THE AI TOOLCHAIN

NO. 024

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED SEPTEMBER 12, 2026 · EVERY DAY

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

▸ // HOW THIS ISSUE IS MADE SHOW

VIEW

ISSUE VIEW

scroll



SCROLL | EXPAND IN PLACE | PREVIEW PANE

INFORMATION DENSITY

everything



BRIEFED | EVERYTHING

Every tool in the issue, in full.

all 347 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 44 tools

▾ // THE BRIEF HIDE SAME ISSUE, SAME PROMPT, TWO WRITERS: GPT-5.5 | Opus 5

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 today: Braintrust now investigates traces and runs automations without your own worker fleet; LangSmith

vim ? help

turns judge setup and multi-turn eval testing into API/CLI work; Composio gives agent tool integrations first-class lifecycle control for webhooks, auth, scopes, and accounts.

OPERATE

Let production traces trigger the next debugging step instead of waiting for a human sweep

Patterns automate trace investigations, and Loop automations now run in a managed runtime. That moves repeated failure triage out of notebooks and ad hoc scripts, so teams can attach investigation logic to live traces without operating their own automation runner.

Braintrust

EVALUATE

Create and regression-test judges from the same automation path as the rest of your evals

LangSmith added CLI-driven LLM-as-judge evaluator creation, multi-turn evaluator testing, and public experiment comparison APIs; Claude Code added `claude plugin eval` for scoring plugin eval suites. The manual handoff from prompt authoring to UI-only judge setup gets smaller, and agent/plugin changes can be checked against repeatable eval suites before they reach users.

LangSmith · Claude Code

BUILD

Give agents safer, reusable access to external systems without rebuilding integration plumbing

Composio added webhook lifecycle management, log search, account revocation, deferred-auth completion, OAuth scope recommendations, and custom toolkits; PydanticAI added reusable `capabilities` bundles; Hermes added a SHA-pinned plugin catalog and password-blind credential vault. Teams can package tool access, auth, and execution behavior as maintained units instead of copying brittle per-agent glue.

Composio · PydanticAI · Hermes

Does **GPT-5.5** read better? 👍 👎

BUILD

◆ AI Coding Agents

Vercel v0

Feedback ⓘ 2 RELEASES · SEEN 2026-09-12 CHANGELOG ↗

RANK

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

> v0's biggest change this window is a new team-wide default chat visibility setting that also alters default behavior for new chats (a breaking change), alongside expanded MCP server tooling, Platform API memory/skill attachment, deployment policy enforcement, and several workspace usability additions.

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

01 Default chat visibility for team workspaces BREAKING

85

Adds Settings → Workspace → Default Chat Visibility with three options — Private, Team can view, and Team can edit — letting team owners control the default visibility for all new chats. New chats created in a team workspace are now visible to the team by default; existing chats are unaffected, but teams that relied on chats being private by default must update this setting to Private.

Set all new team chats to be editable by every team member by default, so collaborators never need an explicit invite to contribute.

📍 In the v0 console, go to Settings → Workspace → Default Chat Visibility and select Team can edit.

Lock down new team chats to invitation-only access to prevent unintended exposure of sensitive work in progress.

📍 In v0, go to Settings → Workspace → Default Chat Visibility and select Private.

— Names exact setting path, options, and required migration action. snapshot-20260912

changelog-20260912-b87953cf

02 v0 MCP server tool additions NEW

75

Adds `listMessages` tool to the v0 MCP server for reading a chat's full message history, and `resolveTask` tool for answering plan approvals, questions, permission requests, and integration prompts programmatically. The MCP server also now returns scope-aware chat URLs alongside tool responses.

— Names exact new MCP tools but gives no usage example. snapshot-20260912

03 **Team skills and memories access controls** IMPROVED 65

Restrict Memories and Skills setting is now available on any paid team plan instead of Enterprise only, and team members with v0 edit access can now manage team skills and memories without requiring a Vercel write role.

— Names exact setting and permission change, lacks UI path. snapshot-20260912

04 **Team deployment policy enforcement** NEW 60

Enforces team Vercel deployment policies in v0's deploy button and GitHub import flow, hiding disallowed repositories and blocking production deploys the team forbids.

— Explains enforcement surfaces but no config key given. snapshot-20260912

05 **Team default design system settings** NEW 60

Team owners can now set a team default design system, including Design Systems 2.0 skills and built-in examples, and choose team or personal scope when saving from the save card.

— Names save-card scope options but no exact UI path. snapshot-20260912

06 **Keyboard shortcut to open Settings** NEW 60

Adds a keyboard shortcut (`Cmd+.` / `Ctrl+.`) to open Settings directly from the editor.

— Exact shortcut given, immediately usable. snapshot-20260912

THINNER COVERAGE BELOW

07 **Recent collaborator activity on project cards** NEW 45

Project cards now display avatars of teammates active in visible project chats within the last 24 hours, ordered by most recent activity.

— Describes UI change but no navigation path given. snapshot-20260912

08 **Platform API memory and skill attachment** NEW 40

The Platform API now supports attaching memory and project skills when creating a chat or sending a message.

— States capability but omits endpoint or parameter names. snapshot-20260912

└──▷ **BREAKING ON UPGRADE**

! New chats created in a team workspace are now visible to the team by default; existing chats are unaffected, but teams that relied on chats being private by default must update Settings → Workspace → Default Chat Visibility to Private .

WAS THIS USEFUL? 👍 👎

Letta Code

Feedback ⓘ 1 RELEASE · 2026-09-12 NOTES ↗ RANK

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

> Letta Code's latest release focuses on sandbox targeting improvements in the CLI, enabling file transfers across conversations and simplifying cross-computer setups.

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

01 Cross-conversation sandbox file transfers in CLI IMPROVED

35

The CLI now supports targeting another conversation when performing sandbox file transfers, and sandbox targeting has been simplified with added guidance for cross-computer scenarios.

— Names the capability but no exact flag or command syntax given. v0.32.3

WAS THIS USEFUL? 👍 👎

Charm Crush

Feedback ⓘ 1 RELEASE · 2026-09-12 NOTES ↗ CODE ↗

RANK

Terminal-based AI coding assistant that integrates LLMs, LSPs, and MCP servers into your development workflow.

> Crush v0.94.1 adds ChatGPT subscription login, a headless reasoning-depth control, and scroll performance improvements to the terminal UI.

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

01 ChatGPT OAuth login and logout NEW 80

Adds `crush login openai` (also aliased as `crush login chatgpt`) to authenticate with a ChatGPT subscription via OAuth, unlocking subscription-granted models without a separate API key. `crush logout openai` (also `crush logout chatgpt`) removes the stored OAuth credentials.

Authenticate with your existing ChatGPT subscription so Crush can use the models your plan grants, without needing a separate API key.

```
$ crush login openai
```

— Names exact commands and aliases, runnable directly. v0.94.1

02 `--reasoning-effort` flag for headless runs NEW 75

Adds a `--reasoning-effort` flag to `crush run` for non-interactive/headless use, accepting values like `max` to control model reasoning depth — useful in CI pipelines or agent orchestration.

Run Crush headlessly with maximum reasoning effort — useful in CI pipelines or agent orchestration where you want the model to think hardest.

```
$ crush run --model "hyper/glm-5.3" --reasoning-effort "max"
```

— Exact flag and example command given, missing full value range. v0.94.1

THINNER COVERAGE BELOW

03 **UI scroll performance gains** IMPROVED 13

The release includes performance improvements to UI scrolling, per the release summary, though no mechanism or metrics are given.

— Only a bare mention with no detail. v0.94.1

└──▷ **ALSO FROM THESE RELEASES**

WAS THIS USEFUL? 👍 👎

Anthropic Claude Code

Feedback ⓘ 1 RELEASE · 2026-09-11

NOTES ↗ CODE ↗ RANK

Claude Code is Anthropic's terminal coding agent that plans, edits, and tests code in local repositories.

> Claude Code v2.1.269 ships a new `claude plugin eval` command for scoring plugin eval suites, a batch of VSCode extension additions (agent map, Hooks dialog, Permission rules dialog, live subagent progress, account-switch cancel), and several new environment variables and settings covering Workflow concurrency, OpenTelemetry repository tagging, gateway timeouts, and Bash-edit diffs, plus a queued-message take-back on the web.

└──▷ **WHAT SHIPPED · 12 FEATURES** most completely described first ? what's the number?

01 OpenTelemetry repository attribute tagging NEW

81

Adds `OTEL_METRICS_INCLUDE_REPOSITORY` to tag OpenTelemetry metrics and events with `vcs.*` repository attributes; commit events additionally get `vcs.ref.head.*` attributes when `OTEL_LOG_TOOL_DETAILS` is also set.

Enrich OpenTelemetry metrics with repository context so traces are filterable by branch and commit.

```
$ export OTEL_METRICS_INCLUDE_REPOSITORY=1
export OTEL_LOG_TOOL_DETAILS=1
claude
```

— Two env vars and exact attribute namespaces named with example. v2.1.269

02 Configurable Workflow agent concurrency limit NEW

80

Adds `CLAUDE_CODE_WORKFLOW_MAX_CONCURRENT_AGENTS` (range 1–256) to raise the Workflow tool's per-run concurrent agent limit for inference-bound fan-outs.

Allow a Workflow tool fan-out to run up to 32 agents in parallel when your workload is inference-bound.

```
$ export CLAUDE_CODE_WORKFLOW_MAX_CONCURRENT_AGENTS=32
claude
```

— Env var, numeric range, and runnable example all given. v2.1.269

03 **Plugin eval scoring command** NEW 73

Adds `claude plugin eval` to run a plugin's eval suite against Claude Code and produce scored, reproducible results in JSON and HTML report formats.

Score a plugin's eval suite to validate its prompts and agent behaviour before shipping.

```
$ claude plugin eval --help
claude plugin eval
```

— Command and two report formats named with runnable example. v2.1.269

04 **Output style switching slash command** NEW 67

Adds `/output-style [name]` slash command to list and switch output styles, including over Remote Control and in cloud and other headless sessions.

— Command syntax and session scope named, no example run. v2.1.269

05 **Extended LLM gateway discovery timeout** NEW 65

Adds `CLAUDE_CODE_GATEWAY_MODEL_DISCOVERY_TIMEOUT_MS` to extend the LLM gateway `/v1/models` discovery timeout beyond the 3s default.

— Env var and endpoint named but no usage example given. v2.1.269

06 **Agent map in VSCode** NEW 65

Adds an agent map in the VSCode extension — an N agents footer pill opens a map of the session's sub-agents with per-agent cards, a Stop agent action, and read-only transcripts.

— UI mechanism detailed but only a navigation path, no command. v2.1.269

07 **Hooks dialog in VSCode command menu** NEW 63

Adds a Hooks dialog to the VSCode command menu for viewing hooks and adding, editing, or removing them in user, project, and local settings; managed, plugin, and session hooks remain read-only.

— Settings scope enumerated; UI path only, no command shown. v2.1.269

08 **Permission rules dialog in VSCode** NEW 63

Adds a Permission rules dialog that lists permission rules and adds or removes them in user, project, and local settings; startup-option, session-only, and managed rules remain read-only.

— Settings scope enumerated; UI path only, no command shown. v2.1.269

THINNER COVERAGE BELOW

09 **Bash tool file-edit diff surfacing** NEW 58

Adds `bashEditDiffEnabled` setting to surface a diff of files changed by a Bash command directly in the Bash tool result when the Bash tool handles file edits.

— Setting name given but no config file path or example. v2.1.269

10 **Take back queued message on the web** NEW 58

Adds the ability to take back a queued message in a Claude Code on the web cloud session before Claude reads it: remove it from the queue, or press `Esc o`.

— Keybinding named; scope limited to cloud sessions. v2.1.269

11 **Live subagent progress in Focus view** NEW 37

Adds live progress rows for running subagents under the tool-call groups in Focus view in the VSCode extension.

— Bare description of a UI addition, no further mechanism. v2.1.269

12 **Cancel option on Switch account screen** NEW 35

Adds a Cancel button to the Switch account screen in the VSCode extension that returns to your session as the current account.

— Single UI control described briefly, no further scope. v2.1.269

WAS THIS USEFUL?  

All Hands AI OpenHands

Feedback ⓘ 1 RELEASE · 2026-09-11

NOTES ↗ RANK

OpenHands is an open-source software-development agent that writes code, runs commands, and browses the web.

> OpenHands added OpenAI-compatible API endpoints for broader client integration and shipped several smaller improvements to automation management and the UI.

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

01 **OpenAI-compatible Chat Completions and Responses endpoints**

NEW 60

OpenHands now exposes its agent via OpenAI-compatible Chat Completions and Responses API endpoints, enabling it to be used from chat UIs, IDEs, voice platforms, and other OpenAI-style clients.

— Names the two endpoint types but not exact paths or params. product docs

THINNER COVERAGE BELOW

02 **Cloud automation editing and identity visibility** NEW

40

Automations can now be edited on cloud backends, and the system shows which identity an automation runs as, giving operators visibility into the execution context.

— Describes two thin automation-area additions without named config or API surfaces. v1.18.0

03 **Full-size image attachment preview** NEW 30

Users can now click an image attachment in the UI to open it at full size.

— Simple UI action described without further mechanism. v1.18.0

04 **Automation re-enable restricted to creator** BREAKING

30

Re-enabling an automation is now restricted to its original creator only, tightening who can resume a disabled automation.

— States the restriction but not enforcement mechanism or scope. v1.18.0

WAS THIS USEFUL? 👍 👎

OpenAI Codex CLI

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗

CODE ↗ RANK

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

> Codex CLI adds thread-scoped instructions and bundles voice support for both Linux and Windows terminals.

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

01 **Voice support bundled for Linux and Windows** NEW 60

Bundles GNU voice helpers and runtimes into Linux musl release archives, with ALSA plugin discovery and increased capture/playback buffering for PipeWire compatibility, and bundles a Windows voice runtime into portable ZIP packages, enabling voice support on Windows.

— Names package formats and audio subsystems but no usage command.

`rust-v0.155.0-alpha.3.10`

WAS THIS USEFUL? 👍 👎

Sourcegraph Amp

Feedback ⓘ 1 RELEASE · 2026-09-11 BLOG ↗

RANK

Amp is Sourcegraph's agentic coding tool for the terminal and editor, running multi-step edits with subagents and shared team threads.

> Amp introduces a Restack button that reorganizes a thread's changes into cleaner, logical commits before review.

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

01 **Restack button for commit reorganization** NEW 55

A new Restack button reorganizes a thread's changes into a cleaner set of commits — splitting large diffs into logical pieces, cleaning up intermediate agent commits, or combining small related commits — without altering final file contents.



— Names the UI action and its behaviors but no command or config.

Restack Your Changes (2026-09-11)

WAS THIS USEFUL? 👍 👎

Amazon Kiro

Feedback ⓘ 1 RELEASE · 2026-08-29 BLOG ↗ RANK

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

> Kiro's Crew release (0.5) adds an AWS Control app for fleet-wide cloud account management, fleet-wide security policy publishing, cross-session agent coordination through a built-in conductor, persistent session tabs, and a consolidated in-process speech-to-text provider — alongside a wave of breaking changes retiring older speech providers, S3 snapshot backups, the standalone Knowledge and Auto-Triage apps, and narrowing legacy approval grants.

└──▷ **WHAT SHIPPED · 22 FEATURES** most completely described first ? what's the number?

01 **Secrets migration into encrypted vault** NEW 85

Adds `kirocrew secrets import` to move plaintext credentials from `.env` into the encrypted vault, leaving a `secret://` reference behind; it runs as a dry run unless `--apply` is passed.

Safely migrate plaintext secrets out of a .env file into the encrypted vault before committing — run without `--apply` first to preview what will change.

```
$ kirocrew secrets import --apply
```

— Exact command with flag and vault mechanism described.

Crew: AWS Control, Session Tabs, and Ag…

02 **Speech-to-text provider consolidation** BREAKING 85

Retires the `whisper`, `mlx parakeet`, and `faster` speech-to-text providers in favor of a single in-process local provider that downloads its own model on first use (a 148 MB base model by default; a stored `turbo` setting still works but downloads 1.6 GB), enabling dictation without installing an external binary, with live text appearing as you speak and filtering of silence hallucinations and caption boilerplate. `kirocrew doctor` now reports stored provider defaults so users can decide which to adopt after the consolidation.

— Names retired/replacement providers, sizes, and doctor command.

Crew: AWS Control, Session Tabs, and Ag…

03 **Fleet-wide security policy publishing** NEW 75

Adds `security_policy.json` published from a URL so administrators can push one security policy to an entire fleet; hosts cache and refresh it without a restart, reject

invalid documents, and fall back to the cached policy when the source is unavailable. A misspelled sandbox key in the file now fails validation, and a malformed `publish` section denies publishing instead of silently dropping restrictions.

— Names config file and cache/fallback mechanism, no exact key list.

Crew: AWS Control, Session Tabs, and Ag…

04 **Atomic in-place update via kirocrew update** NEW 70

Adds `kirocrew update` to install a new version into a signature-verified tree and switch to it atomically; on Windows, Restart & Update shows installer progress and relaunches the new build.

— Exact command and Windows flow named; no rollout detail.

Crew: AWS Control, Session Tabs, and Ag…

05 **Permission prompt display and bulk approval controls** NEW 70

`kirocrew chat` now displays permission prompts so users can answer approval requests directly instead of waiting silently, and approval controls let users approve or reject all waiting tool calls at once, or reject one call while continuing to review the rest; long shell commands show a readable summary with the full command available on hover.

— Describes UI mechanism and bulk controls, no command names.

Crew: AWS Control, Session Tabs, and Ag…

06 **MCP server management improvements** IMPROVED 70

MCP server health view now shows a consecutive-failure count with one-click reset; remote OAuth servers can be connected from the MCP Servers page; and enterprise Kiro accounts honor administrator-managed MCP registry controls, including version pins.

— Names UI surface, OAuth connect, and registry controls.

Crew: AWS Control, Session Tabs, and Ag…

07 **Trusted phone access via Gateway** NEW 70

Adds phone access: one action enables trusted mobile access, restarts the Gateway, publishes through Tailscale, and shows a sign-in QR code; phone sessions survive Gateway restarts and updates.

— Names concrete steps: Gateway, Tailscale, QR sign-in.

Crew: AWS Control, Session Tabs, and Ag…

08 **AWS Control app for account management** NEW 65

Adds the AWS Control app, a single surface to review account health, browse a versioned S3-backed drive, manage access, inspect billing, run cloud backups, and restore snapshots by replacing or merging components with a rollback ledger.

– Names app and surfaces but no commands. Crew: AWS Control, Session Tabs, and Ag…

09 **S3 snapshot backup removed** BREAKING 65

`kirocrew snapshot --to s3://...` and the `--aws-profile` fetch path with `s3://` are no longer available; cloud backups now go through the new AWS Control app instead.

– Exact flags named; clear migration path given. Crew: AWS Control, Session Tabs, and Ag…

10 **Approval grant scope narrowed** BREAKING 65

Durable `always-allow` grants no longer cover structured non-shell tools, and commands that look like help or version checks no longer count as approval; app trust is now tied to the approved repository, so legacy grants that cannot be matched to a repository will prompt for re-consent and some commands may ask for approval again.

– Explains scope narrowing and reconsent trigger clearly.

Crew: AWS Control, Session Tabs, and Ag…

11 **Ephemeral chat modes** NEW 65

Adds Ephemeral chat modes accessible via the `+ New` menu — Incognito (reads memory, writes none) and Temporary (neither reads nor writes memory).

– Names both modes and menu entry point. Crew: AWS Control, Session Tabs, and Ag…

12 **Persistent session tabs** NEW 65

Supports persistent session tabs above the transcript, openable with middle-click or modifier-click, showing live status per tab; pinned, running, and unread sessions stay always visible while dormant sessions fold behind a configurable threshold expander.

– Describes tab behavior and thresholds, no config key.

Crew: AWS Control, Session Tabs, and Ag…

13 **Knowledge content relocated** BREAKING 60

Moves Knowledge content into `Agent Capabilities > Knowledge & instructions`; the standalone Knowledge page is removed from the sidebar and old `/knowledge` links redirect there.

– Names exact new UI path and redirect behavior. Crew: AWS Control, Session Tabs, and Ag…

THINNER COVERAGE BELOW

14 **Global workflow library invocation** NEW 55

Adds `/workflow` invocation for workflows promoted into a versioned global library.

– Names invocation and library scope only. Crew: AWS Control, Session Tabs, and Ag…

15 **Cross-session agent coordination** NEW 55

Agents can send a message into another session as its next turn, and a built-in conductor can divide a larger goal into work items, start a session per item, check results, and decide the next round.

— Describes conductor mechanism but no config surface.

Crew: AWS Control, Session Tabs, and Ag…

16 **App Store Discover, Library and Updates pages** NEW 55

App Store adds separate Discover and Library pages plus an Updates tab listing installed apps with pending updates and an Update All action.

— Names three UI surfaces, no config or command. Crew: AWS Control, Session Tabs, and Ag…

17 **OAuth grant revoked on disconnect** BREAKING 50

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

— Describes trigger and reconnection condition, no UI path.

Crew: AWS Control, Session Tabs, and Ag…

18 **Crew member direct-message threads** NEW 50

Crew member roster gives every member a durable pinned direct-message thread with recent activity, unread indicators, previews, and name search.

— Describes DM thread features, no config surface.

Crew: AWS Control, Session Tabs, and Ag…

19 **Auto-Triage Pipeline merged into Issue Radar** BREAKING 45

The standalone Auto-Triage Pipeline app and its saved repository are retired; Auto-Triage Pipeline becomes a fourth Issue Radar board that follows the selected repository.

— Names retired app and merge target only. Crew: AWS Control, Session Tabs, and Ag…

20 **Non-blocking meeting translation** NEW 40

Meeting translation translates transcript lines as they arrive without blocking the original transcript; failed translations keep the source text.

— One-line translation behavior, no mechanism detail.

Crew: AWS Control, Session Tabs, and Ag…

21 **Pull request watchers expanded triggers** IMPROVED 35

Pull request watchers now wake on comments, reviews, and verdicts in addition to build results.

— One-line trigger expansion, no mechanism detail.

Crew: AWS Control, Session Tabs, and Ag…

22 **Dev Fleet pod resource gauges and reclaim** IMPROVED

35

Dev Fleet pods show resource gauges, and closed-pull-request working trees can be reclaimed with safety checks.

— Two thin UI additions, no config or command named.

Crew: AWS Control, Session Tabs, and Ag…

└──▷ **BREAKING ON UPGRADE**

! The `whisper`, `mlx parakeet`, and `faster` speech-to-text providers are retired; saved settings fall back to the new in-process provider, which downloads the 148 MB base model on first use. A stored `turbo` setting still works but downloads 1.6 GB.

! Snapshot backup to S3 is removed: `kirocrew snapshot --to s3://...` and the `--aws-profile` fetch path with `s3://` are no longer available. Use the AWS Control app for cloud backups.

! App trust is now tied to the approved repository; legacy grants that cannot be matched to a repository will prompt for re-consent, and some commands may ask for approval again.

! Durable `always-allow` grants no longer cover structured non-shell tools, and commands that look like help or version checks no longer count as approval.

! Custom bundle-path environment overrides for the Kiro CLI are removed; Kiro Crew now uses Kiro CLI's published relay.

! The standalone Knowledge page is removed from the sidebar; its content is now under `Agent Capabilities > Knowledge & instructions`.

! The standalone Auto-Triage Pipeline app and its saved repository are retired; its boards become a fourth Issue Radar 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 `security_policy.json` sandbox key now fails validation, and a malformed `publish` section denies publishing instead of silently dropping restrictions.

WAS THIS USEFUL? 👍 👎

Warp

Feedback ⓘ 1 RELEASE · 2026-09-12 CHANGELOG ↗ RANK

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

> Warp's latest update adds Grok Build support, an fzf-powered file search handoff, broader Oz CLI team-selection controls, a new Factories software-factory automation view, and clearer Command X-Ray explanations for git flags.

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

01 fzf file search handoff in terminal NEW

80

Pressing `ctrl-t` in the Warp input invokes fzf file search, letting users select a file path to insert into a command without leaving Warp.

Trigger fzf file search from inside the terminal to quickly insert a file path into a command without leaving Warp.

📍 Press ctrl-t in the Warp input to invoke fzf file search and select a path to insert.

— Exact keybinding given, fully actionable changelog-20260912-9cd42abb

02 Warp Factories for software factory automation NEW

70

Warp introduces Factories, shown via an Activity view that groups tasks into Triage, Planning, and Building states with linked issues, implementation plans, channels, and Needs attention statuses, alongside a 'software factories as code' YAML configuration defining repositories, agent models and roles, and a GitHub pull-request trigger.

.
.
.

— Rich screenshot-based description but no runnable command shown

changelog-20260912-9cd42abb

THINNER COVERAGE BELOW

03 Command X-Ray explains git ls-remote flags IMPROVED

50

Command X-Ray now describes `git ls-remote` flags such as `--branches`, giving inline explanations of what the command options do.

— Explains flag meaning but no usage steps given changelog-20260912-9cd42abb

04 Oz CLI team-selection controls NEW

15

Warp added broad team-selection controls for the Oz CLI, but no additional detail on scope, flags, or usage is provided.

— Named feature but mechanism undescribed changelog-20260912-9cd42abb

05 **Grok Build support added** NEW 10

Warp added support for Grok Build, though no further detail on its functionality or scope is provided.

— Only named, no functional detail given changelog-20260912-9cd42abb

WAS THIS USEFUL? 👍 👎

StackBlitz bolt.new

Feedback ⓘ 1 RELEASE · 2026-09-12

CHANGELOG ↗ RANK

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

> Bolt.new expanded its Bolt Slides feature with direct visual editing that's free until saved, plus new export, organization, and speaker-view tools for decks.

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

01 Visual editing in Bolt Slides preview NEW

73

Bolt Slides now supports direct visual editing: selecting the visual edit tool in the chatbox lets users click an element in the live preview to change its color or font size, with changes free (no token cost) until Save changes is clicked to write the updates to project code.

Tweak brand colors and font sizes in the live preview without spending tokens — changes are free until you commit.

📍 In the chatbox, select the visual edit tool, click an element in the preview to change its color or font size, then click Save changes when done to write the updates to project code.

— Names exact UI steps and the free-until-saved mechanism. changelog-20260912-050f2bb3

THINNER COVERAGE BELOW

02 Bolt Slides export, slide management, and speaker view NEW

58

Bolt Slides decks can now be exported as PDF or JSON directly from the slide editor (via Download > JSON, useful for reusing slide content in another Bolt project); slides can be reordered, duplicated, or deleted from the side panel or grid overview; and a new speaker view was added.

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

📍 In your Slides project, open the deck, then select Download › JSON to save the deck file for import into another project.

— Groups thin one-line additions to the same slides module. changelog-20260912-050f2bb3

WAS THIS USEFUL?  

Cognition Devin Desktop

Feedback ⓘ 1 RELEASE · 2026-09-12

CHANGELOG ↗ RANK

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

> Devin Desktop's September 2026 releases add SSH-based remote agent hosting, a major MCP integration overhaul (Datadog, Slack, Figma, custom servers, and a new Customize → MCPs hub), real-time voice mode, and a Data Analyst Agent, while retiring the legacy Cascade agent and Enable ACP toggle in favor of a single new agent.

└──▷ WHAT SHIPPED · 26 FEATURES most completely described first ? what's the number?

01 MCP server management moved to Customize → MCPs NEW

90

MCP server management moves to Customize → MCPs, alongside plugins, skills, hooks, and rules, at personal, organization, or enterprise scope. Adds a plugin marketplace (Browse marketplace on the Customize page) for installing MCP-backed integrations, and an Add custom MCP option for organization admins to register arbitrary MCP servers, including pre-registered OAuth applications with custom client ID and secret — extended to support custom STDIO, SSE, or HTTP servers. Adds `devin mcp login` CLI command for OAuth authentication against locally running MCP servers. The older Settings → Connections → MCP servers page now redirects to Customize → MCPs; any bookmarks or documentation pointing to the old path must be updated.

Browse and install a marketplace MCP server plugin from the Customize page to connect Devin to an external tool.

📍 In the Devin web app, go to Customize › Plugins, find the MCP server you want in the marketplace, and click Install. For a custom server, choose Add custom server and select STDIO, SSE, or HTTP, then enter the connection details.

Authenticate a locally running OAuth MCP server from the CLI without opening the browser-based settings UI.

```
$ devin mcp login
```

— Names exact config path, CLI command, and redirect notice. product docs

02 Devin Review search, PR actions, and ACU tracking NEW

88

Devin Review gains a `Cmd+F` / `Ctrl+F` find-in-page shortcut inside Devin Review and embedded Devin web surfaces; PR cards gain `Open Devin Review`, `Open in Desktop tab`, `Close PR`, and `Git settings` actions; Devin Review tabs now display the PR title, open GitHub links at their diff anchor, and swap the sessions sidebar for a PR files/review sidebar (or wiki page tree) while a fullscreen review or wiki is open. Devin Review also tracks ACU consumption per review and lets admins manage organization ACU limits.

Use find-in-page to search a large PR diff in Devin Review without leaving the app.

> 📍 Open a PR in Devin Review, then press `Cmd+F` (macOS) or `Ctrl+F` (Windows/Linux) to open the find-in-page bar and search across the diff or wiki content.

— Names exact shortcut and PR card actions; ACU limits under-specified.

`changelog-20260912-6da7d82c`

---

03 **Datadog, Slack, and Figma MCP integrations** NEW 85

Adds Datadog (API key) MCP integration using `DD-API-KEY` and `DD-APPLICATION-KEY` headers, selectable by Datadog site/region; Datadog (OAuth) MCP integration for authorizing Devin against a Datadog account through OAuth; an official Slack remote MCP server; and an official Figma remote MCP server — both installable from the plugin marketplace with OAuth authorization.

Connect Devin to Datadog using an API key so it can query metrics, logs, and monitors during sessions.

> 📍 In Customize → MCPs, open Browse marketplace, select `Datadog (API key)`, choose Connect, and supply your DD-API-KEY and DD-APPLICATION-KEY values along with your Datadog site/region.

— Names exact headers and integrations with setup steps. `product docs`

---

04 **Voice mode for real-time conversation with Devin** NEW 85

Adds voice mode, letting practitioners talk naturally with Devin to explore ideas and hand off work without typing via real-time calls. Supports in-call microphone muting and unmuting via `Mute microphone` / `Unmute microphone` controls, with Space bar as a push-to-talk shortcut while muted, and a `Silence Devin` / `Unsilence Devin` control to suppress Devin's audio output independently of the microphone. Voice call conversation persists in session history and allows in-app navigation while the call stays connected.

Start a voice session with Devin from an existing Agent mode session to talk through an approach hands-free.

📍 1. Open an existing session in Agent mode. 2. Click the voice call button beside the message box. 3. Allow microphone access when prompted — any text already typed is sent automatically. 4. Hover over the waveform icon to confirm `Start voice call` is shown before clicking.

Stay in a voice call while reviewing other parts of the Devin UI, then silence Devin's audio without dropping the call.

📍 1. While on a voice call, navigate to another session or page within Devin — the call stays connected. 2. Click `Silence Devin` to mute Devin's spoken output if you need quiet. 3. Click `Unsilence Devin` when you want to hear responses again. 4. Press and hold Space to speak while your microphone is muted, without needing to click `Unmute microphone`.

— Names exact controls and shortcut with runnable steps. `product docs`

---

05 **Legacy Cascade agent and ACP toggle removed** BREAKING

80

The `Enable ACP` toggle is removed — ACP is always enabled and can no longer be turned off. Several legacy Cascade entry points are removed: new-agent option, annotations, conversation mentions, settings group, automatic Lifeguard checks, Vibe & Replace, Deploy, and workflow commands. The previous default agent has been removed; the new agent is now the only agent available in Devin Desktop, and existing conversations must be migrated using the `Continue in Devin Local` prompt.

.

— Lists exact removed entry points and the migration path. `changelog-20260912-6da7d82c`

---

06 **`devin_list_integrations` MCP tool** NEW

80

Adds `devin_list_integrations` MCP tool to list all native integrations (e.g. GitHub, Jira, Slack) and MCP servers with their install status and settings URLs, filterable by installed, not installed, or all.

— Names exact tool and filter options, no runnable example. `product docs`

---

07 **SSH remote agent hosts in the location selector** NEW

75

Devin Desktop can now connect to remote agents over SSH from the location selector, with saved hosts, status indicators, and a server icon in the sidebar; SSH sessions recover the full transcript after reconnect.

Connect to a remote agent host over SSH so Devin runs on a beefy server while you work locally.

📍 In Devin Desktop, open the location selector, choose `Add Remote Host`, enter your SSH host details, and save. The host appears in the sidebar with a server icon and a status dot. Select it to start a session — if the connection drops, re-opening the session restores the full transcript automatically.

— Names UI path and reconnect behavior but no CLI or config surface.

changelog-20260912-6da7d82c

08 **Okta SSO via OIDC** NEW 70

Adds Okta SSO setup using OIDC, requiring an `auth.devin.ai` sign-in redirect URI configured in the Okta web app.

— Names exact redirect URI needed for configuration. product docs

09 **Devin CLI Team Settings for enterprises** NEW 65

Adds Devin CLI Team Settings for enterprises to configure model allowlists, the default model, MCP servers, and web search team-wide.

Restrict your enterprise team to approved models and set a default to prevent runaway costs.

📍 In Devin CLI Team Settings, open the model allowlist, add the models your team is permitted to use, and set the default model. Save to apply the policy across all team members.

— Names settings scope, example shows allowlist steps. product docs

10 **Network access allowlist card for blocked domains** NEW 60

Network access requests now surface an `Allowlist N domains` card that names the blocking policy and lets permitted users update the governing security profile or automation.

— Names UI card and policy link but no exact config path. changelog-20260912-6da7d82c

11 **`devin plugins` CLI commands** NEW 60

Adds `devin plugins` commands to install, author, and govern plugins in the Devin CLI across personal, organization, and enterprise scopes.

— Names command and scopes but no subcommand syntax. product docs

12 **Adaptive routing and Fusion model pairing in CLI** NEW 60

Adds model configuration in the Devin CLI, including Adaptive routing and Fusion pairings, via the new `Models` settings; Fusion pairing pairs a frontier lead model with a

cost-efficient sidekick to deliver frontier intelligence at lower cost.

— Names settings surface but no configuration steps. product docs

THINNER COVERAGE BELOW

13 **Official Devin MCP server for external AI tools** NEW 55

Adds the official Devin MCP server so external AI tools can manage Devin sessions, playbooks, knowledge, and repository docs.

— Names exposed capabilities but no endpoint or setup detail. product docs

14 **Subagents in the Devin CLI** NEW 55

Adds `subagents` to the Devin CLI to delegate tasks to independent agents running in the foreground or background with their own profiles and permissions.

— Explains mechanism but no command syntax shown. product docs

15 **Secrets & Site Cookies storage** NEW 55

Adds Secrets & Site Cookies storage for organization, personal, repo, and session secrets — including API keys, site cookies, and TOTP codes — so Devin can authenticate to external tools.

— Names scopes and secret types but no UI path. product docs

16 **Local, Worktree, and Cloud location picker** NEW 50

Adds Local, Worktree, and Cloud as top-level options in the agent location picker, with existing worktrees available in a searchable nested submenu.

— Names picker options but no deeper mechanism. changelog-20260912-6da7d82c

17 **Checkout branch discovers local clones and worktrees** IMPROVED 50

Checkout branch now discovers local clones including worktrees and prompts when multiple matches exist.

— Names command but no further mechanism detail. changelog-20260912-6da7d82c

18 **Cloud session archiving, previews, and outage resilience** IMPROVED 50

Archiving a cloud session with open pull requests now offers to close them and archives its child sessions. The cloud session view gains an image preview carousel, hook run visibility, and rich queued messages. Messages sent during a Devin Cloud outage are now held and delivered automatically when the connection returns.

– Describes behavior changes but no UI path or config given. changelog-20260912-6da7d82c

19 **Agent activation failures report stderr** IMPROVED 50

Agent activation failures now report the agent's last stderr line and open that agent's output channel.

– Names concrete diagnostic mechanism. changelog-20260912-6da7d82c

20 **Batch archive/unarchive in agent inbox** NEW 45

Adds per-row checkboxes to the agent inbox list view for batch archive/unarchive operations.

Bulk-archive stale sessions from the inbox to keep your workspace tidy.

> 📍 In the agent inbox list view, tick the per-row checkboxes next to the sessions you want to remove, then choose Archive from the batch action bar that appears at the top of the list.

– Clear UI action but limited scope detail. changelog-20260912-6da7d82c

21 **Data Analyst Agent (DANA)** NEW 45

Adds the Data Analyst Agent (DANA) for fast database queries, data analysis, and visualizations in Slack or the web app.

– Names surfaces but no setup or invocation detail. product docs

22 **Documentation localized into seven languages** NEW 40

Adds localized documentation in Spanish, Japanese, Chinese, French, German, Brazilian Portuguese, and Italian (579 pages each).

– Names languages and page count but no navigation path. product docs

23 **Inbox and sidebar navigation refinements** IMPROVED 35

The agent sidebar now shows sessions updated within the last 30 days by default, and inbox title-bar back/forward arrows now navigate between the inbox and opened sessions.

– Simple UI tweaks, one concrete default value. changelog-20260912-6da7d82c

24 **Clipboard image paste on New Session page** NEW 35

Clipboard images can now be pasted directly into the New Session page.

– Exact page named but minimal further mechanism. changelog-20260912-6da7d82c

25 **Delete local sessions from agent sidebar** NEW 35

Local sessions can now be deleted from the agent sidebar.

— Names exact sidebar location, minimal detail. changelog-20260912-6da7d82c

26 **Desktop edits counted in lines-of-code analytics** IMPROVED

20

Files Devin edits in Desktop now count toward lines-of-code analytics.

— Bare statement with no mechanism or location. changelog-20260912-6da7d82c

└──▷ **BREAKING ON UPGRADE**

! The Enable ACP toggle is removed — ACP is always enabled and can no longer be turned off.

! Several legacy Cascade entry points are removed: new-agent option, annotations, conversation mentions, settings group, automatic Lifeguard checks, Vibe & Replace, Deploy, and workflow commands.

! The previous default agent has been removed; the new agent is now the only agent available in Devin Desktop, requiring use of the Continue in Devin Local prompt to migrate existing conversations.

! The older Settings → Connections → MCP servers page now redirects to Customize → MCPs; any bookmarks or documentation pointing to the old path must be updated.

WAS THIS USEFUL?  

Red Hat ripwire

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗

RANK

ripwire gives coding agents a ranked, deterministic map of a repository — call graph, blast radius, and tests to run — from a CLI or over MCP.

> ripwire v0.6.0 adds Kotlin and Dart as fully indexed languages, ships Hermes and OpenClaw agent integrations, and introduces declined-call disclosure alongside a batch of diagnostic output fields, output-limit increases and performance gains — while raising the CPU floor for prebuilt binaries and bumping the cache format, both of which require action on upgrade.

└──▷ **WHAT SHIPPED · 19 FEATURES** most completely described first ? what's the number?

01 **Declined-call disclosure in graph output** NEW 85

`declined=` appears in the map header and `declined_calls=` appears in `--callers`, `--callees`, and `--impact` output, naming unresolved calls instead of silently dropping them so a resolver's false negatives are no longer counted as zero results.

Audit a C++ repo for callers that were previously hidden as false positives — the new `declined_calls=` field tells you exactly which calls the resolver declined rather than counting them as zero.

```
$ ripwire . --callers SafeString::move
```

— Concrete example command demonstrates the new field directly. v0.6.0

02 **x86-64-v3 CPU requirement for binaries** BREAKING 80

Prebuilt x86-64 binaries now require an x86-64-v3 CPU (Intel Haswell 2013 / AMD Excavator 2015 or newer); the 0.6.0 binary will not run on older x86-64 hardware. Set `RIPWIRE_SKIP_CPU_CHECK=1` to bypass the new CPU feature check in `scripts/install.sh` on VMs that hide CPU flags from the host.

On a VM that hides CPU flags from its host, skip the x86-64-v3 CPU check so the installer can proceed without halting on missing feature flags.

```
$ RIPWIRE_SKIP_CPU_CHECK=1 bash -c "$(curl -fsSL
https://raw.githubusercontent.com/redhat-
et/ripwire/main/scripts/install.sh)"
```

— Runnable bypass command and exact hardware generations named. v0.6.0

03 **Hermes and OpenClaw agent integrations** NEW 75

`skills/install.sh --hermes` and `skills/install.sh --openclaw` activate Hermes and OpenClaw agent integrations, and `ripwire wrap` prints the MCP setup needed to wire ripwire into either agent.

Install the Hermes agent skill and print the MCP setup so you can wire ripwire into a Hermes-based coding agent.

```
$ skills/install.sh --hermes
ripwire wrap
```

— Runnable install commands and MCP setup shown for both agents. v0.6.0

04 **Dart language indexing** NEW 75

Adds Dart (`.dart`) as the 23rd indexed grammar: definitions, call edges, and metrics from a vendored tree-sitter-dart grammar; indexes 71,726 symbols across flutter/packages.

— Concrete symbol count and source repo, but no command shown. v0.6.0

05 **Budget ceiling enforcement for --for** BREAKING 75

`--for` discloses `budget_bytes="7500"` when trimming its default answer, and adds `over_ceiling="1"` on `--for --detail` when the answer exceeds `--max-tokens` . A budgeted `--for` that cannot fit its allowance now takes the ladder's last rung and discloses the overflow, rather than shipping past the allowance silently as before, so some already-over-budget bundles come out larger.

— Full before/after behavior explained, but no runnable example. v0.6.0

06 **Kotlin language indexing** NEW 70

Adds Kotlin (`.kt`) indexing as a first-class language: classes, objects, companion objects, functions, calls, imports, and inheritance, with bidirectional Kotlin/Java call resolution across the JVM boundary.

— Names symbol kinds and cross-language resolution, no runnable example. v0.6.0

07 **Diagnostic disclosure fields across CLI output** NEW 70

`--grep` root output adds `corpus_pruned_dirs=` naming directories pruned from the corpus. Parse output adds `extent_suspect=` for files derailed by a member macro, counted in `unranked_extent_suspect=` so they leave `--hotspots` ranking without silent loss. Graph verb roots add `graph_unindexed=` to disclose unindexed symbol

references. `--version` output adds `emit=` naming the formatted-output emitter compiled into the binary, and `--help-task` adds `no_route` as a possible MCP answer.

— Five named output fields across surfaces, no example usage shown. v0.6.0

08 **Ruby dependency resolution improvements** IMPROVED 70

Improves Ruby dependency resolution to cover superclass constants, `include` / `extend` / `prepend`, `autoload`, and constant receivers such as `User.find` for Zeitwerk/Rails applications.

— Names the specific Ruby/Rails constructs now resolved. v0.6.0

09 **Performance improvements across grep, map, pack-task, and --for**

IMPROVED 65

`--grep` performance on llvm-project improves from 159.7s to 9.2s (warm); the warm default map improves from 248s to 10s; `--pack-task` on Go improves from 8.13s to 5.88s (warm); and a repeated `--for` on llvm-project improves from 274s to 26s.

— Concrete before/after benchmark numbers, no mechanism given. v0.6.0

10 **Gitignore handling in --grep and --regex** BREAKING 60

`--grep` and `--regex` no longer read gitignored files of an extension the indexer skips; `--no-ignore` restores the previous behavior.

— Names the restoring flag, giving a clear remediation step. v0.6.0

THINNER COVERAGE BELOW

11 **Increased output limits for --expand and --handoff** IMPROVED 55

`--expand` now lists up to 100 sibling names per body, up from 8. `--handoff` now shows up to 50 symbols per code file and 12 per prose file, up from 6.

— Exact before/after limits for two named flags. v0.6.0

12 **Cache and parser version bump** BREAKING 55

The cache format moves from 16 to 20 and the parser version from 81 to 92, so caches written by 0.5.0 are not reused; the first run on each tree after upgrading is a cold parse.

— Exact version numbers given, but no migration command. v0.6.0

13 **std::-qualified call resolution change** BREAKING 55

A `std::` `-qualified` C++ call no longer binds an in-repo definition outside namespace `std`; it now counts toward `external=` instead, so `external=` reads higher.

— Explains the resolution change but gives no command to check it. v0.6.0

---

14 **`--hook` flag enforcement in skill installers** BREAKING

55

`skills/install.sh --openclaw --hook` now exits 2 instead of being silently ignored, and `--hermes --hook` is refused the same way.

— Names exact flag combinations and new exit behavior. v0.6.0

---

15 **Per-line anchoring in `--regex`** IMPROVED

50

`--regex` now anchors `^` and `$` per line, preventing matches from spanning lines.

— Names the flag and exact anchoring behavior fixed. v0.6.0

---

16 **Full flag catalog via `--help=all`** IMPROVED

50

`--help=all` prints the full flag catalog, while plain `--help` now prints one line per flag instead of the full listing.

— Exact flag named with clear before/after behavior. v0.6.0

---

17 **Claude Code config directory support** NEW

45

ripwire now reads the `CLAUDE_CONFIG_DIR` environment variable and respects Claude Code's configuration directory wherever it looks for it.

— Single env var named, no further mechanism given. v0.6.0

---

18 **`--quality-delta` per-kind dialing** BREAKING

40

`--quality-delta` now dials each kind separately, and `churn="self"` no longer gates it.

— Names the flag and field but gives no further detail. v0.6.0

---

19 **Budget-ranked section serving in `--recall`** IMPROVED

35

`--recall` now spends its budget on sections in rank order and names what was served in its disclosure.

— One-sentence description with no numbers or example. v0.6.0

└──▷ BREAKING ON UPGRADE

! Prebuilt x86-64 binaries now require an x86-64-v3 CPU (Intel Haswell 2013 / AMD Excavator 2015 or newer); the 0.6.0 binary will not run on older x86-64 hardware.

! The cache format moves from 16 to 20 and parser version from 81 to 92, so caches written by 0.5.0 are not reused and the first run on each tree is a cold parse.

! `--quality-delta` now dials each kind separately, and `churn="self"` no longer gates it.

! `--grep` and `--regex` no longer read gitignored files of an extension the indexer skips; `--no-ignore` restores the previous behavior.

! A `std::` `-qualified` C++ call no longer binds an in-repo definition outside namespace `std` ; it counts toward `external=` instead, so `external=` reads higher.

! `skills/install.sh --openclaw --hook` now exits 2 instead of being silently ignored, and `--hermes --hook` is refused the same way.

! A budgeted `--for` that cannot fit its allowance now takes the ladder's last rung and discloses the overflow, where it previously shipped past the allowance silently, so some already-over-budget bundles come out larger.

WAS THIS USEFUL? 👍 👎

AGENT

◆ AI Agent Frameworks

Daily Pipecat

Feedback ⓘ 1 RELEASE · 2026-09-12 NOTES ↗ RANK

Pipecat is a framework for realtime voice and multimodal agents, orchestrating speech-to-text, model inference, and text-to-speech as a streaming pipeline.

> Pipecat v1.10.0 ships major breaking dependency upgrades (openai 3, anthropic 1, mcp 2.x), a full migration of Speechmatics STT to the Agent API, new server-side turn detection for Gradium STT, continuous TTS via Smallest AI's continuation API, LiveKit audio buffer tuning, DeepSeek reasoning/thinking fixes, and eval-suite dashboard improvements.

└──▷ WHAT SHIPPED · 10 FEATURES most completely described first ? what's the number?

01 **Gradium server-side turn detection** NEW 88

`enable_turn_detection` added to `GradiumSTTService` to delegate end-of-turn decisions to Gradium's server-side endpointing signal instead of the pipeline's VAD, tunable via `eot_horizon_s`, `eot_threshold`, and `post_flush_cooldown_frames`.

Enable Gradium's server-side turn detection so the STT service — not the pipeline VAD — decides when each user turn ends, reducing false cut-offs in noisy environments.

python

```python
stt = GradiumSTTService(
    enable_turn_detection=True,
    eot_horizon_s=0.8,
    eot_threshold=0.6,
    post_flush_cooldown_frames=3,
)
```

— Named flag and tuning params with a runnable code example. v1.10.0

02 **Speechmatics STT migrated to Agent STT** BREAKING 86

`SpeechmaticsSTTService` now targets Speechmatics Agent STT (`/v2/agent`) via the `speechmatics-agent-stt` SDK, replacing `speechmatics-voice[smart]` in the `speechmatics` extra, with a new `model` setting defaulting to `linden-1` and `enable_partials` replacing `include_partials`. Default `turn_detection_mode` is now `TurnDetectionMode.VAD` (pass `TurnDetectionMode.EXTERNAL` to drive turns from Pipecat's own VAD as before), and the service gains automatic reconnection with exponential backoff after dropped connections or recoverable server errors, with permanent-error reporting for rejected credentials or exhausted retries. Removed:

`SpeakerFocusMode`, `SpeakerFocusConfig`, `UpdateParams`, update_params(), and the Settings / `InputParams` fields `focus_speakers`, `ignore_speakers`, `focus_mode`, `speaker_passive_format`, `max_delay`, `end_of_utterance_silence_trigger`, `end_of_utterance_max_delay`, `split_sentences`, `include_results`, and `extra_params`, plus `TurnDetectionMode.FIXED`, `TurnDetectionMode.ADAPTIVE`, and `TurnDetectionMode.SMART_TURN`; `OperatingPoint` is replaced by Model.

— Extensive named surfaces and mechanism, but no runnable usage example. v1.10.0

---

03 **openai 3 SDK support** BREAKING 85

The `openai` dependency is widened to `>=1.74.0,<4` to support the openai 3 SDK, which uses `httpx2` instead of `httpx` for HTTP and verifies TLS against the OS trust store rather than `certifi`. Container images without system CA certificates and TLS-inspecting proxy environments may need `SSL_CERT_FILE` or `SSL_CERT_DIR` set; a Timeout passed to `OpenAITTSService` or Whisper-based STT services must come from `httpx2` on openai 3, or `httpx` on older versions. Pin `openai<3` to stay on the previous HTTP stack.

— Names exact dependency bounds and migration remedy but no code sample. v1.10.0

---

04 **Continuous TTS via Smallest AI continuation API** NEW 81

`max_buffer_delay_ms` added to `SmallestTTSService` to control the server-side buffering window for Smallest AI's continuation API, which joins text fragments within the same LLM turn under a shared `context_id` for unbroken prosody.

Use Smallest AI's continuation API with a 200 ms server buffer so fragments in the same LLM turn are joined into one continuous utterance instead of resetting prosody on each chunk.

python

```python
tts = SmallestTTSService(
    max_buffer_delay_ms=200,
)
```

— Named config key and mechanism with runnable code example. v1.10.0

---

05 **anthropic 1 SDK support** BREAKING 71

The `anthropic` dependency is widened to `>=0.49.0,<2` to support the anthropic 1 SDK; `AnthropicLLMService` now routes `temperature`, `top_k`, and `top_p` through `extra_body`. If a Bedrock client is passed, anthropic 1 requires an explicit region via AsyncAnthropicBedrock(aws_region=...) or the `AWS_REGION` environment variable, where it previously defaulted to `us-east-1`.

— Named config surfaces and default change, no runnable example. v1.10.0

06 **Eval suite dashboard and recording status** IMPROVED

70

The `pipecat eval suite` live dashboard now shows pass rate for in-progress repeated (bot, scenario) rows and replaces the yellow middle band with green (all passed) / red (one failed) colouring. The `recording` line in `pipecat eval suite` settings reports which runs will actually record, e.g. on (3 of 6 runs; text mode skipped) or off (all runs text mode); `pipecat eval run -a` prints the same line.

— Names the exact CLI commands and output strings, no code example. v1.10.0

07 **DeepSeek reasoning content and thinking defaults** BREAKING

67

`DeepSeekLLMAdapter` supplies `reasoning_content` on assistant messages when tool calls are in the conversation, fixing requests in thinking mode. `DeepSeekLLMService` now defaults `thinking` to disabled; pass thinking=DeepSeekLLMService.ThinkingConfig(type='enabled') to enable it or `thinking=None` to use DeepSeek's own default.

— Names the adapter, fix, and default change but no example. v1.10.0

08 **LiveKit outgoing audio buffer control** NEW

65

`LiveKitParams.audio_out_queue_size_ms` configures the outgoing `rtc.AudioSource` buffer size, defaulting to LiveKit's 1000 ms.

Increase the LiveKit outgoing audio buffer to reduce underruns on high-jitter networks without changing any other pipeline settings.

```python
params = LiveKitParams(
    audio_out_queue_size_ms=2000,
)
```

— Named config field with default and a runnable example. v1.10.0

THINNER COVERAGE BELOW

09 **MCP SDK 2.x compatibility** IMPROVED

43

The `mcp` dependency is widened to `mcp[cli]>=1.24.0,<3` so MCPClient works with MCP SDK 2.x as well as 1.x.

— Bare dependency bound with no further mechanism or example. v1.10.0

10 **LmntTTSService removed** DEPRECATED

30

`LmntTTSService` is removed because LMNT has shut down.

— One-line removal notice with no migration path given. `v1.10.0`

└──▷ BREAKING ON UPGRADE

! The `openai` dependency now resolves to openai 3 by default (`>=1.74.0,<4`), which uses `httpx2` instead of `httpx` for HTTP and verifies TLS against the OS trust store rather than `certifi` . Container images without system CA certificates and TLS-inspecting proxy environments may need `SSL_CERT_FILE` or `SSL_CERT_DIR` set. A `Timeout` passed to `OpenAITTSService` or Whisper-based STT services must come from `httpx2` on openai 3, or from `httpx` on older versions. Pin `openai<3` to stay on the previous HTTP stack.

! `SpeechmaticsSTTService` now targets Speechmatics Agent STT (`/v2/agent`) via `speechmatics-agent-stt` (replaces `speechmatics-voice[smart]` in the `speechmatics` extra) and cannot connect to the legacy real-time endpoint.

! `SpeechmaticsSTTService` default `turn_detection_mode` is now `TurnDetectionMode.VAD` ; pass `turn_detection_mode=TurnDetectionMode.EXTERNAL` to drive turns from Pipecat's own VAD as before.

! `SpeechmaticsSTTService` renames `Settings.include_partials` to `enable_partials` .

! `SpeechmaticsSTTService` removes `SpeakerFocusMode` , `SpeakerFocusConfig` , `UpdateParams` , update_params(), and the `Settings` / `InputParams` fields `focus_speakers` , `ignore_speakers` , `focus_mode` , `speaker_passive_format` , `max_delay` , `end_of_utterance_silence_trigger` , `end_of_utterance_max_delay` , `split_sentences` , `include_results` , and `extra_params` , as well as `TurnDetectionMode.FIXED` , `TurnDetectionMode.ADAPTIVE` , and `TurnDetectionMode.SMART_TURN` . `OperatingPoint` is replaced by `Model` .

! `LmntTTSService` is removed; LMNT has shut down.

! The `anthropic` dependency now resolves to anthropic 1 (`>=0.49.0,<2`). If you pass a Bedrock client, anthropic 1 requires an explicit region via AsyncAnthropicBedrock(aws_region=...) or the `AWS_REGION` environment variable, where it previously defaulted to `us-east-1` .

WAS THIS USEFUL? 👍 👎

PydanticAI

Feedback ⓘ 1 RELEASE · 2026-09-12 NOTES ↗ RANK

PydanticAI is a framework that builds type-safe Python agents with dependency injection, model integrations, tools, and structured outputs.

> PydanticAI's biggest addition is `capabilities`, a composable extension system bundling tools, hooks, instructions, and model settings into reusable units, shipped alongside dozens of built-in Core and Harness capabilities spanning provider-native search/thinking, sandboxed execution, memory, guardrails, durable execution and more; clai also picked up a first-run session banner.

└──▷ **WHAT SHIPPED · 25 FEATURES** most completely described first ? what's the number?

01 Capabilities: composable agent extension system NEW

95

Adds the `capabilities` parameter to `Agent` (and `AgentSpec`) for composing reusable units of agent behavior — tools, lifecycle hooks, instructions, model settings, and model selection — in a single import. The `Capability` class declaratively bundles instructions, function tools, and toolsets without subclassing, using decorators like `@refunds.tool_plain` to attach tools; `defer_loading=True` collapses a bundle to a catalog entry until the model loads it. `AbstractCapability` supports subclassing for behavior beyond instructions/tools/toolsets, `CapabilityEvent` (with a `namespace` argument and ctx.emit()) provides typed event publication, and the `Toolset` convenience class wraps an `AbstractToolset` as an authoring primitive.

Equip a research agent with deep thinking and web search in one `capabilities` list — useful when you want thorough, cited answers without threading model settings and tool registrations separately.

python

```python
from pydantic_ai import Agent
from pydantic_ai.capabilities import Thinking, WebSearch

agent = Agent(
    'anthropic:claude-fable-5',
    instructions='You are a research assistant. Be thorough and
cite sources.',
    capabilities=[
        Thinking(effort='high'),
        WebSearch(local='duckduckgo'),
    ],
)
```

Bundle a refund skill — instructions plus a tool — into a named `Capability` so the same unit can be reused across multiple agents without duplication.

python

```python
from pydantic_ai import Agent
from pydantic_ai.capabilities import Capability

refunds = Capability(
    id='refunds',
    description='Use for refund eligibility and refund status.',
    instructions='Always confirm the order ID before issuing a
refund.',
)

@refunds.tool_plain
def refund_status(order_id: str) -> str:
    """Look up the refund status for an order."""
    return f'Order {order_id}: refund issued on 2026-05-01.'

agent = Agent('openai:gpt-5.6-sol', capabilities=[refunds])
```

Emit a typed `CapabilityEvent` from a capability-contributed tool so other capabilities and observability pipelines can react to file writes without polling.

python

```python
from dataclasses import dataclass
from typing import Any
from pydantic_ai import CapabilityEvent, RunContext
from pydantic_ai.capabilities import AbstractCapability
from pydantic_ai.toolsets import AgentToolset, FunctionToolset

WORKSPACE = 'workspace'

@dataclass(kw_only=True)
class FileWriteEvent(CapabilityEvent, namespace=WORKSPACE):
    path: str
    bytes_written: int

workspace = FunctionToolset()

@workspace.tool
async def write_file(ctx: RunContext[Any], path: str, content:
str) -> str:
    await ctx.emit(FileWriteEvent(path=path,
bytes_written=len(content)))
    return f'Wrote {path}'
```

— Runnable code examples plus every named class and flag product docs

02 **StepPersistence for resumable and forkable runs** NEW

73

`StepPersistence` saves, restores, resumes via `continue_run`, and forks via `fork_run` runs, with file, SQLite, and Mongo backends.

— Names two runnable functions and three storage backends product docs

03 **Provider-native model capabilities: search, thinking, images** NEW

68

Adds `Thinking` for provider-adaptive extended thinking at configurable `effort`; `WebSearch` for provider-native search with a `local` fallback (e.g. `local='duckduckgo'`); `WebFetch` for fetching and reading URLs, native or local; `XSearch` for searching X with a subagent fallback on non-xAI providers; `MCP` to connect any MCP server's tools with opt-in provider-native connectors; `ImageGeneration` for generating and editing images, provider-native where supported with a direct image-model fallback; and `NativeTool` to register any provider-native tool with the agent.

— Names each capability and its parameters, no runnable example product docs

04 **Tool definition and execution controls** NEW

65

Adds `ToolSearch` for loading tool definitions on demand instead of carrying all definitions in every prompt; `ToolApproval` for flagging tool calls that require human approval before execution; `HandleDeferredToolCalls` for programmatically resolving approval-deferred tool calls; `PrepareTools` / `PrepareOutputTools` for filtering or modifying tool definitions per step; `PrefixTools` to wrap another capability and prefix its tool names; `IncludeToolReturnSchemas` to include return type schemas in tool definitions sent to the model; `SetToolMetadata` to merge metadata key-value pairs onto selected tools; and `ThreadExecutor` for running sync tools on a shared thread pool.

— Eight named capabilities, mechanism per one but no example product docs

05 **Harness Compaction with model-agnostic strategies** NEW

65

`Compaction` (Harness) provides model-agnostic compaction strategies — tool-result clearing, sliding-window trimming, LLM summarization, and tiered — all window-relative with live usage reporting.

— Names four strategies with behaviour, no example product docs

06 **Sub-agent orchestration capabilities** NEW

64

Adds `Planning` for model-owned task plans with a cache-safe live reminder; `Subagents` for delegating self-contained tasks to named child agents; `DynamicWorkflow` for model-orchestrated sub-agents with fan-out, chain, and vote patterns in a single tool call, enforced by hard `max_agent_calls` budgets; and `Advisor` for letting an executor consult a stronger model mid-run.

— Names the `max_agent_calls` budget and orchestration patterns product docs

07 **CodeMode capability for batched tool calls via sandboxed script**

NEW 64

`CodeMode` lets the model write one Python script calling many tools inside a Monty sandbox, reducing round-trips and keeping intermediate results out of the context window.

— Clear before/after mechanism, no runnable example product docs

08 **Agent memory and history search capabilities** NEW 64

Adds `Memory` providing a persistent, namespaced notebook with bounded prompt injection, on-demand search, and in-memory, file, or Postgres stores; and `ConversationSearch` for BM25 search over stored history including turns compaction dropped.

— Names storage backends and search algorithm, no example product docs

09 **Model selection and lifecycle processing capabilities** NEW 62

Adds `SelectModel` for static or per-step model selection via a callable; `ResolveModelID` for resolving custom application-specific model IDs via a callable; `Hooks` for decorator-based lifecycle hook registration; `ProcessHistory` to wrap a history processor; `ProcessEventStream` to forward agent stream events to a handler function; and `RaiseContentFilterError` to raise `ContentFilterError` whenever a model response has `finish_reason='content_filter'`.

— Six named capabilities with mechanism, no usage example product docs

10 **Observability and durable execution capabilities** NEW 60

Adds `Instrumentation` for OpenTelemetry GenAI spans covering every model and tool call; `Compaction` for provider-native context compaction on OpenAI and Anthropic; and `DurableExecution` for runs that survive restarts and failures on Temporal, DBOS, Prefect, Restate, Kitaru, and Airflow.

— Names concrete platforms and telemetry standard, no example product docs

THINNER COVERAGE BELOW

11 **Sandboxed environments and third-party integrations** NEW

58

Adds `ModalSandbox` for commands and files in an isolated Modal cloud sandbox; `StackOne` for acting on linked SaaS accounts (HRIS, ATS, CRM) via StackOne; `LocalStack` for an emulated AWS environment with AWS CLI tools; and Macroscope for running a local Macroscope code review and handing findings to the agent.

— Four named integrations, mechanism per one, no example product docs

12 **On-demand context loading capabilities** NEW

58

Adds Skills for loading `AgentSkill` (`SKILL.md`) instructions on demand; `RepoContext` for starting runs oriented with `AGENTS.md` / `CLAUDE.md` plus repository structure; and `PydanticAIDocs` for on-demand Pydantic AI documentation lookup.

— Three named capabilities citing exact file conventions product docs

13 **Guardrails capability for input and output validation** NEW

58

Guardrails validates, blocks, or redacts user input, tool calls, tool results, and output, including secret masking and parallel async guards.

— Lists scope and mechanism, no example or config key product docs

14 **FileSystem capability for sandboxed file access** NEW

57

`FileSystem` allows reading, writing, editing, and searching files under a root, with path-traversal and symlink safety and secrets read-only access.

— Names safety mechanisms but no runnable example product docs

15 **Shell capability for controlled command execution** NEW

57

Shell executes commands with allowlists, denylists, timeouts, and credential-stripping.

— Names controls but no config keys or example shown product docs

16 **SpendLimits capability for cross-window cost budgets** NEW

57

`SpendLimits` enforces cross-window USD/token budgets and per-response cost tracking, per model and per tenant.

— Names budget dimensions but no config or example product docs

17 **Web research and browsing sub-agent capabilities** NEW

54

Adds `ExaSearch` for web research via Exa with excerpted search, full-page reads, and opt-in cited deep search; `ExaAgent` for delegating open-ended research to the Exa Agent API; and `BrowserUse` for handing web tasks to an autonomous browser-use agent driving a real browser.

– Three named delegated-research capabilities, no example product docs

18 **Tool output and prompt-cache management** NEW

54

Adds `ToolOutputLimits` for truncating, spilling to a queryable file, or summarizing oversized tool returns at the source; and `WarnOnCacheBusts` for detecting prompt-cache prefix collapses between requests using provider-reported numbers.

– Two named capabilities with mechanism, no example product docs

19 **ManagedPrompt for Logfire-backed instructions** NEW

54

`ManagedPrompt` backs instructions with a Logfire-managed prompt, enabling versioning and rollout without redeployment.

– Names the mechanism and benefit, no config example product docs

20 **CapabilityCreation for agent-authored capabilities** NEW

50

`CapabilityCreation` allows the agent to write, validate, and persist new capabilities during a run, loaded on the next run.

– Clear behaviour described, no example of the flow product docs

21 **CoderHarness complete coding-agent stack** NEW

46

`CoderHarness` bundles a complete coding-agent stack including files, shell, repo context, planning, a read-only explorer sub-agent, and context controls.

– Lists components but no mechanism detail or example product docs

22 **ResearcherHarness complete web-research stack** NEW

46

`ResearcherHarness` bundles a complete web-research stack including search, page fetching, a delegated sub-researcher, and bounded tool output.

– Lists components but no mechanism detail or example product docs

23 **ACP for serving agents over Agent Client Protocol** NEW

46

`ACP` (experimental) serves any agent to editors like Zed over the Agent Client Protocol.

– Names protocol and target editor, marked experimental product docs

24 **SystemReminders capability for guidance re-injection** NEW

35

`SystemReminders` provides cache-safe re-injection of guidance mid-run to counter instruction fade.

– One-line description with no mechanism detail product docs

25 **First-run banner in clai sessions** IMPROVED

30

`clai` interactive sessions now open with a first-run banner that describes the current run.

– Bare description, no configuration or example given v2.43.0

WAS THIS USEFUL?  

Stanford NLP DSPy

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗

RANK

DSPy builds language-model pipelines and optimizes their prompts and weights.

> DSPy 3.4.0b1 introduces a pluggable native LM engine interface built on a new typed `dspy.lm15` module, a local CPython interpreter for code-executing agents, async support in ReActV2, and a GEPA code proposer, alongside a set of breaking changes that remove the experimental typed-LM API and alter ReActV2 and multi-answer behavior.

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

01 **Custom LM engine interface via dspy.lm15** NEW 95

Adds `engine=` parameter to dspy.LM(...) accepting `'auto'`, `'litellm'`, `'lm15'`, or a custom engine object implementing `complete(Request) -> Response`, plus `async_engine=` for async calls, enabling custom backends without subclassing `BaseLM`. Also adds the `dspy.lm15` module exporting `Request`, `Response`, `Message`, `Config`, and `Usage` types for constructing explicit typed LM calls, with `response.text` and `response.message.parts` for accessing results.

Plug in a custom LM backend — e.g. a local model server — without subclassing BaseLM, while letting DSPy handle caching, retries, and history.

python

```python
from dspy.lm15 import Message, Response, Usage
import dspy

class MyEngine:
    def complete(self, request):
        # call your own backend here
        text =
my_model_server.generate(request.messages[-1].text)
        return Response(
            id=None,
            model=request.model,
            message=Message.assistant(text),
            finish_reason='stop',
            usage=Usage(),
        )

lm = dspy.LM('custom/my-model', engine=MyEngine())
dspy.configure(lm=lm)
```

Send an explicit typed request to get a structured response with usage metadata, bypassing DSPy's signature machinery.

```python
import dspy
from dspy.lm15 import Config, Message, Request

lm = dspy.LM('openai/gpt-4o-mini')
response = lm(Request(
    model=lm.model,
    system='Be concise.',
    messages=(Message.user('Summarize zero-trust networking in one sentence.'),),
    config=Config(max_tokens=100),
))
print(response.text)
```

— Names the exact params, types, and shows runnable code `3.4.0b1`

02 **Removal of experimental typed LM classes** BREAKING 90

`dspy.LMRequest`, `dspy.LMResponse`, `dspy.LMMessage`, `dspy.LMConfig`, `dspy.System`, `dspy.User`, and `dspy.Assistant` are removed; importing `dspy.core.types` raises a migration error, and `forward_contract='typed_lm'` is rejected. The `dspy.clients.openai_format` module is also removed, and `experimental=True` no longer changes ordinary LM calls into typed responses. Migrate to `dspy.lm15.Request`, `dspy.lm15.Response`, `dspy.lm15.Message`, `dspy.lm15.Config`, Message.user(text), and Message.assistant(text); old pickles containing the removed 3.3 classes must be loaded and exported in their original environment before upgrading.

— Enumerates every removed surface and gives migration path `3.4.0b1`

03 **LocalInterpreter for RLM and Flex code execution** NEW 85

Adds `dspy.LocalInterpreter` as an `interpreter_factory=` option for `dspy.RLM` and Flex, running generated Python in a persistent local CPython subprocess with `execution_timeout` support, without requiring Deno.

Run a code-executing RLM agent using the host Python environment instead of Deno, suitable for trusted internal tooling.

```python
import dspy

rlm = dspy.RLM(
    'question: str -> answer: int',
    interpreter_factory=dspy.LocalInterpreter,
)
result = rlm(question='What is 17 * 42?')
print(result.answer)
```

— Named config option with runnable example, mechanism given 3.4.0b1

04 **Async agent execution in ReActV2** NEW 75

Adds await agent.acall(...) support to `dspy.ReActV2`, enabling async prediction, async tools, and async MCP tool execution with structured tool-call history preserving call IDs and tool results across turns.

— Names method and mechanism but no runnable example given 3.4.0b1

05 **ReActV2 signature and async-tool restrictions** BREAKING 75

`dspy.ReActV2` rejects signatures with output fields named `history` or `termination_reason`; rename those outputs before upgrading. Calling an async tool synchronously from a running event loop raises a `ValueError` even with `allow_tool_async_sync_conversion` enabled; use await tool.acall(...) instead.

— Names the rejected fields, flag, and required fix 3.4.0b1

06 **Custom code proposer for GEPA** NEW 70

Adds `code_proposer=` parameter to `dspy.GEPA`, complementing the existing `instruction_proposer=`, allowing custom Flex code-generation constraints and domain guidance without monkeypatching the built-in proposer.

— Names the parameter and its purpose, no usage example 3.4.0b1

07 **Native n answers use sequential requests** BREAKING 70

Native `n` answers now use separate sequential requests instead of a single provider-native multi-answer request, which can increase latency and bill input tokens more than once; use `engine='litellm'` to retain the previous behavior.

— Explains before/after behavior and gives a workaround flag 3.4.0b1

THINNER COVERAGE BELOW

08 Pydantic 2.11.0 minimum version

BREAKING

20

Pydantic 2.11.0 is now the minimum supported version.

— Bare version bump with no further detail `3.4.0b1`

└──▷ **BREAKING ON UPGRADE**

! `dspy.LMRequest`, `dspy.LMResponse`, `dspy.LMMessage`, `dspy.LMConfig`, `dspy.System`, `dspy.User`, and `dspy.Assistant` are removed; importing `dspy.core.types` raises a migration error, and `forward_contract='typed_lm'` is rejected. Migrate to `dspy.lm15.Request`, `dspy.lm15.Response`, `dspy.lm15.Message`, `dspy.lm15.Config`, Message.user(text), and Message.assistant(text).

! Old pickles containing the removed experimental 3.3 classes are not automatically migrated; load and export them in their original environment before upgrading.

! `experimental=True` no longer changes ordinary LM calls into typed responses.

! Pydantic 2.11.0 is now the minimum supported version.

! `dspy.ReActV2` rejects signatures with output fields named `history` or `termination_reason`; rename those outputs before upgrading.

! Calling an async tool synchronously from a running event loop raises a `ValueError` even with `allow_tool_async_sync_conversion` enabled; use await tool.acall(...) instead.

! The `dspy.clients.openai_format` module is removed; custom adapters must not depend on it.

! Native `n` answers now use separate sequential requests instead of a single provider-native multi-answer request, which can increase latency and bill input tokens more than once; use `engine='litellm'` to retain the previous behavior.

WAS THIS USEFUL? 👍 👎

CopilotKit OpenBot

Feedback ⓘ 1 RELEASE · 2026-09-11

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's v0.0.9 release focuses on hardening setup and secrets handling — installing container engines automatically, moving credentials off `.env` into platform-native stores, and adding a script to sweep abandoned staged attachments — alongside new multi-file message attachments and support for keyless local model endpoints.

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

01 **Attachment sweep script for staged uploads** NEW 90

Adds `bun scripts/cull-staged-attachments.ts` (run from `/app/server`) to sweep unsent staged attachments older than one day, requiring only `DATABASE_URL`; accepts an optional retention-window argument to change the cutoff.

Run a one-pass attachment sweep from an external cron to prevent users hitting the 32-file staged limit — only `DATABASE_URL` is needed.

```
$ DATABASE_URL=postgres://user:pass@host:5432/openbot bun
scripts/cull-staged-attachments.ts 86400
```

— Names exact script path, required env var, and gives a runnable command v0.0.9

02 **Multi-file message attachments** NEW 70

Messages can now include up to eight file attachments: images (PNG, JPEG, GIF, WebP) up to 8 MiB and text files (plain text, Markdown, CSV, JSON) up to 1 MiB. Files are uploaded at staging time so sends are immediate and failures preserve the attachments.

— Concrete limits and formats given but no config or command to invoke v0.0.9

03 **Keyless local model endpoint support** NEW 65

Supports connecting to keyless endpoints — Ollama, vLLM, LM Studio, llama.cpp — by requiring only a base URL and model name when no API key applies.

— Names four supported backends and the two config fields needed v0.0.9

04 **Credentials moved to platform credential store** IMPROVED 65

Credentials (model keys, plan tokens, inter-service tokens) are now stored in the platform credential store — macOS login Keychain, Windows DPAPI, or an owner-only file on Linux — rather than in `.env`.

— Names each platform mechanism and what it replaces v0.0.9

05 **Automatic container engine installation during setup** NEW

60

Setup now automatically installs a container engine (Podman or Docker) and a Compose implementation when neither is present, pinned to a tested digest and verified before execution.

— Describes pinning and verification mechanism but is automatic, not user-driven v0.0.9

THINNER COVERAGE BELOW

06 **AGENT_BOT_MODEL env var for bundled Bot** NEW 50

The bundled Bot now reads `AGENT_BOT_MODEL` so a locally-chosen model name reaches the container.

— Names the env var but gives no further mechanism v0.0.9

07 **Setup verification step and clipboard paste support** IMPROVED

45

Setup now ends with a live question-and-answer verification step to confirm the model key, plan, and Bot are functional before handover, and an Edit menu was added to the setup window so clipboard paste works on screens whose instruction is to paste a code.

— Two thin UX fixes to the same setup window, no deeper mechanism given v0.0.9

08 **Subscription-based Bot vendor re-pointing** IMPROVED

45

A subscription now re-points the Bot to the vendor that can spend it, and the model screen names the resulting Bot before the screen is dismissed.

— Explains behaviour but not the underlying routing mechanism v0.0.9

WAS THIS USEFUL? 👍 👎

Nous Research Hermes

Feedback ⓘ 1 RELEASE · 2026-09-11

NOTES ↗ CODE ↗ RANK

A terminal-native, self-improving AI agent for autonomous coding and tasks, with persistent memory, agent-created skills, and a multi-platform messaging gateway.

> Hermes shipped a major expansion across four releases: a Desktop app, a Skills System with Curator maintenance, asynchronous peer-agent runs, a curated SHA-pinned plugin catalog, and a password-blind credential vault for signing in and paying on the web, alongside dozens of new config keys, secret backends, execution targets, and CLI subcommands.

└──▷ WHAT SHIPPED · 43 FEATURES most completely described first ? what's the number?

01 **Plugin catalog and installation system** NEW 95

Adds `plugin install <identifier> [--force] [--ref COMMIT_SHA] [--allow-removed]` to install plugins from the Hermes plugin catalog, a Git URL, or shorthand, with `--ref` pinning to a full 40-character commit SHA and `--allow-removed` bypassing the removed-plugin blocklist. Ships a curated, SHA-pinned plugin catalog with CLI, admission CI, docs, and a dashboard view; Desktop now has a single Plugins page covering agent and desktop plugins, install, catalog, and per-commit pinning; and private git plugins can now install using stored vault credentials.

Search the plugin catalog for recon-related plugins and install one pinned to a specific reviewed commit.

```
$ hermes plugin search recon --json
hermes plugin install recon-toolkit --ref
a3f8c2d1e0b7654321fedcba9876543210abcdef
```

— Full flag syntax plus runnable install example v2026.9.11

02 **Asynchronous peer run commands** NEW 90

Adds `hermes peer run <peer>[/<agent>] [message]` to start a long canonical Bot Chat turn asynchronously, returning `run_id`, session ID, and idempotency key; `hermes peer status <peer>[/<agent>] <run_id>` to poll a run and print its final output when complete; and `hermes peer stop <peer>[/<agent>] <run_id>` to stop an exact run without targeting another turn.

Run a one-shot agent task asynchronously across a peer, then poll until done — useful in CI pipelines where you need the result but cannot block the initiating process.

```
$ run_id=$(hermes peer run myagent/analyst 'Summarize open
incidents' | jq -r .run_id)
hermes peer status myagent/analyst "$run_id"
```

— Full command set with runnable example product docs

03 **Config key dot-escaping in config set/get/unset** IMPROVED

85

Adds dot-escaping support to `hermes config set/get/unset`: literal dots in key names (e.g. model IDs like `grok-4.6`, Matrix room IDs like `!room:example.org`) are navigated by preferring existing literal keys; new dotted keys require backslash-escaping (e.g. `grok-4\.7`).

— Exact command and escaping syntax given product docs

04 **Context and memory files for agent personality** NEW

80

Adds `SOUL.md`, `USER.md`, `MEMORY.md`, and `AGENTS.md` context files with defined roles for controlling agent personality and persistent cross-session memory. Adds `.hermes.md`, `AGENTS.md`, `CLAUDE.md`, and `.cursorrules` project context files automatically injected into every conversation, and `@ -syntax` inline context references for attaching files, folders, git diffs, and URLs directly into messages.

— Names all context files but no usage syntax detail product docs

05 **Recurring session loops via /heartbeat and /loop** NEW

80

Adds `/heartbeat` command to re-enter an idle session on a recurring interval (e.g. `/heartbeat every 10m`), and a `/loop` equivalent via Recurring Loops to re-run a prompt on a configurable recurring interval inside a session.

Keep a long-running deployment watch loop alive in a session — Hermes re-prompts itself every 10 minutes when idle.

```
$ /heartbeat every 10m Check the deployment.
```

— Runnable command example included product docs

06 **hermes chat --oneshot and --query-file flags** NEW 80

Adds `--oneshot` flag to `hermes chat` to answer a query and exit (pre-0.21 single-query behavior), and `--query-file` to read the query from a file.

– Exact flags and command given product docs

07 **Automatic delegation joining in finite chat runs** IMPROVED 80

When `chat --oneshot` or non-TTY stdio is used, `delegate_task` now waits for children and returns results to the parent in the same turn, with `delegation.max_concurrent_children` and `approvals.single_query_mode` still enforced.

– Named config keys but no worked example product docs

08 **Fleet health check subcommands** NEW 80

Adds `runs` and `incidents` subcommands for read-only fleet health checks: failed runs, failed deliveries, overdue/missing `next_run_at`, missing scripts or workdirs; exits non-zero when issues are found.

– Named subcommands and exit behavior, no full syntax product docs

09 **docker_snap_compat for snap-packaged Docker** NEW 80

Adds `docker_snap_compat` config key to drop `--init` and `--security-opt no-new-privileges` for snap-packaged Docker on Ubuntu/Azure hosts where AppArmor confinement would otherwise kill the container.

– Names key and exact flags dropped product docs

10 **/fast slash command for Fast Mode** NEW 80

Adds `/fast normal|fast|auto|cold` slash command to switch Fast Mode for the current session, with a `--persist` flag to save the choice.

– Exact command syntax and flag given product docs

11 **Automatic config backups before risky operations** NEW 80

Adds automatic point-in-time config backups to `backups/config/config.yaml.<reason>.<timestamp>` before `hermes setup`, `hermes migrate`, or on parse failure, keeping at most the newest five per reason.

– Exact path and retention count, no restore command product docs

12 **hermes update --check command** NEW 80

Adds `hermes update --check` CLI command and Check for Updates... / Settings → About → 'Check now' UI paths to bypass the 24-hour passive update-check cache.

Force Hermes to check for updates immediately without waiting for the 24-hour passive cache window.

```
$ hermes update --check
```

— Runnable command and UI path both given product docs

13 **Credential priority and refresh auth subcommands** NEW

75

Adds `hermes auth priority` subcommand to move a credential to the front of `fill_first` order, and `hermes auth refresh` to refresh one OAuth credential and clear its cooldown.

— Exact subcommands named with mechanism product docs

14 **Backup import safeguards** IMPROVED 75

Adds a warning when importing an older backup over newer work, reporting session and message counts when the imported `state.db` holds fewer messages than the one it replaced. Extends backup import to write SQLite databases (e.g. `response_store.db`) in-place into the existing file rather than renaming, preventing open connections from diverging; skipped databases are listed under Warnings (N files skipped).

— Names files and behavior, no restore command shown product docs

15 **agent.budget_warning_ratio checkpoint config** NEW 75

Adds `agent.budget_warning_ratio` config key to append a model-visible checkpoint notice after a configurable fraction of the iteration budget is consumed, with Kanban dispatcher workers defaulting to a 90% threshold.

— Named key and default, no file path shown product docs

16 **terminal.env_passthrough for remote SSH sessions** NEW

75

Adds `terminal.env_passthrough` config key to forward skill-declared variables to remote SSH sessions via OpenSSH `SendEnv`; provider credentials are never forwarded even if listed.

— Named key and mechanism, no example values product docs

17 **Guided onboarding and free Nous tier** NEW 75

Adds `HERMES_GUEST_ONBOARDING=1` environment variable to enable a guided first-launch experience for new users; adds `/login` command to sign in to a Nous account directly from a chat session; and adds free Nous inference and connectors out of the box, activated with one sign-in command.

— Named env var and command, activation path clear v2026.9.11

18 **Multi-provider model configuration via config.yaml** NEW

70

Adds `config.yaml` configuration for providers, models, and API keys — compatible with Nous Portal, OpenRouter, OpenAI, Anthropic, Google, and any OpenAI-compatible endpoint. Also supports local model execution with no account or API key required.

— Names config file and providers but no key schema shown product docs

19 **One-command import from Claude Code or Codex CLI** NEW

70

Supports one-command import of Claude Code (`~/.claude`) or OpenAI Codex CLI (`~/.codex`) setups — including instructions, allowlists, MCP servers, skills, and memories.

— Names source paths but not the import command product docs

20 **GIT_SSH_COMMAND override for git operations** NEW 70

Adds `GIT_SSH_COMMAND` environment variable support for internal Git operations, taking precedence over Hermes's default `ssh -o BatchMode=yes` for custom identity or transport commands.

— Named variable and default override, no example command product docs

21 **Per-profile environment variable isolation** NEW 70

Introduces per-profile environment variable isolation under a multiplexed multi-profile gateway: `${VAR}` references in a profile resolve only against that profile's own secret scope, not the shared process environment.

— Mechanism explained, no config key named product docs

22 **Password-blind credential vault for web tasks** NEW

70

Enables the agent to sign in, pay, and fill addresses using credentials from 1Password, Bitwarden, or the local Hermes vault without exposing secrets to the model; two-factor codes are supplied from a saved authenticator key or prompted in the user's UI.

— Mechanism detailed but no setup command shown v2026.9.11

23 **LSP-powered semantic diagnostics on file writes** NEW

65

Supports LSP-powered semantic diagnostics via `write_file` and `patch` post-write lint checks using real language servers (`pyright`, `gopls`, `rust-analyzer`, and others).

— Names tools and functions but no setup steps product docs

24 **Nix setup support** NEW 65

Adds Nix setup support including `nix run` and a fully declarative NixOS module with container mode.

— Names nix run and module but no full config example product docs

25 **context_file_read_timeout for slow context sources** NEW 65

Adds `context_file_read_timeout` config key (default unspecified seconds) to skip slow network-backed context files (iCloud Drive, OneDrive, NFS) rather than stalling the system prompt load.

— Named key but default value unspecified product docs

26 **Verification evidence ledger** NEW 65

Stores verification evidence in `~/.hermes/verification_evidence.db`; the ledger is only written while `verify_on_stop` is enabled.

— Named file path and flag, no query example product docs

27 **Hermes Desktop app** NEW 60

Adds Hermes Desktop app with streaming tool output, side-by-side previews, file browser, voice, cron, profiles, skills, and settings — available on macOS, Windows, and Linux.

— Feature list given but no navigation path shown product docs

THINNER COVERAGE BELOW

28 **External memory provider integrations** NEW 55

Supports external memory provider plugins: Honcho, OpenViking, Mem0, Hindsight, Holographic, RetainDB, ByteRover, and Supermemory.

— Lists providers but no setup instructions product docs

29 **Google Workspace integration via OAuth2** NEW 55

Supports Google Workspace integration (Gmail, Calendar, Drive, Sheets, Docs) via OAuth2-authenticated Google APIs.

— Names apps and auth method, no setup path product docs

30 **Skills System and Curator maintenance** NEW 55

Adds Skills System with on-demand knowledge documents, progressive disclosure, agent-managed skills, and a Skills Hub. Adds Curator for background maintenance of

agent-created skills — usage tracking, staleness detection, archival, and LLM-driven review.

— Describes mechanism but no invocation commands product docs

31 **Extensible web dashboard** NEW 55

Adds extensible web dashboard supporting themes, custom tabs, shell slots, page-scoped slots, and backend API routes.

— Names extension points but no API details product docs

32 **tool_search for discovering connector tools** NEW 55

Adds `tool_search` to make connector tools (Gmail, Linear, Notion, and others) discoverable without knowing their exact names.

— Names surface but no usage syntax shown v2026.9.11

33 **Subscription Proxy for exposing Nous Portal as OpenAI endpoint**

NEW 50

Adds Subscription Proxy to expose a Nous Portal subscription (or other OAuth provider) as an OpenAI-compatible endpoint for external apps.

— No setup command or endpoint path given product docs

34 **Deliverable Mode for native file attachments** NEW 50

Adds Deliverable Mode to ship generated charts, PDFs, spreadsheets, and other files as native attachments in messaging platforms.

— Describes output but no activation steps product docs

35 **Additional secret backends for credential storage** NEW

50

Supports Bitwarden Secrets Manager, 1Password, and a Command Helper Secret Source as secret backends.

— Names backends but no config keys shown product docs

36 **Multi-backend execution support** NEW 50

Supports execution on local, Docker, SSH, Daytona, Modal, and Singularity backends.

— Lists backends only, no setup flags product docs

37 **Managed Scope for admin-pinned configuration** NEW 45

Adds Managed Scope: administrator-pinned, user-immutable config and secrets via a system-level managed directory.

— No directory path or command given product docs

38 **Egress credential-injection proxy (iron-proxy)** NEW 45

Adds Egress credential-injection proxy (`iron-proxy`) for secure outbound credential handling.

— Names the proxy but no usage steps product docs

39 **Git Worktrees for parallel agent runs** NEW 45

Supports Git Worktrees to run multiple Hermes agents safely on the same repository using isolated checkouts.

— No command shown for enabling worktrees product docs

40 **Cross-platform installation targets** NEW 45

Supports native Windows 10/11, macOS, Linux, WSL2, and Android via Termux installation targets.

— Names platforms but no install commands product docs

41 **Mixture of Agents (MoA) presets** NEW 35

Adds Mixture of Agents (MoA) presets that appear as selectable models under the Mixture of Agents provider.

— Thin description with no config detail product docs

42 **Profile Distributions for sharing agent configs** NEW 30

Adds Profile Distributions to package and share a complete agent configuration.

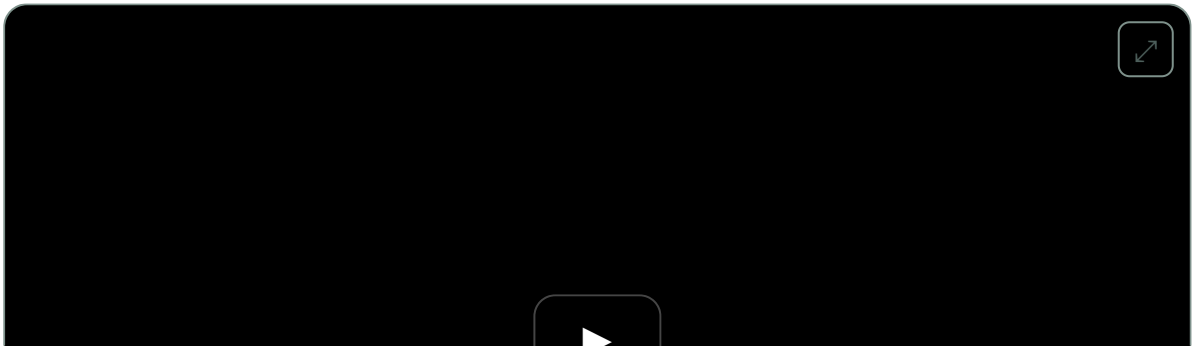
— Very thin, no format or command named product docs

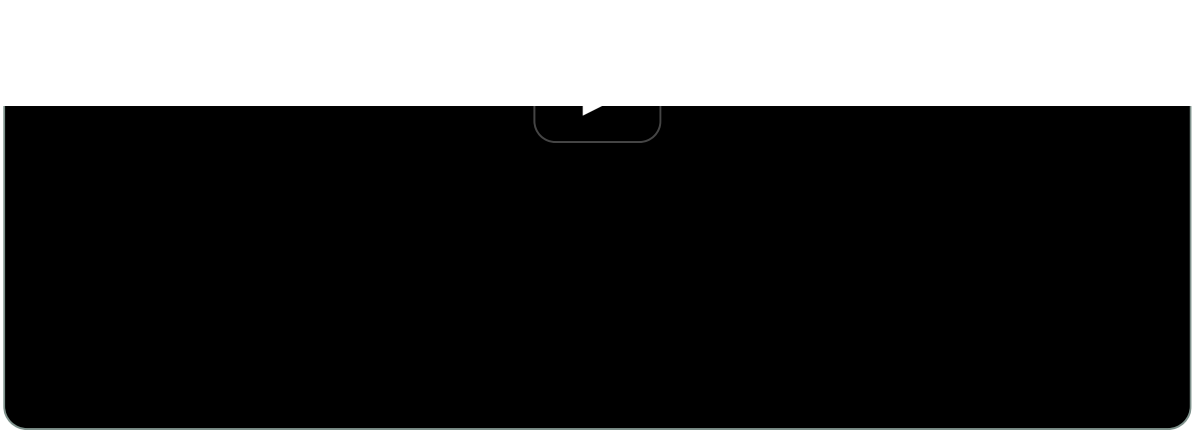
43 **Radio SDK plugin** NEW 25

Radio ships as an opt-in SDK plugin.

— Extremely thin description, no mechanism v2026.9.11

└──▷ ALSO FROM THESE RELEASES





Work around a vLLM tool-call parser that leaks markup into plain text, causing delegated tasks to silently no-op.

```yaml
model.streaming: false
```

WAS THIS USEFUL? 👍 👎

Vercel AI SDK

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗ RANK

The Vercel AI SDK provides TypeScript APIs for model generation, structured output, tool use, and streaming application interfaces.

> Vercel AI SDK 7.0.98 extends embedding and reranking with runtime context attribution, adds batch image generation, and introduces a stable UI message type for tool output errors.

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

01 **Runtime context attribution for embed and rerank** NEW

45

Adds runtime context attribution support to the `embed`, `embedMany`, and `rerank` functions, allowing context to be tracked through these calls.

— Names three functions but no schema or usage shown ai@7.0.98

02 **Batch image generation requests** NEW

23

Image generation now supports submitting requests in batches.

— No batch size, API, or usage details given ai@7.0.98

03 **Stable UI message type for tool errors** NEW

16

Introduces a stable UI message type for representing tool output errors.

— Only mentioned in summary, no structure or fields given ai@7.0.98

WAS THIS USEFUL? 👍 👎

LangChain

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗ RANK

LangChain is an open-source framework that orchestrates applications powered by language models.

> langchain-core 1.6.3 adds the ability for gateway responses to override model name and provider fields in tracing metadata, improving trace attribution when requests pass through a proxy.

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

01 **Gateway override of tracing model metadata** IMPROVED

45

When a proxy or gateway sits in front of the underlying model, langchain-core can now use the gateway response's model name and provider values to override tracing metadata, giving accurate attribution in traces.

– Explains mechanism and purpose but no config key or flag named `langchain-core==1.6.3`

WAS THIS USEFUL? 👍 👎

Microsoft Agent Framework

Feedback ⓘ

1 RELEASE · 2026-09-11 NOTES ↗ RANK

The Microsoft Agent Framework builds, orchestrates, and deploys agents and multi-agent workflows in Python and .NET, consolidating AutoGen and Semantic Kernel.

> Agent Framework .NET 1.21.0 ships several breaking changes—moving the line-numbering contract onto `AgentFileStore`, clarifying A2A run modes, and restricting MCP skill archives to ZIP—alongside hosted-response improvements, sandbox hardening for LocalCodeAct and Hyperlight, and dependency upgrades for `Azure.AI.Projects` and AWS Bedrock.

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

01 **Line-numbering moved onto AgentFileStore** BREAKING 70

Adds `file_access_read_lines` to `AgentFileStore`, moving the line-numbering contract onto the store itself; callers that previously handled line numbering outside the store must migrate to the new contract.

— Names the exact method and breaking migration but no code sample dotnet-1.21.0

THINNER COVERAGE BELOW

02 **Dependency upgrades for Azure and AWS integrations** IMPROVED 55

Upgrades the `Azure.AI.Projects` dependency to `3.0.0-beta.2`, and switches the AWS Bedrock integration from the deprecated `AWSSDK.Extensions.Bedrock.MEAI` package to `AWS.Bedrock.MEAI`.

— Names exact package versions, a clear upgrade path dotnet-1.21.0

03 **A2A task-state tracking and run-mode clarification** BREAKING 50

Tracks and updates A2A task state across agent runs, and clarifies A2A agent run modes via a breaking API change — existing A2A agent configurations must be updated to match the new run-mode contract.

— Migration is flagged but no concrete config keys shown dotnet-1.21.0

04 **Sandbox hardening for LocalCodeAct and Hyperlight** IMPROVED 50

Isolates the `LocalCodeAct` subprocess environment for sandboxed code execution, hardens `LocalCodeAct` OS validation, and implements the same approval process for

`LocalCodeAct` as used by Hyperlight; also canonicalizes Hyperlight sandbox fingerprints.

— Names both sandbox components but mechanism details are brief `dotnet-1.21.0`

05 **Hosted agent response handling improvements** IMPROVED

45

Opts in to client function forwarding for Responses hosting, enabling hosted agents to invoke client-side functions; scopes OpenAI hosting storage by isolation key; and preserves streamed annotations in Foundry hosted responses.

— Describes three hosting behaviours but no config or API names `dotnet-1.21.0`

06 **MCP skill archives restricted to ZIP** BREAKING

40

Limits MCP skill archives to the ZIP format only; any non-ZIP archive that was previously accepted will now be rejected.

— Clear before/after but no migration steps given `dotnet-1.21.0`

└──▷ **BREAKING ON UPGRADE**

! `file_access_read_lines` is added and the line-numbering contract is moved onto `AgentFileStore` — callers that previously handled line numbering outside the store must migrate.

! A2A agent run modes have been clarified with a breaking API change — existing A2A agent configurations must be updated to match the new run-mode contract.

! `LocalCodeAct` subprocess environment is now isolated — code that relied on inheriting the parent process environment will no longer receive those variables.

! MCP skill archives are now restricted to ZIP format only — any non-ZIP archive previously accepted will be rejected.

WAS THIS USEFUL? 👍 👎

assistant-ui

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗ RANK

Assistant-ui provides React components and state management for chat interfaces with streaming, tool calls, and attachments.

> assistant-ui shipped an Assistant Cloud 0.2 telemetry stack — server-side trace export merged with client run reports, batched engagement events, custom scores and feedback APIs, presigned file downloads, structured API errors, and anonymous-thread claiming on sign-in.

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

01 Trace export and run reports for Assistant Cloud NEW

95

Adds `createAssistantCloudTraceExporter`, `createAssistantCloudSpanProcessor`, `assistantCloudTraceMetadata`, and `withAssistantCloudTraceMetadata` in `assistant-cloud/telemetry` to send AI SDK GenAI spans to `POST /v1/traces` and merge server spans with client run reports into a single trace. Also adds a `createRunReport` builder in `assistant-cloud` to assemble run-report bodies carrying `provider`, `outcome_type` (`aborted`, `disconnected`, `length`, `content_filter`), `error_code`, `error`, `message_id`, `first_token_ms`, `duration_ms`, `finish_reason`, `trace_id`, `environment`, `release`, and `tags` from the `telemetry` config.

Ship server-side OpenTelemetry traces to Assistant Cloud so every LLM call is linked to its client-side run report.

javascript

```
import { createAssistantCloudTraceExporter,
createAssistantCloudSpanProcessor } from 'assistant-
cloud/telemetry';

const exporter = createAssistantCloudTraceExporter({ cloud });
const processor = createAssistantCloudSpanProcessor(exporter);
// register processor with your OpenTelemetry SDK provider
```

— Names exact functions, endpoint, and run-report fields with a runnable example.

`@assistant-ui/react@0.15.19`

02 Engagement-event batching to Assistant Cloud NEW

75

Adds engagement-event batching to `POST /v1/events` covering sends, edits, stops, regenerates, copies, branch switches, suggestions, attachments, thread switches, speech, voice, and shown errors; opt out with `telemetry.events: false`.

Opt out of engagement-event collection (sends, copies, branch switches, etc.) while keeping run-report telemetry active.

```javascript
const runtime = useCloudChat({
  cloud,
  telemetry: {
    events: false
  }
});
```

— Names endpoint, event types, and exact opt-out config. `@assistant-ui/react@0.15.19`

THINNER COVERAGE BELOW

03 **Custom scores and message feedback APIs** NEW 55

Adds `cloud.scores.create` for submitting custom scores and `useCloudChat().feedback` for message feedback alongside the existing `FeedbackAdapter`.

— Names the two new APIs but gives no further mechanism. `@assistant-ui/react@0.15.19`

04 **Presigned download URLs for uploaded files** NEW 55

Adds `cloud.files.generatePresignedDownloadUrl` and exposes the object `key` on file upload responses.

— Names method and response field, minimal added detail. `@assistant-ui/react@0.15.19`

05 **Structured error codes on CloudAPIError** NEW 50

Adds `CloudAPIError.code` and `CloudAPIError.details`, including a `plan_limit_reached` code on HTTP 402 responses.

— Names fields and one error code, no usage example. `@assistant-ui/react@0.15.19`

06 **Anonymous-thread claiming on sign-in** NEW 40

Exposes the browser's anonymous refresh token and migrates its threads into an authenticated Assistant Cloud workspace after sign-in.

— Describes behavior but names no API or command. `@assistant-ui/react@0.15.19`

WAS THIS USEFUL? 👍 👎

◆ MCP Servers & Tooling

Composio

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗ RANK

Composio connects AI agents to external tools and services with managed authentication and integrations.

> Composio's TypeScript SDK (@composio/core@1.0.0-beta.1) picked up a wide set of new capabilities — webhook lifecycle management, log search, connected-account revocation and deferred-auth completion, OAuth scope recommendations, custom toolkits, pluggable logging, and more — alongside two breaking behavior changes to webhook validation and connected-account refresh, while the platform API quietly deprecated an old tool-scopes endpoint.

└──▷ WHAT SHIPPED · 15 FEATURES most completely described first ? what's the number?

01 **Configurable SDK logging** NEW 90

Adds `logger` and `logLevel` options to new Composio({...}): `logger` accepts any `{ error, warn, info, debug }` sink (console, pino, winston, etc.) and `logLevel` accepts `'silent' | 'error' | 'warn' | 'info' | 'debug'`, overriding the `COMPOSIO_LOG_LEVEL` environment variable.

Route all Composio SDK logs through pino at debug level to capture per-request lifecycle events in production.

javascript

```
import pino from 'pino';
import { Composio } from '@composio/core';

const logger = pino();
const composio = new Composio({
  apiKey: process.env.COMPOSIO_API_KEY,
  logger,
  logLevel: 'debug'
});
```

— Runnable code shows exact config and override behavior `@composio/core@1.0.0-beta.1`

02 **Full webhook lifecycle management via SDK** NEW 80

Adds `composio.webhooks.subscriptions` with `list`, `get`, `set`, `update`, `delete`, `rotateSecret`, and `listEventTypes` methods, and `composio.webhooks.endpoints` with `list`, `get`, `create`, `replace`, and `update` methods for full webhook lifecycle management.

— Every method named but no runnable example given `@composio/core@1.0.0-beta.1`

03 **OAuth scope recommendations for toolkits** NEW 80

Adds composio.toolkits.recommendScopes(toolkitSlug, { tools, ... }) and composio.toolkits.listGrantContexts(toolkitSlug) for OAuth scope recommendations (API-beta), letting callers request only the scopes a given tool set needs.

Get the OAuth scopes recommended for a set of tools before creating a connected account, so you request only what is needed.

javascript

```
const scopes = await composio.toolkits.recommendScopes('github',
{
  tools: ['GITHUB_CREATE_ISSUE', 'GITHUB_LIST_REPOS']
});
console.log(scopes);
```

— Runnable example shows exact call and output `@composio/core@1.0.0-beta.1`

04 **Connected account revocation with typed errors** NEW 70

Adds composio.connectedAccounts.revoke(), surfacing the API's 400/409 errors as `ComposioConnectedAccountRevocationNotSupportedError` and `ComposioConnectedAccountNotRevokableError`.

— Names method and error types but no code example `@composio/core@1.0.0-beta.1`

05 **Deferred OAuth connection completion** NEW 70

Adds composio.connectedAccounts.completeAuth({ userId, sessionUri }) to complete a deferred OAuth connection after callback verification.

— Signature named but flow not demonstrated `@composio/core@1.0.0-beta.1`

06 **Exposed download size cap for URL uploads** NEW 70

Exports `readResponseBodyWithLimit` and `MAX_URL_UPLOAD_SIZE_BYTES` so downstream packages can apply the SDK's 100 MiB cap when downloading from user-supplied URLs.

— Names exports and the 100 MiB limit, no example call `@composio/core@1.0.0-beta.1`

07 **Tool-execution log search API** NEW 60

Adds composio.logs.search() and composio.logs.get() for querying tool-execution logs.

— Methods named, no usage example or query shape given `@composio/core@1.0.0-beta.1`

08 **connectedAccounts.refresh() drops validateCredentials option** 60

BREAKING

connectedAccounts.refresh() now ignores the `validateCredentials` option and logs a warning when it is set, because the upstream API removed `validate_credentials` from the connected-account refresh endpoint.

– Explains cause and new warning behavior, no migration steps `@composio/core@1.0.0-beta.1`

THINNER COVERAGE BELOW

09 **Custom toolkits pilot** NEW 55

Adds `composio.experimental.customToolkits` with `upsert`, `sync`, and `delete` for project-owned custom toolkits (pilot, shape may change).

– Named methods, pilot status flagged, no example `@composio/core@1.0.0-beta.1`

10 **Keyring transfer key listing** NEW 55

Adds composio.keyring.listTransferKeys(), returning the public JWKs of the organization's customer-managed keyring.

– Method and return shape named, no usage shown `@composio/core@1.0.0-beta.1`

11 **Toolkits bulk retrieval and changelog access** NEW 50

Adds composio.toolkits.getMany(slugs) and composio.toolkits.changelog() for bulk toolkit retrieval and changelog access.

– Two thin one-line additions to the toolkits namespace `@composio/core@1.0.0-beta.1`

12 **Stricter validation on trigger webhook subscriptions** BREAKING 50

composio.triggers.setWebhookSubscription() now rejects malformed API responses with a `ValidationError` instead of reading them leniently.

– Behavior change stated, no migration guidance `@composio/core@1.0.0-beta.1`

13 **Experimental usage reporting API** NEW 45

Adds composio.experimental.usage.summary() and composio.experimental.usage.breakdown() for usage reporting (experimental, shape may change).

– Named but explicitly unstable, no example `@composio/core@1.0.0-beta.1`

14 **New CIMD_OAUTH auth scheme** NEW 35

Adds `CIMD_OAUTH` to `AuthSchemeTypes`.

– Single enum value added, no context given `@composio/core@1.0.0-beta.1`

15 **Session config history** NEW 30

Adds session.configHistory() on sessions.

— Bare method name, no further detail `@composio/core@1.0.0-beta.1`

└──▷ **BREAKING ON UPGRADE**

! Deprecated endpoint POST `/api` /v3.1/tools/scopes/required (still works)

! composio.triggers.setWebhookSubscription() now rejects malformed API responses with a `ValidationError` instead of reading them leniently.

! connectedAccounts.refresh() now ignores the `validateCredentials` option and logs a warning when it is set, because the upstream API removed `validate_credentials` from the connected-account refresh endpoint.

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ Agentic Infrastructure

HeyGen HyperFrames

Feedback ⓘ 1 RELEASE · 2026-09-11

NOTES ↗ RANK

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

> HyperFrames v0.8.35 tunes its audio-ducking defaults so narration stays clear over background music without abrupt dips at sentence breaks.

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

01 Stronger default voiceover carve in carve.mjs

IMPROVED

75

The default carve strength in `carve.mjs` was raised from 0.25 to 0.8, and the level-envelope release was extended from 1.6s to 2.4s, keeping narration clear over background music without the bed dropping out at every sentence break.

— Names exact file, old and new values, but no usage example `v0.8.35`

WAS THIS USEFUL? 👍 👎

NVIDIA Transformer Engine

Feedback ⓘ

1 RELEASE · 2026-09-11 NOTES ↗ RANK

Transformer Engine provides optimized kernels and low-precision training support for Transformer models on NVIDIA GPUs.

> Transformer Engine v2.19 adds Rubin GPU support, experimental per-direction hybrid quantization, opt-in 2D MXFP8 block scaling, an experimental GatedDeltaNetAttention kernel, and expanded THD context parallelism, alongside expert-parallelism, attention and build-system improvements and two breaking API changes.

└──▷ WHAT SHIPPED · 20 FEATURES most completely described first ? what's the number?

01 **2D block scaling for MXFP8 weights** NEW 85

Adds opt-in 2D block scaling for MXFP8 weights via MXFP8BlockScaling(enable_2d_quantization=True), giving finer-grained quantization granularity during training.

Enable 2D block scaling for MXFP8 weights on Blackwell GPUs to get finer-grained quantization granularity during training.

```python
import transformer_engine.pytorch as te
from transformer_engine.common.recipe import MXFP8BlockScaling

recipe = MXFP8BlockScaling(enable_2d_quantization=True)

model = te.Linear(1024, 4096)
with te.autocast(enabled=True, recipe=recipe):
    out = model(inp)
```

— Runnable example and named config surface v2.19

02 **NCCL-EP Python API signature change** BREAKING 85

The NCCL-EP Python API changed to support eager dispatch and drop-on-overflow: pass `num_topk` to `ep_bootstrap`, update positional `ep_bootstrap` and `EpBuffer` calls, and replace `EpBuffer.token_counts` with `EpBuffer.tokens_per_expert`.

— Names exact params and fields to migrate v2.19

03 **parallel_cross_entropy memory and performance** IMPROVED 75

Adds an `overwrite_input` option to `parallel_cross_entropy` to reduce memory usage by reusing input storage in place, and further optimizes the function's performance and memory use.

Reduce `parallel_cross_entropy` peak memory by reusing the input tensor storage in-place, useful when input activations are not needed after the loss.

```python
import transformer_engine.pytorch as te

loss = te.parallel_cross_entropy(logits, labels,
overwrite_input=True)
```

— Named flag with a runnable code example v2.19

04 **MXFP8 quantization in NCCL-EP dispatch and combine** NEW

65

Adds MXFP8 quantization support to NCCL-EP dispatch forward and combine backward, with `GroupedLinear` and fused grouped MLP support for prequantized grouped inputs.

— Names layers and paths, no example v2.19

05 **cuDNN FP8 fused attention for packed THD inputs** NEW

65

Adds cuDNN FP8 fused attention support for packed THD inputs and context-parallel execution; FP8 THD sink-softmax backward requires cuDNN 9.26 or later.

— Names version requirement and scope v2.19

06 **NVFP4 precision extensions** NEW

65

Enables row-scaled NVFP4 weight-gradient computation in dense `Linear` and `GroupedLinear` layers, and adds NVFP4 stochastic-rounding and RHT split-quantization support on SM120 and SM121 GPUs.

— Names layers, GPU targets, and techniques v2.19

07 **Expanded THD context parallelism** IMPROVED

65

Expands THD context parallelism with hierarchical `a2a+p2p`, experimental no-load-balance AllGather, and FlashAttention 4 support on the `p2p`, `all_gather`, and `a2a` paths.

— Names concrete communication paths and mechanisms v2.19

08 **Build and packaging improvements** IMPROVED

65

Adds CMake discovery of NCCL headers installed by `nvidia-nccl-cu12` and `nvidia-nccl-cu13`, enables NVRTC to locate CUDA headers in Python `site-packages` at

runtime for improved JIT compilation in relocated and build-isolated installations, and reduces library, wheel, and container sizes by compressing embedded CUDA binaries.

— Names package sources and build mechanisms v2.19

09 **mhc_fused_projection H output now FP32** BREAKING 65

`mhc_fused_projection` now always returns its `H` output as FP32 instead of matching the input dtype; cast explicitly where BF16 is required.

— Clear before/after with migration guidance v2.19

10 **Fused MLA Q up-projection with RoPE and MXFP8** NEW 60

Adds forward and backward support to `FusedMLAQUpProjRopeQuant` for fused MLA Q up-projection, RoPE, and MXFP8 quantization.

— Names the fused op, no example v2.19

11 **ep_bootstrap support for orthogonal mesh axes** NEW 60

Adds `ep_bootstrap` support for meshes with axes orthogonal to expert parallelism by deriving communicator domains from the active mesh.

— Mechanism named, no runnable example v2.19

THINNER COVERAGE BELOW

12 **Per-direction hybrid quantization via CustomRecipe** NEW 55

Adds experimental per-direction hybrid quantization through `CustomRecipe`, allowing rowwise and columnwise paths to use different quantizers.

— Names the API but no usage example v2.19

13 **capture_time_hooks for graphed callables** NEW 55

Adds per-callable `capture_time_hooks` to `make_graphed_callables` for non-capturable warmup and graph-construction work.

— Names the API, no example given v2.19

14 **Tensor-parallel Newton-Schulz orthogonalization** NEW 55

Adds `newton_schulz_tp` for tensor-parallel Newton-Schulz orthogonalization of replicated or sharded matrices.

— Named function, no usage shown v2.19

15 **Named extra-output channels in te.ops.Sequential** NEW

55

Adds named extra-output channels to `te.ops.Sequential`, allowing later fusible operations to consume outputs from earlier operations.

– Mechanism described, no example v2.19

16 **torch.compile support for attention backends** IMPROVED

55

Enables torch.compile(fullgraph=True) for supported FlashAttention and the non-FP8 unfused attention backend.

– Named flag and scope, no example v2.19

17 **GatedDeltaNetAttention kernel** NEW

45

Adds experimental cuDNN-backed `GatedDeltaNetAttention` for dense and THD inputs.

– Named but described in one line v2.19

18 **Per-head maximum-logit outputs in attention** NEW

45

Adds per-head maximum-logit outputs to `fused_attn` and Flax `DotProductAttention`.

– Named APIs, minimal description v2.19

19 **Rubin SM107a GPU support** NEW

45

Adds Rubin SM107a support when building with CUDA 13.4 or later.

– Named GPU and CUDA version, thin description v2.19

20 **JAX Expert Parallelism APIs** NEW

45

Adds JAX Expert Parallelism APIs with reduced receive capacity and overflow detection.

– Describes behavior but no named endpoints v2.19

└──▷ **BREAKING ON UPGRADE**

! `mhc_fused_projection` now always returns its `H` output as FP32 instead of matching the input dtype; cast explicitly where BF16 is required.

! The NCCL-EP Python API changed to support eager dispatch and drop-on-overflow: pass `num_topk` to `ep_bootstrap`, update positional `ep_bootstrap` and `EpBuffer` calls, and replace `EpBuffer.token_counts` with `EpBuffer.tokens_per_expert`.

WAS THIS USEFUL?  

Baseten

Feedback ⓘ 2 RELEASES · 2026-09-11 NOTES ↗ RANK

Baseten deploys and serves machine learning models as autoscaling inference endpoints, with a model API catalog and dedicated GPU deployments.

> Baseten added a Management API endpoint for tracking Model API spend and brought credential-free OIDC/AssumeRole authentication to training jobs.

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

01 **Model API billing endpoint** NEW 83

New `GET /v1/billing/model_apis` endpoint on the Management API returns exact per-day spend subtotals for Model API usage, callable via API key auth for cost-allocation or budget-alerting pipelines.

Pull a daily breakdown of Model API costs programmatically to feed into cost-allocation or budget-alerting pipelines.

```
$ curl -X GET 'https://api.baseten.co/v1/billing/model_apis' \
  -H 'Authorization: Api-Key <your_api_key>'
```

— Named endpoint with runnable curl example Model API costs

THINNER COVERAGE BELOW

02 **OIDC and AWS AssumeRole for training jobs** NEW 45

Training jobs now support OIDC and AWS AssumeRole authentication to pull private container images and download model weights or training data, without storing long-lived cloud credentials in Baseten.

— Describes mechanism but no config or command shown

OIDC and AWS AssumeRole for training jo…

WAS THIS USEFUL? 👍 👎

Together AI

Feedback ⓘ 1 RELEASE · 2026-09-11 BLOG ↗ RANK

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

> Together AI substantially expanded its Fine-Tuning service with live metrics, expert-layer LoRA, early stopping, sample weighting, pre-flight dataset validation, and support for many new models, while removing two RL training-session API endpoints.

└──▷ **WHAT SHIPPED · 10 FEATURES** most completely described first ? what's the number?

01 **Expert-layer LoRA for MoE fine-tuning** NEW 86

Adds `lora_trainable_modules` config to include expert layers in LoRA fine-tuning jobs on Mixture-of-Experts models, enabling up to 89% recall on newly learned facts vs. 15% with attention-only adapters.

Fine-tune a Mixture-of-Experts model with expert-layer LoRA to maximize retention of newly learned knowledge rather than just adapting attention.

```json
{ "lora_trainable_modules": ["attention", "experts"] }
```

— Names config key and gives concrete recall numbers plus a runnable config example.

launch-20260911-5550c113

02 **Gradient accumulation for large effective batch sizes** NEW 81

Adds `gradient_accumulation_steps` config to support arbitrary effective batch sizes (effective batch = `batch_size` x `gradient_accumulation_steps`) regardless of single-pass GPU memory limits.

Train with a large effective batch size on a long-sequence model that cannot fit the full batch in a single GPU pass.

```json
{ "batch_size": 4, "gradient_accumulation_steps": 16 }
```

— Exact config keys and formula with a runnable example config. launch-20260911-5550c113

03 **Per-example sample weighting in training data** NEW 75

Adds a `weight` field to JSONL training examples for per-example sample weighting, controlling each data point's contribution to the loss across all supported formats and training methods.

Add per-example sample weights in a JSONL training file to up-weight high-quality human-annotated examples over synthetic ones.

```json
{"messages": [{"role": "user", "content": "Explain zero-day exploits."}, {"role": "assistant", "content": "A zero-day exploit..."}], "weight": 3.0}
{"messages": [{"role": "user", "content": "What is phishing?"}, {"role": "assistant", "content": "Phishing is..."}], "weight": 1.0}
```

— Named field with runnable JSONL example, scope across formats stated.

launch-20260911-5550c113

04 **Early stopping for fine-tuning jobs** NEW 70

Adds early stopping for fine-tuning jobs: enable it with a validation set and the run halts when validation loss plateaus, retaining the best checkpoint and refunding unused training steps; patience, sensitivity, and warmup are tunable.

— Explains mechanism and tunables in depth but omits exact parameter names.

launch-20260911-5550c113

05 **Dataset inspection before training** NEW 68

Adds dataset inspection capability: preview tokenized rows (token IDs, token strings, per-token labels, loss spans, truncation status) before a job starts, and retrieve the fully prepared training dataset via the Python SDK or the Download link in the console.

— Names retrieval paths (SDK, console link) and inspected fields.

launch-20260911-5550c113

06 **Pre-flight server-side dataset validation** NEW 63

Adds pre-flight server-side dataset validation that runs immediately on upload and writes results to the file object, covering message structure, tool definitions, tool-call/response pairing, reasoning fields, multimodal content, preference-pair structure, and sample weight types.

— Explains what is validated and when, but no example or command.

launch-20260911-5550c113

07 **Live per-step training metrics via API, CLI, SDK, UI** NEW 62

Exposes live per-step training metrics (loss, gradient norm, learning rate) and evaluation metrics through the Together API, CLI, Python SDK, and UI dashboard Metrics tab, with filtering by range and downsampling.

– Names surfaces and metric types but no exact command or endpoint given.

launch-20260911-5550c113

THINNER COVERAGE BELOW

08 **Sequence packing control in pre-tokenized format** NEW 59

Adds `position_ids` support in the pre-tokenized data format to control the degree of sequence packing, and a flag to disable packing entirely.

– Names the field but not the disable flag itself or a usage example.

launch-20260911-5550c113

09 **Fine-tuning support for new model families** NEW 58

Adds fine-tuning support for GLM 5.3, GLM-5.2, GLM-5.1, DeepSeek-V4-Flash-0731, DeepSeek-V4-Flash, Kimi K2.7-Code, Kimi K2.6, Qwen 3.8-27B, Qwen 3.6-35B-A3B, Qwen 3.6-27B, Qwen 3.5-27B, Qwen 3.5-9B, Qwen 3.5-4B, Qwen 3.5-2B, Qwen 3.5-0.8B, Gemma 4-31B, and Gemma 4-26B-A4B.

– Lists exact model names but no usage detail beyond availability.

launch-20260911-5550c113

10 **Multi-run comparison view in dashboard** NEW 50

Adds a multi-run comparison view in the dashboard Metrics tab, allowing several jobs to be charted together on one set of axes while runs are still in flight.

– Describes behaviour and UI location but no config or command. launch-20260911-5550c113

└──▷ **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

WAS THIS USEFUL? 👍 👎

OpenRouter

Feedback ⓘ 1 RELEASE · 2026-09-08 NOTES ↗ RANK

OpenRouter is a routing gateway that exposes hundreds of models from many providers behind one OpenAI-compatible API with failover and unified billing.

> OpenRouter introduced a hosted Linux shell tool and a companion Files API, letting any model on the platform execute commands and exchange files inside an isolated container.

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

01 **Hosted shell tool for models** NEW 75

The new `shell` server tool gives any model on OpenRouter a hosted Linux shell running in an isolated container, returning `stdout`, `stderr`, and exit codes back to the model. It is available in beta on the Responses and Messages APIs.

— Names mechanism and APIs but marked beta, no full example snapshot-20260912

THINNER COVERAGE BELOW

02 **Files API for shell container** NEW 50

A new Files API moves files into and out of the shell container, enabling download of files written during a run.

— Describes function but no endpoint names or limits given snapshot-20260912

WAS THIS USEFUL? 👍 👎

Fireworks AI

Feedback ⓘ 1 RELEASE · 2026-09-12 CHANGELOG ↗

RANK

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

> Fireworks AI shipped firectl subcommands for managing deployment annotation tags, enforced a new `custom/` namespace for customer-managed annotations (with a breaking change to API responses), and added a pre-run training cost estimator.

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

01 **Custom/ annotation namespace enforcement** BREAKING 90

Customer-managed annotation keys sent to the REST API must now begin with `custom/` (e.g. `custom/environment`); bare keys such as `environment` now return HTTP `PERMISSION_DENIED`. In addition, `GetDeployment` and `ListDeployments` responses for regular account users now include only `custom/*` annotation entries — keys outside that namespace are silently omitted rather than raising an error.

— Names exact endpoints, prefix, and error code with migration implications

changelog-20260912-cc2c9a07

02 **Deployment tag management in firectl** NEW 80

firectl v1.8.3 adds `deployment tag set`, `deployment tag unset`, and `deployment tag list` subcommands, including atomic batch operations, for managing customer-owned deployment annotations.

Tag a deployment with an environment label and verify it using the new firectl subcommands.

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

— Exact subcommands and runnable example given, batch mechanism named

changelog-20260912-cc2c9a07

THINNER COVERAGE BELOW

03 **Training cost estimator** NEW 45

Adds a training cost estimator that provides pre-run cost estimates using per-token rates for Managed/Serverless deployments and GPU-hour rates for Dedicated

deployments.

— Explains mechanism but gives no command or UI path changelog-20260912-cc2c9a07

└──▷ **BREAKING ON UPGRADE**

! Customer-managed annotation keys sent to the REST API without the `custom/` prefix (e.g. bare `environment`) now return HTTP `PERMISSION_DENIED`; callers must set `custom/environment` and update reads to use that canonical key.

! `GetDeployment` and `ListDeployments` responses for regular account users now omit annotation keys outside the `custom/*` namespace without an error, so any code reading non-prefixed keys will silently receive no data for those keys.

WAS THIS USEFUL? 👍 👎

Perplexity API

Feedback ⓘ 1 RELEASE · 2026-09-12 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 OAuth-based sign-in for its remote MCP server and made the google/gemini-3.8-flash model available with promotional per-token pricing.

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

01 **google/gemini-3.8-flash model availability** NEW 62

The model `google/gemini-3.8-flash` is now available via the API, callable through the `/chat/completions` endpoint, and offers promotional per-token pricing for cost-sensitive use.

Call google/gemini-3.8-flash via the Perplexity API to leverage its promotional per-token pricing in a cost-sensitive pipeline.

```
$ 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": "Summarize the latest threat intelligence
report."}]}'
```

— Named model with runnable curl example, but pricing detail thin

changelog-20260912-0270d2d7

THINNER COVERAGE BELOW

02 **OAuth sign-in for remote MCP server** NEW 58

The remote Perplexity MCP server now supports OAuth, letting MCP clients such as claude.ai, Claude Code, Cursor, or VS Code connect without pasting an API key; the billing organization is selected at sign-in.

— Explains mechanism and named clients but no config steps changelog-20260912-0270d2d7

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

LM Studio

Feedback ⓘ 1 RELEASE · 2026-09-12 CHANGELOG ↗ RANK

LM Studio is a desktop application that runs local LLMs and provides model discovery, downloads, chat, and a local API server.

> LM Studio 0.4.24 added advanced llama.cpp argument overrides for GGUF model loading and improved drafter compatibility heuristics for its assistant models.

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

01 **Llama.cpp argument overrides for GGUF loading** NEW

40

LM Studio 0.4.24 adds advanced llama.cpp argument overrides for GGUF model loading via the Engine Protocol.

– Names the mechanism and protocol but no flags or examples given

changelog-20260912-898ba2a3

02 **DSpark/DFlash drafter compatibility heuristics** IMPROVED

23

Improves DSpark and DFlash assistant drafter compatibility heuristics.

– Bare mention of an improvement with no detail on scope changelog-20260912-898ba2a3

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

Pinecone

Feedback ⓘ 1 RELEASE · SEEN 2026-09-12 NOTES ↗ RANK

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

> Pinecone shipped a local Docker emulator for offline development, launched Nexus — a new knowledge-curation and retrieval product with a Bring-Your-Own-Cloud mode and console-based context design — and introduced a schema-based Documents API alongside a full OAuth service-account management API.

└──▷ WHAT SHIPPED · 7 FEATURES most completely described first ? what's the number?

01 **Pinecone Local Docker emulator** NEW 95

Pinecone Local is an in-memory, Docker-based emulator for Pinecone Database that lets developers build and test locally without a Pinecone account, API-key authentication, or usage fees, supporting up to 100,000 records per index and requiring Python SDK v6.x+, Node.js SDK v5.x+, Java SDK v4.x+, or Go SDK v3.x+. It ships two modes: an index emulator image `ghcr.io/pinecone-io/pinecone-index:latest` configured via `PORT`, `INDEX_TYPE`, `VECTOR_TYPE`, `DIMENSION`, and `METRIC` env vars to spin up pre-configured dense or sparse serverless/pod indexes, and a full database emulator image `ghcr.io/pinecone-io/pinecone-local:latest` configured via `PORT` and `PINECONE_HOST` env vars that also handles index lifecycle management.

Stand up a dense and a sparse index locally via Docker Compose for offline development without touching your Pinecone account.

```yaml
services:
  dense-index:
    image: ghcr.io/pinecone-io/pinecone-index:latest
    container_name: dense-index
    environment:
      PORT: 5081
      INDEX_TYPE: serverless
      VECTOR_TYPE: dense
      DIMENSION: 1536
      METRIC: cosine
    ports:
      - "5081:5081"
    platform: linux/amd64
  sparse-index:
    image: ghcr.io/pinecone-io/pinecone-index:latest
    container_navigation: sparse-index
    environment:
      PORT: 5082
      INDEX_TYPE: serverless
      VECTOR_TYPE: sparse
      DIMENSION: 0
      METRIC: dotproduct
    ports:
      - "5082:5082"
    platform: linux/amd64
```

Run the full database emulator to test index creation and lifecycle management locally before deploying to production.

```
$ docker run -d \
  --name pinecone-local \
  -e PORT=5080 \
  -e PINECONE_HOST=localhost \
  -p 5080-5090:5080-5090 \
  --platform linux/amd64 \
  ghcr.io/pinecone-io/pinecone-local:latest
```

— Full mechanism, image names, env vars, and runnable commands given. product docs

02 **Service account management API** NEW 80

Adds a full set of service account management endpoints: list service accounts, create a service account with optional role bindings (client secret returned once), get service account details, update a service account's name, delete a service account (revokes

minted tokens within seconds), and rotate the OAuth client secret via `POST /service-accounts/{service_account_id}/rotate-secret` (previous secret and tokens revoked within seconds).

Rotate a compromised service account OAuth secret; the previous secret and all tokens it minted are revoked within seconds.

```
$ curl -X POST 'https://api.pinecone.io/service-
accounts/{service_account_id}/rotate-secret' \
  -H 'Api-Key: YOUR_API_KEY'
```

– Exact rotate-secret endpoint and curl example provided. snapshot-20260912

03 **Schema-based Documents API** BREAKING 70

Introduces the Documents API (API version `2026-07`) with schema-based indexes, replacing the previous unstructured record model. Adds a document patch endpoint to update individual fields on a document by ID, including record metadata, and a document retrieval endpoint to fetch documents by ID or metadata filter. Existing code using the previous record model must be migrated per the Adopt the Documents API guide.

– Endpoints and migration note named but no exact HTTP paths given. snapshot-20260912

04 **Context design tooling in Nexus console** NEW 65

The Nexus console adds context design tooling that lets practitioners define custom artifact and edge types in a manifest and trigger curation directly: navigate to Nexus > Contexts, select or create a context, open Context design, choose Design your own manifest, define custom artifact and edge types, then save and trigger curation to process sources.

Define custom artifact and edge types for a Nexus context manifest via the Pinecone console.

📍 1. In the Pinecone console, navigate to Nexus › Contexts. 2. Select or create a context. 3. Open Context design and choose Design your own manifest. 4. Define custom artifact types and edge types. 5. Save and trigger curation to process your sources.

– Exact console navigation steps given, though no API surface. product docs

05 **2026-07 API version OpenAPI specs** NEW 60

Adds a `2026-07` API version with new OpenAPI specs for the inference (`inference_2026-07.oas`) and admin (`admin_2026-07.oas`) surfaces.

Fetch the new 2026-07 inference API spec to explore available endpoints before integrating.

```
$ curl -O https://raw.githubusercontent.com/pinecone-io/pinecone-api/refs/heads/main/2026-07/inference_2026-07.oas.yaml
```

— Names exact spec files and fetch command but no endpoint behavior. product docs

THINNER COVERAGE BELOW

06 **Nexus Bring Your Own Cloud mode** NEW 50

Nexus BYOC (Bring Your Own Cloud) mode adds defined data-residency boundaries, authentication, encryption, and cluster-footprint limits for running Nexus in a customer-controlled cloud.

— Names mechanism elements but gives no setup or config steps. product docs

07 **Pinecone Nexus knowledge curation and retrieval** NEW 45

Nexus is a new product that curates sources into chunks and artifacts via a manifest, then answers queries with grounded, cited responses through a retrieval SDK.

— States purpose but no endpoint or command for retrieval itself. product docs

└──▷ **BREAKING ON UPGRADE**

! API version `2026-07` introduces schema-based indexes via the Documents API; existing code using the previous record model must be migrated — review the Adopt the Documents API guide for affected endpoints and required changes.

WAS THIS USEFUL? 👍 👎

okf-memory OKF Agent Memory

Feedback ⓘ

1 RELEASE · 2026-09-12 NOTES ↗ RANK

Git-native persistent memory for AI coding agents. 2 with sub-300µs in-memory BM25 search, embedded MCP server, and progressive disclosure.

> OKF Agent Memory v0.2.0 adds scoped pre-edit discovery for AI coding agents via a new `--for-path` search flag, a 3-tier epistemic governance model, and `code_refs` bindings that tie architecture decisions to source globs, backed by new path-normalization, traversal-prevention, and workspace-confinement safeguards.

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

01 **Scoped pre-edit discovery via --for-path** NEW 93

Adds `--for-path <file>` to `okf search` to retrieve only the governance rules (`hold` , `constraint` , `context`) governing a specific source file before it is edited, with results prioritized by authority (`hold` > `constraint` > `context`) and sub-300 µs latency. The same capability is exposed to agent platforms via the native MCP tool as okf_search({ "for_path": "<file>" }) over the MCP `stdio` protocol, for use with Claude Code and Cursor.

Query all active constraints and holds governing an auth file before making changes — stop a coding agent from touching frozen code without human signoff.

```
$ okf search --for-path services/auth/jwt/token.go knowledge
```

— Exact CLI flag, MCP call shape, latency, and priority order given. v0.2.0

02 **Path traversal and workspace confinement safeguards** NEW 88

`ValidateBundle` now rejects `code_refs` values containing `..` path traversal or absolute system paths (e.g. `/etc/...` , `C:\...`) as fatal `GateFindings` . The MCP server's `resolveBundleDir` confines the `bundle` argument to the server's canonical workspace root, preventing escape. `okf validate --drift` verifies that `code_refs` point to files that actually exist within the project root, while strictly confining all file checks inside that root to prevent side-channel probing.

Validate that all `code_refs` in the knowledge bundle point to files that actually exist, catching stale bindings after a codebase refactor.

```
$ okf validate knowledge --strict --drift
```

— Names exact validation mechanisms, a runnable flag, and failure modes. v0.2.0

03 **3-tier epistemic governance model** NEW 83

Introduces governance via the `governance:` frontmatter field, supporting `constraint` (mandatory guardrail), `hold` (execution freeze — no edits without human signoff), and `context` (advisory background). EffectiveGovernance() inference defaults files under `knowledge/convention/` to `constraint` and domain/architecture files to `context` when no explicit `governance:` value is set.

Bind an architecture decision to the exact source globs it governs, so `--for-path` lookups surface it automatically for any file under those paths.

yaml

```yaml
---
type: Decision
title: Ed25519 Stateless JWT Tokens
governance: constraint
code_refs:
  - services/auth/jwt/**
  - pkg/auth/**
---
```

— Field values and default inference rules named precisely. v0.2.0

04 **code_refs binding of decisions to source globs** NEW 70

Adds a `code_refs` frontmatter attribute to OKF knowledge documents, letting teams bind architecture decisions directly to source file globs (e.g. `services/auth/jwt/**`) so agents can deterministically discover applicable constraints before touching code.

Bind an architecture decision to the exact source globs it governs, so `--for-path` lookups surface it automatically for any file under those paths.

```yaml
---
type: Decision
title: Ed25519 Stateless JWT Tokens
governance: constraint
code_refs:
  - services/auth/jwt/**
  - pkg/auth/**
---
```

— Names the frontmatter field and gives a glob example. v0.2.0

05 **Asset drift detection for embedded skill bundle** NEW

68

Adds a `make sync-assets` command and a `TestDogfoodingAssetDrift` CI gate to detect drift between the live `.agents/skills/okf-memory/` skill directory and the embedded bootstrap assets in `pkg/okf/assets/skill/`.

— Names exact paths and CI gate, no usage example given. v0.2.0

06 **Path normalization in matchCodeRef engine** IMPROVED

65

Adds path normalization in the `matchCodeRef` engine, resolving relative paths, leading dot-slashes, leading slashes, Windows backslashes, and full absolute IDE system paths via suffix and subpath resolution so `code_refs` matches work across differing path formats.

— Detailed mechanism but no direct user command. v0.2.0

WAS THIS USEFUL? 👍 👎

Weaviate

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗ RANK

Weaviate is an open-source vector database that stores both objects and vectors, allowing for the combination of vector search with structured filtering with the fault tolerance and scalability of a cloud-native database.

> Weaviate's latest releases removed the `GET /mcp` endpoint, changed response schemas for two search endpoints, and added node-level query admission control alongside a more flexible HFresh debug reassignment API.

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

01 **Optional shard for HFresh debug reassignment** IMPROVED

60

Makes shard optional for HFresh reassignment in the `debug` API, allowing cluster-wide orphaned-postings recovery without specifying a shard.

— Names debug API and behaviour change but no exact path v1.39.4

THINNER COVERAGE BELOW

02 **Node-level query admission control** NEW

40

Adds node-level query admission control (P1b) to shed load and protect individual nodes from query overload.

— Describes purpose but gives no config key or flag v1.39.4

└──▷ **BREAKING ON UPGRADE**

! Removed endpoint GET `/mcp`

! POST `/search`/{collection}`/hybrid` : response schema changed

! POST `/search`/{collection}`/near-text` : response schema changed

WAS THIS USEFUL? 👍 👎

timgordontg engrim

Feedback ⓘ 1 RELEASE · 2026-09-11

NOTES ↗ RANK

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

> engrim v1.4.2 adds native Codex CLI hook integration, letting sessions auto-load and save project memory without MCP.

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

01 **Codex CLI hook integration** NEW 88

`engrim setup --codex` wires `SessionStart`, `SessionEnd`, `Stop`, and `UserPromptSubmit` command hooks into `~/.codex/hooks.json`, with deduplication and zero-dependency failure handling; `engrim uninstall --codex` removes it.

Wire engrim memory into Codex CLI so every session automatically loads and saves project context without requiring MCP.

```
$ engrim setup --codex
```

— Names exact hooks, config file path, and setup/uninstall commands. v1.4.2

WAS THIS USEFUL? 👍 👎

ralforion OrionBelt Ontology Builder

Feedback ⓘ

1 RELEASE · 2026-09-11 NOTES ↗ RANK

Browser-based ontology workbench for OWL ontologies and SKOS vocabularies. Streamlit + rdflib, no Java, no Protégé.

> OrionBelt tunes graph visualization performance with a new edge-curve rendering control and fixes overly persistent Find and Centre behavior.

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

01 Edge Curves display option for graph rendering NEW

75

A new Edge curves setting in the Visualization controls offers three modes — Auto, Curved, and Straight — to control how edges render at any graph size. Auto curves graphs up to 300 edges and straightens above that threshold, counting edges rather than nodes; under 6x CPU throttle this saves a measured 4.7ms per redraw at 400 edges and 7.4ms at 800 edges when switching from curved to straight.

On a large ontology with hundreds of edges, force straight edges to eliminate per-frame curve overhead and keep the canvas responsive.

📍 In the Visualization page, open display options, find Edge curves, and select Straight.

– Names exact settings, threshold, and measured timing gains v1.27.4

THINNER COVERAGE BELOW

02 Find and Centre no longer permanently seeds focus IMPROVED

37

Picking an entity via Find and Centre is now used as a render seed for one redraw only, instead of persisting as a permanent focus seed; clearing the picker forgets the prior pick.

– Describes before/after behavior but no config surface or steps v1.27.4

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

PromptLayer

Feedback ⓘ 1 RELEASE · 2026-09-12 CHANGELOG ↗ RANK

PromptLayer is a platform for managing, evaluating, and observing prompts and LLM applications.

> PromptLayer added support for Google's Gemini 3.8 Flash model across multiple providers with expanded context and multimodal input handling.

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

01 **Gemini 3.8 Flash model support** NEW 60

PromptLayer now supports Google's Gemini 3.8 Flash model across Google AI Studio, Vertex AI, and OpenRouter providers, with configurable reasoning effort levels (low, medium, high) and 1M+ token context length. The model also supports multimodal inputs — text, images, video, audio, and PDF files.

— Names providers, effort levels, context length, and input types.

changelog-20260912-c7f9afb7

WAS THIS USEFUL? 👍 👎

LangChain LangSmith

Feedback ⓘ 1 RELEASE · 2026-08-20

NOTES ↗ RANK

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

> LangSmith's latest release adds CLI-driven LLM-as-judge evaluator creation, bulk and thread-aware annotation queue APIs, multi-turn evaluator testing, and a new public experiment comparison API that replaces legacy comparison helpers, alongside smaller permission, trace-limit, and dataset-split UI improvements.

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

01 Bulk annotation queue item APIs for runs and threads IMPROVED

95

`POST /annotation-queues/<id>/items` adds RUN items to a single annotation queue in bulk (up to 200 items per request), resolving runs via ClickHouse or SmithDB and returning a standards-shaped items envelope. The endpoint now also accepts `item_type THREAD` (with `thread_id` + `session_id`) so conversation threads can be added to annotation queues, and mixed RUN and THREAD batches are supported in a single request.

Add a batch of conversation threads to an annotation queue for human review, mixing thread and run items in one request.

```
$ curl -X POST 'https://api.smith.langchain.com/annotation-queues/<queue_id>/items' \
  -H 'Content-Type: application/json' \
  -H 'x-api-key: <LANGSMITH_API_KEY>' \
  -d '{"items": [{"item_type": "THREAD", "thread_id": "<thread_id>", "session_id": "<session_id>"}, {"item_type": "RUN", "run_id": "<run_id>"}]}'
```

— Full endpoint, limits, and mixed-batch behavior with runnable example snapshot-20260912

02 Multi-turn evaluator testing before saving NEW

86

The `/runs/rules/validate` endpoint gains `test_thread_id` and `session_id` parameters to test a multi-turn (thread) evaluator against a real conversation before saving, and the UI adds a matching Multi-turn evaluator Test action that runs the evaluator against a sample thread before saving the rule.

Validate a multi-turn evaluator rule against a real conversation thread before committing it, to catch mapping or logic issues early.

```
$ curl -X POST
'https://api.smith.langchain.com/runs/rules/validate' \
  -H 'Content-Type: application/json' \
  -H 'x-api-key: <LANGSMITH_API_KEY>' \
  -d '{"rule_id": "<rule_id>", "test_thread_id": "<thread_id>",
"session_id": "<session_id>"}'
```

— Names endpoint params and matching UI action, with example `snapshot-20260912`

---

03 **Public experiment comparison API replaces legacy helpers**

78

BREAKING

`POST /v2/datasets/<id>/experiment-runs` is now the public API for paginated experiment comparison. Legacy dataset comparison helpers are removed from the public OpenAPI spec and generated SDKs — existing HTTP routes still work for LangSmith UI clients, but SDK-generated clients using those helpers must migrate.

— Endpoint and removal named but no migration example given `snapshot-20260912`

---

04 **CLI evaluator creation for LLM-as-judge rules** NEW

70

The `langsmith evaluator create-llm` CLI command defines structured LLM-as-judge evaluator rules from a prompt, schema, and model config file, and targets a specific project or dataset.

— Names CLI command and inputs but no full runnable example `snapshot-20260912`

---

05 **Legacy feedback formula endpoints deprecated** DEPRECATED

70

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

— Endpoints and removal date given, no migration walkthrough `snapshot-20260912`

---

06 **In-place update for code evaluator uploads** IMPROVED

65

The LangSmith CLI `evaluator upload --replace` flag now updates existing code evaluator rules in place, avoiding a delete-before-create window if the replacement upload fails.

— Flag and behavior named, no example command `snapshot-20260912`

---

07 Split assignment column and inline editing in experiment views

NEW

60

Adds a reorderable `Splits (latest)` column to the experiment comparison view, showing each example's current dataset split assignments as chips reflecting live membership. Each split chip in a row's Splits cell in experiment results and comparison views is now interactive, with an Edit splits action that opens the single-example split picker.

— UI path described but no config or API name given `snapshot-20260912`

THINNER COVERAGE BELOW

08 Separate download datasets permission NEW

50

Introduces a separate `download datasets` permission, enforced in both the application and APIs, with a disabled download button shown for users lacking the permission.

— Permission named but no config or API surface given `snapshot-20260912`

└──▷ **BREAKING ON UPGRADE**

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

! Legacy dataset comparison helpers are removed from the public OpenAPI spec and generated SDKs; update any SDK-generated clients that used those helpers (existing HTTP routes continue to work for LangSmith UI clients).

WAS THIS USEFUL? 👍 👎

Arize Phoenix

Feedback ⓘ 3 RELEASES · 2026-09-11 → 2026-09-12

NOTES ↗ RANK

Arize Phoenix is an open-source platform that monitors LLM application traces and evaluates their outputs.

> Phoenix this window added three new evaluators (retrieval relevance, completeness, session-level PII detection), expanded its MCP integrations with a `/mcp` mount and Claude Code, Codex, and Cursor plugins, and rounded out the release with GraphQL note mutations, span-level cost filtering in the DSL, and prompt version metadata exposed across REST and GraphQL.

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

01 **MCP server integrations and plugins** NEW 73

Adds an MCP mount at `/mcp`, a Claude Code plugin with marketplace support for connecting Claude Code directly to a Phoenix MCP server, an installable Phoenix Codex plugin, and a Cursor plugin for the remote MCP server that includes public skills and the Phoenix docs MCP.

— Names the `/mcp` endpoint and each plugin but no install steps given.

arize-phoenix-v20.10.0 arize-phoenix-v20.11.0

02 **New evaluators for retrieval, completeness, PII** NEW 70

Phoenix Evals adds a retrieval relevance evaluator for scoring how well retrieved documents match a query, a `completeness` evaluator for measuring whether a response fully addresses the input, and a session-level `pii_detection` evaluator that operates across an entire conversation session rather than individual spans.

— Names evaluator behaviour and two exact evaluator identifiers, but no usage example.

arize-phoenix-evals-v3.8.0 arize-phoenix-v20.10.0

THINNER COVERAGE BELOW

03 **Span-level cost filtering in DSL** NEW 55

Adds span-level cost filtering to the DSL, enabling programmatic filtering of traces by token cost.

— States mechanism and scope but no filter syntax example. arize-phoenix-v20.10.0

04 **phoenix-error-analysis skill** NEW 50

Adds a `phoenix-error-analysis` skill to the Phoenix skills library for AI-assisted error analysis.

— Names the exact skill but not how it's invoked. arize-phoenix-v20.11.0

05 **Prompt version metadata in REST API and GraphQL** NEW 50

Exposes prompt version metadata in both the REST API and GraphQL.

— Names both surfaces but not the specific fields returned. arize-phoenix-v20.10.0

06 **Cache read/write token breakdowns in UI** IMPROVED 45

Shows cache read/write breakdowns in cumulative token count tooltips in the UI.

— Names the UI element but no exact navigation path. arize-phoenix-v20.10.0

07 **GraphQL mutations for span, trace, session notes** NEW 40

Adds GraphQL mutations for attaching notes to spans, traces, and sessions.

— Names the surface (GraphQL mutations) but no mutation name or schema.

arize-phoenix-v20.11.0

08 **Annotation filtering by identifier** NEW 30

Supports filtering annotations by identifier in the filter layer.

— Bare description with no filter syntax or endpoint named. arize-phoenix-v20.10.0

WAS THIS USEFUL?  

Braintrust

Feedback ⓘ 1 RELEASE · 2026-09-01 NOTES ↗ RANK

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

> Braintrust's biggest window items are Patterns (automated trace investigations) and Loop automations running in a new managed runtime, alongside a major expansion of bt CLI tracing across seven AI coding agents and new environment/prompt-versioning, project, and template management commands — plus two breaking changes to bt sync pull and the OpenCode/pi trace integrations.

└──▷ **WHAT SHIPPED · 17 FEATURES** most completely described first ? what's the number?

01 Coding agent tracing integrations and trace CLI management

NEW 90

Adds tracing integrations for Claude Code, Claude Cowork, Codex, Google Antigravity, Grok, OpenCode, and pi, each connecting sessions to Braintrust via the MCP server or built-in tools. The `bt` CLI gained `bt trace enable antigravity` and `bt trace import antigravity` (v0.19.1+), `bt trace update` (v0.19.3+) to update installed integrations without rewriting tracing settings, `bt trace disable` to remove tracing even if plugin uninstall fails, and `bt trace doctor` to warn about outdated plugins and mismatched package versions. The `--tag` flag and `BRAINTRUST_TAGS` env var (v0.19.3+) attach session tags for Claude Code, Codex, Google Antigravity, OpenCode, and pi traces; Grok coding sessions are traced via `bt` CLI (v0.19.3+) capturing Git repository metadata without interrupting the agent on failure. Span metadata for Google Antigravity, Grok, OpenCode, and pi sessions now includes the local username.

— full command surface named across many agents snapshot-20260912

02 Environment management and prompt version pinning in bt CLI

NEW 90

Adds `bt environments create <name> --slug` to create an organization-level environment, `bt prompts versions <prompt>` to list all versions of a prompt, `bt prompts assign <prompt> --version <version> --environment <env>` to pin a version to an environment, and `bt prompts unassign <prompt> --environment <env>` to remove a prompt from an environment.

Create a Production environment and pin a tested prompt version to it as part of a deployment workflow.

```
$ bt environments create Production --slug production
bt prompts versions my-prompt
bt prompts assign my-prompt --version <version> --environment
production
```

Roll back or retire a prompt from an environment without deleting the prompt itself.

```
$ bt prompts unassign my-prompt --environment production
```

— commands and example usage fully specified `product docs`

---

03 **bt sync pull default history window changed** BREAKING

85

`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.

— flag and env var given for reverting default behavior `snapshot-20260912`

---

04 **bt CLI deployment and TLS configuration flags** NEW

85

Adds `--app-public-url` flag and `BRAINTRUST_APP_PUBLIC_URL` env var (bt CLI v0.19.3+) for deployments where the browser-facing app URL differs from the authentication endpoint, and `BRAINTRUST_CUSTOM_CA_BUNDLE` env var (v0.19.3+) so Rust SDK tracing and data uploads can trust private or corporate certificate authorities by supplying PEM certificate contents directly.

— exact flags and env vars given, ready to use `snapshot-20260912`

---

05 **bt observability template commands** NEW

80

Adds `bt observability template` commands (v0.19.0+) to export a project's facets and Loop automations as a portable JSON template and apply it to another project.

— exact command with export/apply mechanism described `snapshot-20260912`

---

06 **bt projects CLI commands with JSON output** NEW

80

Adds `bt projects create`, `view`, and `delete` commands with JSON output support for automation; `view --json` returns project details without opening a browser.

List all projects and get machine-readable JSON output to feed into a CI pipeline or audit script.

```
$ bt projects view --json
```

— runnable example provided (bt projects view --json) snapshot-20260912

07 **Loop automations with managed runtime** NEW 75

Loop automations are scheduled Loop runs with their own instruction, model, and write permissions that leave a resumable read-only thread and can post results to a Slack channel or webhook. Loop now runs in a Braintrust-managed runtime so it persists threads, works across logs, experiments, and datasets, and can create and edit prompts, scorers, datasets, facets, custom views, dashboards, and automations; writes pause for approval unless auto-accept is enabled. It runs on built-in `GPT-5.6 Sol`, `GPT-5.6 Terra`, and `GPT-5.6 Luna` models so threads work without configuring an AI provider.

— rich mechanism described but no exact command or endpoint snapshot-20260912

08 **Brainstore log search streaming** IMPROVED 65

Brainstore log search now streams results as segments finish scanning on an asynchronous execution engine, covering full-text, phrase, and multi-clause ANY_SPAN() trace searches including negated clauses.

— engine-level change with no user action described snapshot-20260912

09 **GLM-5.3 Flash model support** NEW 65

Adds `GLM-5.3 Flash`, a multimodal reasoning model, as a built-in open-source model selectable under the Braintrust provider in playgrounds, prompts, and scorers, or requestable as `glm-5.3-flash` through the Braintrust Gateway.

— model name and gateway request format given snapshot-20260912

10 **bt trace v2 migration for OpenCode and pi** BREAKING 60

`bt trace enable opencode` and `bt trace enable pi` now install v2 of their respective tracing integrations; existing setups must follow the v0.19.3 migration instructions.

— migration instructions referenced but not reproduced snapshot-20260912

11 **Braintrust MCP server integration** NEW 60

The Braintrust MCP server lets Claude Code, Cowork, Cursor, Codex, and other MCP clients query logs, author scorers, configure Topics, and run evaluations directly.

— no setup command given, just a capability list product docs

THINNER COVERAGE BELOW

12 **Patterns automated trace investigations** NEW 55

Patterns is a scheduled Loop-powered investigation that searches traces for recurring problems and trends not covered by existing scorers, saving each finding with supporting traces and suggested fixes; findings can seed new scorers, classifiers, or Loop investigations.

– no CLI/UI entry point named for triggering it `snapshot-20260912`

13 **Metric columnstore** NEW 55

Metric columnstore, enabled in project settings alongside log search optimization, writes columnar artifacts during compaction to speed up loading and sorting traces on high-volume projects.

– project setting named but no exact key or path `snapshot-20260912`

14 **Blind human reviews project setting** NEW 50

Blind human reviews hides peer scores, comments, and aggregates for a row until a reviewer fills in their own scores, preserving reviewer independence throughout a review.

– described as project setting without exact config key `snapshot-20260912`

15 **Dashboard sections** NEW 45

Adds Dashboard sections to group charts into named, collapsible, reorderable, and duplicatable sections on a dashboard.

– UI capability with no exact navigation path `snapshot-20260912`

16 **ElevenLabs SDK tracing integration** NEW 35

Adds ElevenLabs SDK tracing integration to capture audio generation calls in Braintrust for debugging, evaluation, and production monitoring.

– thin one-line description, no mechanism or command `product docs`

17 **Go SDK v0.13.0 reference documentation** NEW 15

Adds Braintrust Go SDK v0.13.0 reference documentation.

– just a docs addition, no capability details `product docs`

└──▷ **BREAKING ON UPGRADE**

! `bt trace enable opencode` and `bt trace enable pi` now install v2 of their respective tracing integrations — existing setups must follow the v0.19.3 migration instructions.

! `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.

WAS THIS USEFUL? 👍 👎

GOVERN

◆ AI/LLM Security

FailproofAI

Feedback ⓘ 1 RELEASE · 2026-09-12 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 1.0.4 introduces credential-exposure detection across agent transcripts with automated local audits, alerting, and native OpenClaw SQLite support, while stripping the audit's scoring and persona sections down to credential findings.

└──▷ WHAT SHIPPED · 7 FEATURES most completely described first ? what's the number?

01 **OpenClaw SQLite transcript support** NEW 73

Supports OpenClaw 2026.9.2's per-agent SQLite databases for live transcript discovery and tailing, with deduplication of rewritten transcripts and continued legacy JSONL support; these sessions now appear in the local dashboard and can be downloaded as JSONL.

— Names exact tool version, storage format, and dashboard download option 1.0.4

02 **Credential-exposure detection in transcripts** NEW 67

Audits now detect credentials exposed in agent transcripts, grouping repeated sightings into a single finding that shows a masked value, source location, carrier agent, and rotation guidance in the audit dashboard.

— Clear mechanism and dashboard fields, but no command or endpoint given 1.0.4

THINNER COVERAGE BELOW

03 **Alerting for exposed credentials** NEW 55

Linux and macOS desktop notifications alert you while a credential remains exposed, with macOS delivery via a per-user LaunchAgent; optional email alerts include the newest masked exposure and its source.

— Names delivery mechanisms (LaunchAgent, email) but no setup steps 1.0.4

04 **Expanded redaction coverage** IMPROVED 55

Extends redaction coverage to JSON, YAML, spaced assignments, camelCase credential names, terminal output, and exported reports.

— Lists concrete formats covered but no config or example 1.0.4

05 **Audit report simplified to credential findings** BREAKING

55

The audit score, letter grade, projected score, persona report, and behavioral sections have been removed; the audit now shows only credential findings and scan statistics.

— Names exact removed sections but no migration guidance 1.0.4

06 **Scheduled local audits every 7 days** NEW

50

Scheduled audits run automatically every seven days after setup, entirely locally without requiring an account.

— States cadence and no-account behavior but no config surface 1.0.4

07 **Claude subagent and workflow transcript audits** IMPROVED

35

Claude audits now include nested subagent and workflow transcripts.

— Brief statement with no further mechanism detail 1.0.4

└──▷ **BREAKING ON UPGRADE**

! The audit score, letter grade, projected score, persona report, and behavioral sections have been removed; the audit now shows only credential findings and scan statistics.

WAS THIS USEFUL?  

ai-safe2-framework

Feedback ⓘ 1 RELEASE · 2026-09-11

NOTES ↗ RANK

Ai-Safe2-Framework provides governance, risk, and compliance controls for securing agentic AI systems and non-human identities.

> SAFE2 CLI 0.3.0 introduces a full `safe2 feedback` subcommand suite for capturing, verifying, and signing agent task evidence, backed by seven new versioned JSON contracts for automation.

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

01 safe2 feedback capture-pytest for test evidence NEW

95

The `safe2 feedback capture-pytest` subcommand runs selected tests and captures fresh evidence, recording the observed exit code separately from test-pass status. Supported options include `--execute`, `--python`, `--cwd`, `--task-id`, `--revision`, `--environment`, `--call-id`, and `--output`.

— Names full flag set and behavior; missing only usage example 2026-09-11_safe2_CLI_0.3.0

02 safe2 feedback receipt for task evidence receipts NEW

75

The `safe2 feedback receipt` subcommand produces JSON or Markdown evidence receipts for agent-generated task claims, configurable via `--artifact-root`, `--format`, and `--output` options.

— Names command and flags but no example of output 2026-09-11_safe2_CLI_0.3.0

03 safe2 feedback report signing and verification NEW

65

`safe2 feedback sign-report` creates a detached signature authenticating test/tool report bytes against an operator-trusted key, and `safe2 feedback verify-report` checks a detached signature against a supplied trusted key.

— Two paired thin commands named but no flag or key detail 2026-09-11_safe2_CLI_0.3.0

THINNER COVERAGE BELOW

04 safe2 feedback usage for resource correlation NEW

55

The `safe2 feedback usage` subcommand correlates declared per-task resource use, separating reported, estimated, and unknown values.

— Command named with behavior but no flags given 2026-09-11_safe2_CLI_0.3.0

05 **safe2 feedback import-junit for JUnit normalization** NEW

55

The `safe2 feedback import-junit` subcommand normalizes supported JUnit XML reports into the framework's evidence format.

— Command named but no supported schema or flag detail 2026-09-11_safe2_CLI_0.3.0

06 **safe2 feedback verify-pytest for capture consistency** NEW

50

The `safe2 feedback verify-pytest` subcommand rechecks a saved pytest capture for internal consistency without re-executing the tests.

— Single-line description with no flags or mechanism detail 2026-09-11_safe2_CLI_0.3.0

07 **safe2 feedback capture-process for command observation** NEW

50

The `safe2 feedback capture-process` subcommand explicitly runs and observes a local command, keeping process capture distinct from test-result assessment.

— Named command but no options or output format listed 2026-09-11_safe2_CLI_0.3.0

08 **Versioned JSON contracts for evidence workflows** NEW

35

SAFE2 CLI 0.3.0 publishes seven versioned JSON contracts to support automation of the new evidence and receipt workflows.

— Count given but no schema names or locations 2026-09-11_safe2_CLI_0.3.0

WAS THIS USEFUL?  

NoLabs nono

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗ CODE ↗

RANK

The nono runtime runs AI agents in isolated environments with policy-controlled filesystem, network, and credential access.

> nono v0.77.0 prepares for a v1.0 API freeze by ripping out legacy profile, flag, and environment-variable aliases, and changes how network-notification sessions bootstrap.

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

01 Legacy profile, flag, env-var and CLI aliases removed

85

BREAKING

Ahead of the v1.0.0 API freeze, nono removes all legacy profile-key aliases, the deprecated `nono policy` subcommand (including `nono policy groups`, `nono policy profiles`, `nono policy show`), legacy CLI flag aliases (e.g. `--allow-net`, `--net-allow`), and legacy environment variable aliases `NONO_NET_BLOCK`, `NONO_NET_ALLOW`, `NONO_ALLOW_PROXY`, `NONO_PROXY_ALLOW`, `NONO_PROXY_CREDENTIAL`, `NONO_EXTERNAL_PROXY`, and `NONO_EXTERNAL_PROXY_BYPASS`. Profile JSON using old flat-schema keys such as `security.allowed_commands` or `policy.add_deny_commands` will now fail to parse instead of warning; migrate to `commands.allow` / `commands.deny` and `nono profile` equivalents, since old env vars will now silently have no effect and old CLI flags cause parse errors.

— Names every removed key, flag, env var, and gives migration path. v0.77.0

THINNER COVERAGE BELOW

02 CLONE_FILES bootstrap for network-notification sessions

IMPROVED

35

Network-notification session startup now uses `CLONE_FILES` bootstrap, replacing the previous fork model.

— Names the mechanism but not behavioural impact or scope. v0.77.0

└──▷ **BREAKING ON UPGRADE**

! Legacy profile-key aliases and the deprecated `nono policy` subcommand now fail to parse instead of warning — any profile JSON still using the old flat-schema keys (e.g. `security.allowed_commands`, `policy.add_deny_commands`) or any script calling `nono policy groups` / `nono policy profiles` / `nono policy show` will break on upgrade. Migrate to `commands.allow` / `commands.deny` and `nono profile` equivalents.

! Legacy environment variable aliases `NONO_NET_BLOCK`, `NONO_NET_ALLOW`, `NONO_ALLOW_PROXY`, `NONO_PROXY_ALLOW`, `NONO_PROXY_CREDENTIAL`, `NONO_EXTERNAL_PROXY`, and `NONO_EXTERNAL_PROXY_BYPASS` are no longer recognized — scripts or CI pipelines that set these will silently have no effect.

! Legacy CLI flag aliases (e.g. `--allow-net`, `--net-allow`) are removed and will now cause a parse error instead of emitting a deprecation warning.

WAS THIS USEFUL? 👍 👎

ToolHive

Feedback ⓘ 1 RELEASE · 2026-09-11 NOTES ↗ RANK

ToolHive runs MCP servers in isolated containers and enforces identity and access policies for local and Kubernetes deployments.

> ToolHive v0.49.0 replaces its vMCP dial-control hook with a workload-aware resolver, ships a generated Go client for its management API, tightens AWS STS role-claim evaluation, and bumps its Go toolchain requirement to 1.27.

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

01 Workload-aware dial-control resolver in vMCP session

BREAKING

93

`pkg/vmcp/session` adds `session.WithDialControlResolver`, replacing the removed `session.WithDialControl`. It lets embedders express a per-backend dial-control policy keyed by workload ID — the resolver is a `func(workloadID string) func(network, address string, c syscall.RawConn) error`, and returning `nil` leaves that backend on `http.DefaultTransport`. Callers using the old hook get a compile error and must wrap it in the new resolver shape.

Enforce per-backend SSRF/private-range dial policy in a vMCP embedder — deny private ranges for most backends, but allow private dialing for a trusted internal workload.

```go
factory := session.NewSessionFactory(registry,
    session.WithDialControlResolver(
        func(workloadID string) func(network, address string, c
syscall.RawConn) error {
            if allowsPrivateDialing(workloadID) {
                return nil
            }
            return denyPrivateRanges
        },
    ),
)
```

— Names the exact API, migration path, and includes a runnable code example v0.49.0

02 OAuth2 upstream token endpoint auth method reverted

BREAKING

80

The default `tokenEndpointAuthMethod` for pre-registered `oauth2` upstream providers is reverted: upgrading from v0.48.0 with a client secret and no explicit

`tokenEndpointAuthMethod` switches back from `client_secret_basic` to POST-body credentials. IdPs requiring HTTP Basic must now set `tokenEndpointAuthMethod: client_secret_basic` explicitly in the `upstreamProviders` config.

Opt a pre-registered OAuth2 upstream into HTTP Basic auth at the token endpoint when your IdP requires it.

```yaml
upstreamProviders:
  - name: my-idp
    type: oauth2
    oauth2Config:
      clientId: my-client
      clientSecretRef:
        name: idp-client-secret
        key: client-secret
      tokenEndpoint: https://idp.example.com/oauth2/token
      tokenEndpointAuthMethod: client_secret_basic
```

— Includes exact config key, values, and a working YAML example `v0.49.0`

---

03 **AWS STS role-claim evaluation tightened** BREAKING 67

AWS STS role-claim evaluation now supports bare-string claims — a string exactly matching a configured `claim` selects its mapped role instead of `fallbackRoleArn`. As a breaking change, role claims that are an object, number, boolean, or null now fail closed with HTTP 403 (`Failed to determine IAM role`) instead of silently receiving the fallback role.

— States the exact fail-closed error and old-vs-new behaviour `v0.49.0`

---

04 **Go 1.27 toolchain requirement and builder image bump** BREAKING 62

The root Go module now requires Go 1.27.0 — builds using `GOTOOLCHAIN=local` pinned to Go 1.26 fail with `go: go.mod requires go >= 1.27`. The default builder image for `go://` workloads is updated from `golang:1.26-alpine` to `golang:1.27-alpine` for workloads without an explicit image pin.

— Gives exact error message and image tags before/after `v0.49.0`

THINNER COVERAGE BELOW

05 **Generated Go client for management API** NEW 25

Ships a dependency-light generated Go client for the ToolHive management API.

— No package path, endpoints, or usage shown beyond one line v0.49.0

└──▷ **BREAKING ON UPGRADE**

! `session.WithDialControl` in `pkg/vmcp/session` is removed; callers get a compile error and must migrate to `session.WithDialControlResolver` wrapping their hook in a `func(workloadID string) func(network, address string, c syscall.RawConn) error` resolver.

! OAuth2 upstream `tokenEndpointAuthMethod` default is reverted: upgrading from v0.48.0 with a pre-registered `oauth2` upstream that has a client secret and no explicit `tokenEndpointAuthMethod` switches back from `client_secret_basic` to POST-body credentials; IdPs requiring HTTP Basic must now set `tokenEndpointAuthMethod: client_secret_basic` explicitly.

! AWS STS role claims that are an object, number, boolean, or null now fail closed with HTTP 403 (Failed to determine IAM role) instead of silently receiving the fallback role.

! Root Go module now requires Go 1.27.0; builds using `GOTOOLCHAIN=local` pinned to Go 1.26 fail with `go: go.mod requires go >= 1.27`. `go://` workloads without an explicit image pin now build against `golang:1.27-alpine` instead of `golang:1.26-alpine`.

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Frontier Models

Anthropic

Feedback ⓘ 1 RELEASE · 2026-09-12 CHANGELOG ↗ RANK

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

> Anthropic added live terminal control over Claude Managed Agents sessions and richer permission-policy telemetry for agent and MCP tool calls.

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

01 Live session control via ant beta:sessions connect NEW

86

The CLI gains `ant beta:sessions connect` to attach your terminal to a running Claude Managed Agents session, letting you send messages live and interactively approve or deny pending tool calls. Passing `--web` instead serves the Claude Console session viewer locally for graphical monitoring and approval.

Interactively supervise a running Managed Agents session from CI or a local terminal — approve or deny tool calls in real time without leaving the shell.

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

Open the graphical Claude Console session viewer locally to monitor and approve tool calls when a browser is available.

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

— Runnable command with documented flag and behavior. changelog-20260912-a850a3bd

02 Permission policy evaluation fields for agent tool calls

IMPROVED

63

Claude Managed Agents permission policies now support `agent.tool_use` and `agent.mcp_tool_use` event types, each carrying an `evaluation` field and an `evaluated_permission` field that report whether a given agent or MCP tool call was run, denied, or paused for approval.

— Named event types and fields but no usage example shown. changelog-20260912-a850a3bd

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · September 12, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue