLM applications.

> LangSmith capped SaaS trace retention at 180 days, tightened dataset-export and model-list permissions, and renamed legacy role permissions. It also added SSO deep-linking, a model-pricing MCP tool, key-less Vertex AI access via the auth proxy, new annotation-queue and evaluator-validation API endpoints, and deprecated the legacy feedback-formula endpoints ahead of removal.

└──▷ WHAT SHIPPED · 14 FEATURES most completely described first ? what's the number?

01 **redirect_to parameter for SSO login** NEW 85

The SSO login URL now supports a `redirect_to` query parameter, e.g. `/sso/login/<slug>?redirect_to=/o/<org>/projects/p/<project>/r/<run>`, landing users on the target page after SAML sign-in instead of the workspace root.

Deep-link users into a specific run after SAML sign-in, so an alert email can land reviewers directly on the failing trace.

```
$ https://<your-langsmith-host>/sso/login/<sso-slug>?
redirect_to=/o/<org>/projects/p/<project>/r/<run>
```

— Exact param and runnable example URL given changelog-20260914-66aa8d21

02 **180-day cap on SaaS trace retention** BREAKING 75

Extended retention for traces ingested into LangSmith SaaS is capped at 180 days starting September 14, 2026; existing settings above 180 days are applied as 180 days for new traces, and new settings above 180 days are rejected outright. Self-hosted and BYOC deployments are unchanged.

— Clear before/after and exact cutover date but no reader action

changelog-20260914-66aa8d21

03 **GET /v1/models scoped to access policies** BREAKING 75

`GET /v1/models` now returns only the models permitted by a workspace's model access policies, so models that would previously appear in the list but be rejected on use no longer show up.

— Named endpoint with clear before/after behaviour `changelog-20260914-66aa8d21`

04 **repos:* and run-rules:* permissions renamed** BREAKING

75

Built-in workspace roles no longer return the retired `repos:*` and `run-rules:*` permissions, which have been renamed to `prompts:*` and `rules:*`; custom role creation now rejects the old permission names.

— Exact old and new permission names given `changelog-20260914-66aa8d21`

05 **Model configurations provider option in Model Access policies**

NEW

60

Model Access policies gain a 'Model configurations' provider option, letting administrators explicitly permit a workspace's saved model configurations — either all of them or a chosen subset.

— Mechanism explained but no exact UI path or API `changelog-20260914-66aa8d21`

THINNER COVERAGE BELOW

06 **list_model_price_maps MCP tool** NEW

55

Adds the `list_model_price_maps` tool to LangSmith MCP for searching and paginating model pricing rules configured in a workspace.

— Named MCP tool but no usage example `changelog-20260914-66aa8d21`

07 **Workspace-level Agent Auth connections view** NEW

50

Workspace administrators can now view workspace-level Agent Auth connections — including authentication type and update details — from the Integrations settings page.

— UI path given but no further mechanism `changelog-20260914-66aa8d21`

08 **Vertex AI access via LLM auth proxy without stored keys** NEW

50

Organizations with the LLM auth proxy enabled can now select Google Vertex AI models in the playground and Agent Builder without storing a Vertex service account key.

— Names surfaces affected but no config steps `changelog-20260914-66aa8d21`

09 **Screen-aware, streaming LangSmith Chat** IMPROVED

50

LangSmith Chat within a tracing project now reads the active filters, view, time window, and selection from the screen, and streams its reasoning while working.

— Behaviour described but no invocation detail `changelog-20260914-66aa8d21`

10 **Overflow dropdown for chips in Experiments table** NEW

50

A clickable +N overflow dropdown on model, prompt, and tool chips in the Experiments table config cells now exposes filter, group-by, open-in-playground, and details actions.

– UI location and actions named but no exact path steps `snapshot-20260914`

---

11 **Context Hub repository file limit raised to 1,500** IMPROVED

40

Context Hub repositories now support up to 1,500 files, up from the previous lower limit.

– Concrete number but no other mechanism `changelog-20260914-66aa8d21`

---

12 **Managed Deep Agent missing-connection messaging** IMPROVED

40

Managed Deep Agent channel replies now name the missing workspace connection and show how an agent developer can configure it, instead of reporting a generic run failure.

– Behaviour change described without exact config surface `changelog-20260914-009c9b77`

---

13 **IAM role assumption for ECR image access** NEW

35

LangSmith Cloud can now assume tagged customer IAM roles for ECR repository discovery and snapshot image pulls.

– Bare description, no setup steps given `changelog-20260914-66aa8d21`

---

14 **gpt-6-astra suggested in model dropdowns** NEW

30

OpenAI model configuration dropdowns now suggest `gpt-6-astra`.

– Single-line addition with no further context `changelog-20260914-66aa8d21`

└──▷ BREAKING ON UPGRADE

! Starting September 14, 2026, extended retention settings above 180 days on LangSmith SaaS are applied as 180 days for new traces; new settings above 180 days are rejected outright. Self-hosted and BYOC deployments are unchanged.

! Dataset CSV and JSONL exports now require download permission in addition to dataset read permission; existing API clients that lack download permission will receive a forbidden response.

! Built-in workspace roles no longer return the retired `repos:*` and `run-rules:*` permissions (renamed to `prompts:*` and `rules:*`); custom role creation rejects these old permission names.

! `GET /v1/models` now returns only models permitted by your model access policies, so previously advertised models that would be rejected on use no longer appear in the list.

! Dataset CSV and JSONL exports now require both dataset read and download permissions; direct API requests without the download permission return a forbidden response instead

of succeeding.

! The legacy feedback formula endpoints (`POST/GET /feedback/formulas`, `GET/PUT/DELETE /feedback/formulas/{feedback_formula_id}`) are deprecated and scheduled for removal on 2026-08-20; migrate to composite evaluators before that date.

! Legacy dataset comparison helpers are removed from the public OpenAPI spec and generated SDKs; existing HTTP routes continue to work only for LangSmith UI clients.

Braintrust

3 RELEASES · 2026-09-01 → 2026-09-14 | CHANGELOG ↗ | RANK

Braintrust provides evaluation, tracing, and improvement workflows for AI applications.

> Braintrust's biggest window addition is two new public-preview agents — Patterns for automated failure-mode detection and Debugger for single-trace failure analysis — plus a managed-runtime upgrade to Loop with built-in models, alongside portable observability templates, sync history window controls, Java SDK eval concurrency, and a broad tightening of tracing-plugin configuration.

└──▷ **WHAT SHIPPED · 16 FEATURES** most completely described first ? what's the number?

01 Tracing plugin configuration and environment-variable removal

90

BREAKING

The `pi` and `OpenCode` tracing plugins, along with normal Claude Code and Codex sessions, no longer read environment-variable overrides for tracing enablement, profile, organization, project, or metadata (including `TRACE_TO_BRAINTRUST` and `BRAINTRUST_ADDITIONAL_METADATA`); settings must instead be saved with `bt trace enable opencode` or `bt trace enable pi`. `bt trace run codex` now rejects `--dangerously-bypass-hook-trust` to prevent duplicate Braintrust hooks. The Braintrust Claude and Codex marketplaces no longer distribute their MCP-specific plugins (including Codex's bundled routing skill) — configure Braintrust MCP directly or use a remote connector in Cowork. `bt trace enable opencode` and `bt trace enable pi` now install v2 of their respective tracing integrations (see the v0.19.3 migration instructions).

— Every affected env var, flag and command named `snapshot-20260914`

`changelog-20260914-72c92428` `changelog-20260914-53261673`

02 History window controls for bt sync

BREAKING

80

`bt sync` and `bt sync pull` now include all matching experiment and dataset history by default instead of a limited window; pass `--window` or set `BT_SYNC_WINDOW` to restore the previous windowed behavior. Project logs still retain the default three-day window.

— Exact flag and env var given, before/after behavior clear `snapshot-20260914`

`changelog-20260914-72c92428` `changelog-20260914-53261673`

03 Portable observability template export and apply

NEW

80

The CLI adds `bt observability template export` and `bt observability template apply` commands to export a project's observability setup (facets, Loop automations, etc.) as a portable JSON template and apply it to a new project, avoiding manual rebuilding.

Export a fully configured project's observability setup as a portable JSON template and apply it to a new project, avoiding manual rebuilding of facets and Loop automations.

```
$ bt observability template export --project my-llm-app >
template.json
bt observability template apply --project new-llm-app
template.json
```

— Exact runnable commands given in example `snapshot-20260914`

---

04 **Custom CA bundle trust for trace uploads** NEW 80

Setting `BRAINTRUST_CUSTOM_CA_BUNDLE` (optionally alongside `BRAINTRUST_APP_PUBLIC_URL`) lets Rust SDK tracing (e.g. via `bt trace run pi`) trust a corporate CA when uploading traces in air-gapped or private-PKI environments.

Allow the Rust SDK to trust a corporate CA when uploading traces in an air-gapped or private-PKI environment.

```
$ export BRAINTRUST_CUSTOM_CA_BUNDLE="$(cat
/etc/ssl/certs/corporate-ca.pem)"
export
BRAINTRUST_APP_PUBLIC_URL="https://braintrust.internal.corp"
bt trace run pi -- python my_agent.py
```

Trust a corporate CA bundle for all Rust SDK tracing and data uploads in a self-hosted environment.

```
$ export BRAINTRUST_CUSTOM_CA_BUNDLE="$(cat
/etc/ssl/certs/corporate-ca.pem)"
bt trace run pi
```

— Exact env vars and runnable command given `snapshot-20260914`

`changelog-20260914-72c92428`

---

05 **Loop runs in managed runtime with built-in models** IMPROVED 75

Loop now runs in a Braintrust-managed runtime, persisting threads and operating across logs, experiments, and datasets project-wide, and supporting auto-accept for writes. It runs on built-in GPT-5.6 Sol, GPT-5.6 Terra, and GPT-5.6 Luna models with no AI provider configuration required.

— Named built-in models, but no config path shown `snapshot-20260914`

06 **Concurrent eval execution in Java SDK** BREAKING 75

Java SDK eval cases now run concurrently by default (3 at a time), requiring `Task` and `Classifier` implementations to be thread-safe. A new .executor(Executor) method on `Devserver.Builder` lets callers supply a custom `Executor` to control eval concurrency.

— Exact builder method and default concurrency named `changelog-20260914-72c92428` `changelog-20260914-53261673`

07 **Error and duration filters for trace.get_spans()** NEW 70

trace.get_spans()'s `SpanFilters` gains `has_error` and `duration` fields, enabling trace-level scorers to filter spans by error state, metadata values, and execution time bounds.

— Exact field names and method given, no example `changelog-20260914-72c92428`

08 **Patterns automated failure-mode detection (public preview)** NEW 65

Patterns (public preview, requires data plane) automatically runs Loop against a project on a schedule to surface recurring failure modes, cost trends, and quality issues, saving each finding with supporting traces and optional suggested fixes.

— Clear mechanism and scope, no UI path given `snapshot-20260914`

09 **Debugger trace failure analysis (public preview)** NEW 65

Debugger (public preview, requires data plane) reports likely failure modes for a single long agent trace, citing spans, tool calls, and model outputs, alongside an 'Analyze trace' view that groups a run into Work sections.

— Names the 'Analyze trace' view as an entry point `snapshot-20260914`

10 **Expanded agent span metadata** IMPROVED 65

Agent spans now include `instructions` and `system_prompt` in span input, tools in model-call metadata, and reasoning token counts for Anthropic and Google models.

— Named fields added, no usage example `changelog-20260914-72c92428`

11 **Instrumentation wrappers for ElevenLabs and legacy Google Generative AI SDK**

NEW 60

Adds `wrapElevenLabs` instrumentation for tracing ElevenLabs text-to-speech and speech-to-text calls, including streaming and timestamped audio, and `wrapGoogleGenerativeAI` instrumentation for the legacy `@google/generative-ai` SDK (v0.24.x).

— Named wrapper functions and SDK version, no usage snippet changelog-20260914-72c92428

THINNER COVERAGE BELOW

12 **Cursor pagination for MCP search_patterns** IMPROVED 55

The MCP server's `search_patterns` tool now supports cursor pagination for listings and text searches, letting clients retrieve more than 100 matching patterns in a single query.

— No parameter name given for cursor usage changelog-20260914-53261673

13 **Progress indicator for large trace loads** NEW 35

The trace viewer now shows a progress indicator with the span fetch count and percentage complete while loading large traces.

— UI behavior described, no navigation path given changelog-20260914-53261673

14 **input_audio content type in prompt templates** NEW 35

Mustache and Nunjucks prompt template rendering now supports the `input_audio` content type.

— Bare mention of new content type, no example changelog-20260914-72c92428

15 **Google Antigravity tracing authentication migration** BREAKING 30

Google Antigravity tracing now moves authentication and trace delivery to a new mechanism, replacing plugin-owned credentials and endpoint configuration; see the migration instructions for details.

— No specifics on the new mechanism given changelog-20260914-72c92428

16 **Stable profile IDs across renames** IMPROVED 30

Tracing references now use stable profile IDs, so renaming a profile no longer breaks existing tracing configuration references.

— No mechanism detail beyond ID stability changelog-20260914-72c92428

└──▷ BREAKING ON UPGRADE

- ! The pi and OpenCode tracing plugins no longer read environment-variable overrides for tracing enablement, profile, organization, project, or metadata — save these settings with `bt trace enable opencode` or `bt trace enable pi` instead.
- ! Normal Claude Code and Codex sessions no longer read `TRACE_TO_BRAINTRUST` or `BRAINTRUST_ADDITIONAL_METADATA` from the environment.
- ! `bt trace run codex` rejects `--dangerously-bypass-hook-trust` to prevent duplicate Braintrust hooks.
- ! The Braintrust Claude marketplace no longer distributes the MCP-specific plugin — configure Braintrust MCP directly or use a remote connector in Cowork.
- ! The Braintrust Codex marketplace no longer distributes the MCP-specific plugin and its bundled routing skill — configure Braintrust MCP directly.
- ! Java SDK `Task` and `Classifier` implementations must now be thread-safe (eval cases run concurrently, default 3 at a time).
- ! `bt sync` now includes all matching experiment and dataset history unless you pass `--window` or set `BT_SYNC_WINDOW`; project logs retain the default three-day window.
- ! The `bt trace enable pi` and `bt trace enable opencode` plugins no longer read environment-variable overrides for tracing enablement, profile, organization, project, or metadata — save these settings with `bt trace enable opencode` or `bt trace enable pi` instead.
- ! `bt trace run codex` now rejects `--dangerously-bypass-hook-trust` to prevent duplicate Braintrust hooks.
- ! The Braintrust Claude marketplace no longer distributes the MCP-specific plugin; configure Braintrust MCP directly in the app or use a remote connector in Cowork.
- ! The Braintrust Codex marketplace no longer distributes the MCP-specific plugin and its bundled routing skill; configure Braintrust MCP directly in the app.
- ! Java SDK `Task` and `Classifier` implementations must now be thread-safe (eval cases run concurrently by default).
- ! Google Antigravity tracing moves authentication and trace delivery to a new mechanism, replacing plugin-owned credentials and endpoint configuration (see migration instructions).
- ! `bt trace enable opencode` and `bt trace enable pi` now install v2 of their respective tracing integrations; see the v0.19.3 migration instructions before upgrading.
- ! `bt sync pull` now includes all matching experiment and dataset history by default; pass `--window` or set `BT_SYNC_WINDOW` to restore the previous windowed behavior (project logs retain the default three-day window); see the v0.19.2 migration instructions.

Superlog Labs Superlog ⓘ changes since 2026-08-15 CODE ↗

Superlog monitors AI agent telemetry with OpenTelemetry and groups signals into incidents for observability.

Superlog adds GCP disconnect, Cloudflare Worker selection, shared auth rate limiting, and an external-content prompt boundary.

└──▷ **GET THIS VERSION**

```
$ git clone --branch commits-2026-08-15
https://github.com/superloglabs/superlog.git
# already have the repo? check out this version:
$ git checkout commits-2026-08-15
```

└──▷ **TRY IT**

Disconnect a provisioned GCP integration as a manager — removes cloud resources and revokes the ingest key while preserving stored telemetry.

```
$ curl -X POST
https://api.superlog.example/api/projects/<projectId>/gcp/disconnect-
url \
  -H 'Authorization: Bearer <token>' \
  -H 'Content-Type: application/json'
```

- Adds `POST /api/projects/:projectId/gcp/disconnect-url` and `POST /api/gcp/authorizations/:authorizationId/disconnect` endpoints for a manager-only, re-authenticated Google Cloud disconnect flow that removes provisioned sink, delivery, and monitoring resources, revokes the connection ingest key, and preserves previously stored telemetry.
- Adds `action=disconnect` support on the GCP OAuth callback, with rollback-safe recovery if the final persistence step fails.
- Adds per-Worker controls ('Wire all' and 'Unwire all') for Cloudflare connections, with manual Worker selection now the default for new Cloudflare connections.
- Enables shared per-IP rate limiting on authentication endpoints (sign-in, sign-up, password recovery, verification-email) across all environments, with support for forwarded client-IP handling on deployments behind a trusted proxy.

Comet Opik ⓘ 2 RELEASES · 2026-09-14 NOTES ↗ RANK

Opik traces, evaluates, and monitors LLM applications and agentic workflows, with datasets, an LLM-as-judge metric library, and production dashboards.

> Opik improved dataset upload performance in the Python SDK and added Gemini 3.8 Flash thinking-level controls across the playground and Vertex backend.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **Gemini 3.8 Flash thinking levels** NEW 65

Adds Gemini 3.8 Flash thinking level selection in the playground and online scoring, and sends the native `thinking_level` parameter for Vertex Gemini 3 backend calls.

— Names exact parameter and surfaces affected 2.2.60

THINNER COVERAGE BELOW

02 **Streaming dataset item uploads in Python SDK** IMPROVED 40

The Python SDK now streams dataset item uploads instead of materialising them in memory, reducing memory pressure for large datasets.

— Names mechanism but no API or size figures 2.2.61

WAS THIS USEFUL? 👍 👎

agentacct

1 RELEASE · 2026-09-14 | NOTES ↗ | RANK

The agentacct dashboard records coding-agent work steps, tools, file changes, tests, time, and token costs locally.

> agentacct v0.10.8 focuses on making agent work auditable, adding in-app recording setup, a redesigned time-anchored Work timeline, and a Work Receipt document with a markdown export command.

└──▷ **WHAT SHIPPED · 5 FEATURES** most completely described first ? what's the number?

01 **In-app recording setup for coding clients** NEW — 70

Recording setup now happens inside the macOS app: connect a coding client (Claude Code, Codex, OpenCode, or Hermes), review a read-only plan of files that will change, run setup, and confirm capture via a fresh event.

– Names the four supported clients and setup flow, but no CLI/config keys given v0.10.8

02 **Time-anchored Work activity timeline** IMPROVED — 60

Work is redesigned as a time-anchored activity timeline with drag-to-pan and scroll/pinch zoom, plus inline task details that open below the timeline when a task is selected.

– Describes interaction mechanics but only as UI navigation, no command v0.10.8

THINNER COVERAGE BELOW

03 **Recording-health notices and offline saved-work view** NEW — 30

Adds recording-health notices and a saved-work view that stays readable while the recorder is offline.

– States behaviour but no mechanism, triggers, or UI path v0.10.8

04 **Plain-text export of current review** NEW — 23

Adds a plain-text export of the current review from within the app.

– Bare one-line addition with no format or location detail v0.10.8

05 **Per-agent coverage matrix in README** NEW — 23

Adds a per-agent coverage matrix to the README for comparing agent capabilities at a glance.

– Documentation-only addition with no operational surface v0.10.8

WAS THIS USEFUL? 👍 👎

GOVERN

◆ AI/LLM Security

FailproofAI

1 RELEASE · 2026-09-13 NOTES ↗ RANK

Observability and enforcement for AI agent harnesses. Capture every run and runtime reliability with policy enforcement. 40 built-in policies, a local dashboard, no account required with a generous free cloud plan

> FailproofAI 1.0.5 adds full support for OpenClaw 2026.9, bundling new harness discovery, transcript ingestion and policy-management commands alongside model-visible PreToolUse policy enforcement.

└──▷ WHAT SHIPPED · 2 FEATURES most completely described first ? what's the number?

01 OpenClaw harness discovery and transcript ingestion NEW

90

FailproofAI now fully supports OpenClaw 2026.9 agent harnesses: `failproofai harness add-path openclaw profile-name=` registers non-default OpenClaw profile paths for discovery, `failproofai policies --install --cli openclaw --scope user` installs the FailproofAI plugin across default and named OpenClaw profiles, `failproofai backfill` recovers previously missed OpenClaw sessions into the local dashboard, and `failproofai harness list openclaw` enumerates discovered OpenClaw agents across profiles. It adds transcript ingestion from OpenClaw's per-agent SQLite databases (alongside existing JSONL support), reads committed rows from live SQLite WAL files across all supported Node.js versions, gives OpenClaw agents stable per-profile namespacing, and automatically discovers agents under the default OpenClaw profile and any configured named profile paths.

Wire up a non-default OpenClaw profile so FailproofAI discovers its agents, installs policies, and backfills any missed sessions.

```
$ failproofai harness add-path openclaw profile-
name="$HOME/.openclaw-profile-name"
failproofai policies --install --cli openclaw --scope user
failproofai backfill
```

Verify OpenClaw agent discovery and confirm all profiles and agents are visible after registration.

```
$ failproofai harness list openclaw
```

— Names every command and mechanism, with runnable usage examples 1.0.5

THINNER COVERAGE BELOW

02 **Model-visible instruct() policies for OpenClaw** NEW

55

OpenClaw `PreToolUse` instruct() policies are now model-visible: the first matching tool call is interrupted with guidance shown to the model, and a guarded retry can continue afterward.

— Explains behaviour but gives no config or command example `1.0.5`

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Frontier Models

Anthropic ⓘ 1 RELEASE · 2026-09-14 CHANGELOG ↗ RANK

Anthropic provides Claude AI models and APIs for building applications that generate, analyze, and automate work.

> Anthropic shipped a new `ant apply` CLI subcommand for declaratively syncing agents, environments, skills, memory stores and deployments from repository files, plus a way to watch and interact with running Managed Agents sessions from a browser.

└──▷ WHAT SHIPPED · 2 FEATURES most completely described first ? what's the number?

01 **`ant apply` CLI subcommand for resource sync** NEW 75

The `ant apply` subcommand (v1.30.0) creates and updates agents, environments, skills, memory stores, and deployments from repository files. It writes a `claude-lock.json` lockfile so subsequent runs — including in CI — update existing resources instead of creating duplicates.

— Names command, version and lockfile mechanism but no flags shown

changelog-20260914-a850a3bd

02 **Browser viewing and control of Managed Agents sessions** NEW 70

Using `ant beta:sessions connect <session-id> --web`, a running Managed Agents session can be watched in the browser, allowing interactive approval or denial of tool calls without leaving the local environment.

Watch a running Managed Agents session in your browser and interactively approve or deny tool calls without leaving your local environment.

```
$ ant beta:sessions connect <session-id> --web
```

— Exact runnable command given, but mechanism lightly described

changelog-20260914-a850a3bd

WAS THIS USEFUL? 👍 👎

OpenAI ⓘ changelog · Sep 14, 2026 CHANGELOG ↗

OpenAI provides AI models and APIs for building applications that generate and reason with text, images, and audio.

└──▷ **TRY IT**

Start a full-duplex voice session with GPT-Live 1 using Responses delegation, so a backend model handles reasoning while the voice channel stays open.

```
$ curl https://api.openai.com/v1/live/sessions \
  -H 'Authorization: Bearer $OPENAI_API_KEY' \
  -H 'Content-Type: application/json' \
  -d '{"model": "gpt-live-1", "delegation": "responses"}'
```

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · September 14, 2026

Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe