THE AI TOOLCHAIN

NO. 022

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED SEPTEMBER 10, 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 282 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 57 tools

▾ // THE BRIEF

HIDE SAME ISSUE, SAME PROMPT, TWO WRITERS: Opus 5 | GPT-5.5

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. LangGraph's Deep Agents now hold per-caller OAuth identities and agent-owned secrets, so an agent acts

as the user without secrets in prompts or code. Pinecone made BM25 keyword ranking GA inside the vector index, ending the second search system. And Braintrust's Loop runs on Braintrust-managed infrastructure on a schedule, not your laptop.

GOVERN

Let an agent call a third-party API as the end user, without stuffing tokens into config

LangGraph's Connections resolves credentials at runtime from a 23-service catalog with per-caller OAuth identities, self-registering MCP OAuth and deployment-wide shared grants, so the agent no longer needs a single fat service token that can reach every user's data. NemoClaw's local credential helper injects secrets only into approved executables, and gridctl pins MCP server schemas trust-on-first-use so a server cannot silently change its tool definitions after you approved them.

LangGraph · NemoClaw · gridctl

BUILD

Run hybrid keyword-plus-vector retrieval out of one index instead of stitching two stores together

Full-text BM25 ranking is now GA inside the Pinecone vector index, which removes the usual pattern of running Elasticsearch or OpenSearch alongside the vector DB purely to catch exact identifiers, SKUs and rare tokens that embeddings blur. Dify's dataset-scoped Knowledge Base API keys mean a service that should only read one corpus gets a key that can only read one corpus.

Pinecone · Dify

EVALUATE

Have the eval loop run on a schedule and hand you the pattern instead of a wall of traces

Loop moves off local execution into a Braintrust-managed runtime with scheduled automations, so prompt-and-scorer iteration keeps running after you close the tab; Patterns and Debugger cluster failing traces rather than leaving you to skim them. LangSmith's new paginated experiment-runs endpoint plus composite evaluators and bulk dataset splits make it practical to score one run against several criteria and hold a fixed test slice steady between experiments.

Braintrust

Does **Opus 5** read better? 👍 👎

BUILD

◆ AI Coding Agents

Letta Code

Feedback ⓘ 2 RELEASES · 2026-09-09 → 2026-09-10 NOTES ↗

RANK

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

> Letta Code's latest releases add CLI model management commands, smarter headless agent routing, proactive learning from feedback, safer message queue handling on interrupt, and live credential renewal for Desktop.

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

01 **Model inspection and switching CLI commands** NEW 67

Adds `model get`, `model list`, and `model set` subcommands to the CLI for inspecting and switching models without entering the interactive session.

— Names exact subcommands but no example of use v0.32.1

02 **Resume queue for interrupted messages** NEW 60

Adds `resume_queue` to allow queued user messages parked on interrupt to be resumed without dropping them.

— Names the exact mechanism but no usage example given v0.32.0

THINNER COVERAGE BELOW

03 **Ambient agent inference for headless routing** IMPROVED 58

Infers the ambient agent automatically when using `--computer` for headless routing, removing the need to specify `--agent` explicitly in common cases.

Send a headless message through a named remote computer without specifying an agent ID, letting Letta Code infer the ambient agent automatically.

```
$ letta -p --computer "work-laptop" "run the nightly audit"
```

— Concrete flag plus a runnable example command v0.32.1

04 **Proactive learning from human feedback** NEW 31

Teaches agents proactive learning from human feedback, enabling prompt and memory updates driven by conversational corrections.

— Described only in prose, no mechanism or surface named v0.32.1

05 **Live Desktop credential renewal** IMPROVED 28

The server now accepts renewed Desktop credentials without requiring a restart.

— Behaviour stated but no config key or command shown v0.32.0

WAS THIS USEFUL?  

Cline

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗ CODE ↗ RANK

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

> Cline desktop v0.0.25 focuses on account and provider ergonomics — a dedicated session-import page, clearer provider/model pickers, expanded local-CLI authentication, and a refreshed model catalog including a breaking change to ChatGPT Subscription (Codex) model limits.

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

01 ChatGPT Subscription (Codex) model picker overhaul

BREAKING

85

The Codex model picker now lists only models your plan can actually use, with `gpt-5.6-terra` as the new default. Per-model context caps of 400K/272K/128K replace the previously inherited API limit of 1.05M, and the retired models `gpt-5.4` and `gpt-5.4-mini` are removed, so accounts using either will have their default model changed.

— Names exact model IDs, caps and retirement, but no migration command desktop-v0.0.25

02 Local CLI provider authentication expanded NEW

70

Providers that authenticate through a local CLI — Claude Code and Codex CLI — can now start sessions without an API key. OpenCode is now also treated as a local CLI provider, showing the local CLI notice instead of a non-functional browser sign-in button, and authenticates using credentials stored by the `opencode` CLI.

— Explains mechanism and names all three CLI providers desktop-v0.0.25

03 Model catalog additions and default refresh NEW

65

Adds NaN (`nan.builders`) to the model catalog and refreshes default models for 36 providers, including Bedrock, Vertex, OpenRouter, Kilo, GitHub Copilot, Gemini, Cerebras, Fireworks, Requesty, and Vercel AI Gateway.

— Names new provider and lists many affected providers, no per-provider detail

desktop-v0.0.25

THINNER COVERAGE BELOW

04 Session import moved to dedicated Settings page IMPROVED

55

Session import from Claude Code, Codex, and opencode now has its own dedicated page in Settings instead of being a row buried in General.

— Clear before/after and named sources, but only a UI location `desktop-v0.0.25`

05 Model picker distinguishes same-named models IMPROVED

45

The model picker now distinguishes models that share a name, and Cline Pass subscription models are listed separately from the free fallback tier.

— Describes the change but not the underlying mechanism `desktop-v0.0.25`

06 Provider picker shows configured providers IMPROVED

35

The composer's provider picker now marks which providers you have already configured.

— Bare description with no mechanism or exact location `desktop-v0.0.25`

└──▷ **BREAKING ON UPGRADE**

! ChatGPT Subscription (Codex) accounts using `gpt-5.4` or `gpt-5.4-mini` as their model will find those models removed; the default moves to `gpt-5.6-terra`.

WAS THIS USEFUL?  

Diagram Design

Feedback ⓘ changes since 2026-08-12 CODE ↗

Diagram Design is a Claude Code skill that generates editorial HTML and SVG diagrams and imports draw.io or Mermaid sources.

Diagram Design gains ten new chart types, a headless-render linter, named client profiles, and native Droid plugin packaging.

└──▷ **TRY IT**

Catch clipped or collapsed SVG diagrams in CI before they reach reviewers — runs headless Chromium and diffs overflow-released screenshots against the authored viewport.

```
$ python3 scripts/lint-render.py
```

Run the adversarial sankey verification suite to confirm the checker fires on real defects (ribbon narrowing, node volume loss, label mismatch) and stays quiet on clean files.

```
$ python3 scripts/test-verify-sankey.py
```

Adds `scripts/lint-render.py`, a headless-Chromium linter that screenshots diagrams at authored viewport size and with `overflow` released, diffs the two to catch clipped SVG paint, collapsed SVGs, sideways page overflow, and runtime errors that source-level linting cannot detect.

Adds `verify-beeswarm.py` with nine geometric invariants — shared value scale, no-overprint, and more — verifying that every drawn dot sits at its declared value on the linear axis and that crowding is resolved only by swarm offset, never by moving a dot along the value axis.

Adds `test-verify-sankey.py`, an adversarial test suite for `verify-sankey.py` that covers both polarities (fires when it should, stays quiet when it should not) across ribbon narrowing, node volume loss, stage carry-loss, label/bar mismatch, and dark-variant drift.

Adds a treemap type for part-of-whole by area (`feat(types): add treemap`).

Adds a dumbbell as a Bar variant (`feat(bar): add dumbbell`).

+8 more

WAS THIS USEFUL? 👍 👎

DeepSeek Harness

Feedback ⓘ 1 RELEASE · 2026-09-10 CODE ↗

RANK

DeepSeek Harness is an open-source agent harness for building coding agents with plugins.

> DeepSeek Harness's rc.1 release adds a new default model, DeepSeek-V41-Flash, with text and image input support, and adds a small guide UI enhancement for short entry lists.

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

01 **DeepSeek-V41-Flash as new default model** BREAKING 73

Adds a `deepseek-flash` model entry (`DeepSeek-V41-Flash`) to the default catalog with `inputModalities: ['text', 'image']` and `systemPromptUpdate: 'in-history'` , and sets it as the new default for Chat Completions, replacing the previous `DeepSeek-V4-Flash` default. Any integration relying on the previous default without explicitly setting a model ID will now target the new model.

— Names exact model ID, config fields and breaking default change `dsh-v0.1.5-rc.1`

THINNER COVERAGE BELOW

02 **Optional description field in sidebar guide entries** NEW 57

Adds an optional `description` field to `SidebarRightGuideEntry` for a one-line description shown beneath a capsule title in the guide, displayed when the entry list is short.

— Names the exact field and its display condition `dsh-v0.1.5-rc.1`

└──▷ **BREAKING ON UPGRADE**

! The default Chat Completions model is now `DeepSeek-V41-Flash` (`deepseek-flash`); any integration that relied on the previous default (`DeepSeek-V4-Flash`) without explicitly setting a model ID will now target the new model.

WAS THIS USEFUL? 👍 👎

OpenAI Codex CLI

Feedback ⓘ 2 RELEASES · 2026-09-09 → 2026-09-10

NOTES ↗ CODE ↗ RANK

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

> Codex CLI's headline addition this window is live WebRTC voice chat in the TUI, paired with experimental `--worktree` session isolation, macOS Secure Enclave user verification, enterprise OIDC login, GPT-6-Astra model support, inline question answering, and daemon-sharing improvements; the long-deprecated `codex mcp-server` entry point has now been removed.

└──▷ WHAT SHIPPED · 7 FEATURES most completely described first ? what's the number?

01 **Worktree isolation for sessions** NEW 90

Adds experimental worktree support via `--worktree` flag or `/worktree` command to create isolated checkouts for new or forked sessions, with browse and resume capabilities.

Run a task in an isolated worktree so the main checkout is never touched — useful for risky refactors or parallel experiments.

```
$ codex --worktree 'Refactor the auth module to use the new token interface'
```

— Named flag and command with a runnable example provided. rust-v0.154.0

02 **/copy command enhancements** IMPROVED 70

Adds the `/copy` command to copy status output or individual session fields, with rich-text formatting preserved when copying responses; a later release adds completed commentary in the `/copy` picker.

— Named command across two releases, usable directly from the terminal.

rust-v0.155.0-alpha.1 rust-v0.154.0

THINNER COVERAGE BELOW

03 **GPT-6-Astra model support** NEW 45

Adds GPT-6-Astra to the model picker and Amazon Bedrock catalogs.

— Names the model and catalogs but no usage instructions. rust-v0.154.0

04 **Inline question answering during tasks** NEW 45

Enables inline question answering — answer questions with suggested choices or custom text while Codex continues working without losing your main draft.

— Explains behaviour but no named command or flag to trigger it. `rust-v0.154.0`

---

05 **Removal of codex mcp-server entry point** BREAKING 45

The `codex mcp-server` entry point is no longer available (previously deprecated).

— Names the exact removed entry point but no migration path given. `rust-v0.154.0`

---

06 **Enterprise OIDC login and logout** NEW 25

Adds staged enterprise OIDC login and coordinated logout.

— Bare mention with no detail on flow or configuration. `rust-v0.155.0-alpha.1`

---

07 **Worktree classification in thread telemetry** NEW 25

Adds worktree classification to thread telemetry.

— One-line internal telemetry note with no further detail. `rust-v0.155.0-alpha.1`

└──▷ **BREAKING ON UPGRADE**

! The `codex mcp-server` entry point is no longer available (previously deprecated).

WAS THIS USEFUL?  

Charm Crush

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

RANK

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

> Crush v0.93.1 lets users disable mouse support in the TUI and shows which model served each conversation turn when using Hyper.

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

01 Mouse support toggle in TUI NEW

80

Adds `option ui mouse false` to `crushrc` and `{ "options": { "tui": { "mouse": false } } }` to `crush.json` to disable mouse support in the TUI, so terminal text selection and right-click work natively. Also toggleable via the command palette.

Disable mouse support so terminal text selection and right-click work natively during a Crush session.

```
$ option ui mouse false
```

— Exact config keys and command given v0.93.1

THINNER COVERAGE BELOW

02 Routed model display for Hyper turns NEW

30

Shows the routed model that served each conversation turn when using Hyper.

— Describes behavior but no UI path or mechanism v0.93.1

WAS THIS USEFUL? 👍 👎

mex

Feedback ⓘ 1 RELEASE · 2026-09-09 CODE ↗ RANK

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

> mex 0.8.1 adds a visual Context graph to the Project Hub, an Inbox for proposing Wiki knowledge changes, open-team Relays with a new draft-save command, configurable agent logging modes, and code graph ignore globs.

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

01 **Configurable agent logging modes** NEW 90

Adds `mex logging` command with `significant` (default), `checkpoints`, and `manual` modes for checkout-local agent logging preferences, also configurable in Hub Settings. Adds `AgentLoggingModeSchema` (`significant` , `checkpoints` , `manual`), `AgentLoggingPolicySchema` , and `AgentLoggingUpdateRequestSchema` to the Hub contracts API.

Switch agent logging to checkpoint mode to capture more granular activity without full verbose output.

```
$ mex logging checkpoints
```

— Command, mode names, and API schemas all named v0.8.1

02 **Open-team Relays and draft workflow** NEW 85

Adds `mex relay draft save --from <draft.json>` to save open-team Relay local drafts through the existing signed workflow, with Hub and CLI showing audience, current eligibility, and the sharing boundary. Relays can now be open-to-team so eligible active Members can take them, including teammates who join later, and local drafts may leave recipients undecided.

Save an open-team Relay draft from a JSON file so any eligible active member can pick it up.

```
$ mex relay draft save --from handoff-draft.json
```

— Names exact command and describes eligibility mechanism v0.8.1

03 **Graph ignore globs for code indexing** NEW 75

Adds additive Graph ignore globs under the `graph.ignore` key in `.mex/config.json` , with repository-relative validation that behaves consistently

across platforms.

Exclude generated or vendored directories from the code graph to reduce noise in who-calls results.

```json
{
  "graph": {
    "ignore": ["vendor/**", "dist/**", "**/__generated__/**"]
  }
}
```

— Exact config key and file path given with example v0.8.1

04 **Inbox contributions to Wiki knowledge** NEW 65

Adds Inbox contributions for one addition or correction to existing architecture, component, convention, decision, pattern, or guide knowledge; local drafts publish as Git-shareable Markdown proposals and explicit approval writes the existing Wiki knowledge.

— Workflow explained but no command or UI path shown v0.8.1

05 **Context graph in Project Hub** NEW 60

Adds a Context graph in the Project Hub showing existing Wiki entities, relationships, and direct code groundings, with type filters, selection details, pan/zoom, and a list alternative.

— UI feature described but no exact navigation path v0.8.1

THINNER COVERAGE BELOW

06 **Member reactivation and bounded Timeline notes** NEW 35

Adds member reactivation with the original identity; managed agent instructions now retrieve relevant bounded Timeline notes without automatically promoting those notes to accepted project knowledge.

— Thin description, no mechanism or interface detail v0.8.1

WAS THIS USEFUL? 👍 👎

Anthropic Claude Code

Feedback ⓘ 1 RELEASE · 2026-09-09

NOTES ↗ CODE ↗ RANK

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

> Claude Code v2.1.267 adds a flag for live system-prompt iteration and a cross-provider cap on model effort level.

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

01 **Live system prompt iteration flag** NEW 85

The `--system-prompt-snapshot off` flag makes Claude Code re-render the system prompt fresh on every request instead of replaying the recorded snapshot, enabling live iteration on prompt text without restarting the session.

Iterate on a system prompt without restarting the session — each request re-reads the prompt as written rather than replaying the snapshot recorded at session start.

```
$ claude --system-prompt-snapshot off -p 'Refactor the auth module'
```

— Names exact flag and runnable example command. v2.1.267

02 **maxEffortLevel cap across providers** NEW 65

A new `maxEffortLevel` setting, configurable top-level or per model under `modelSettings`, caps the effort level across all providers including Bedrock, Vertex, and Foundry, while still letting users select a lower level.

— Names config key and scope, but no example usage. v2.1.267

WAS THIS USEFUL? 👍 👎

All Hands AI OpenHands

Feedback ⓘ 1 RELEASE · 2026-09-09

NOTES ↗ RANK

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

> OpenHands v1.17.0 tightens automation access control with split view/manage permissions, adds LLM provider connections for cloud deployments, and rounds out Agent Canvas with conversation tagging, custom manifests, and several automation-management UI refinements.

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

01 **View/manage permission split for automations** NEW 60

Automation permissions are now split into separate view and manage roles, with a creator escape hatch that automatically grants automation creators full manage rights over automations they created regardless of their team-wide role.

Give automation creators full control over their own automations while restricting other team members to view-only access.

> 📍 In the automations settings, assign view permission to team members and manage permission to automation owners; the creator escape hatch automatically grants creators full manage rights over automations they created regardless of team-wide role.

— Mechanism and roles named, with a worked example v1.17.0

THINNER COVERAGE BELOW

02 **Agent Canvas automation management enhancements** IMPROVED 50

Canvas gains support for generic custom automation manifests, lets users edit a custom cron expression directly from the edit-automation modal, shows manifest-declared value statements on dashboard cards and rows, makes the automation run UI task-outcome aware, and adds a unified cloud settings entrypoint for Agent Canvas.

— Five distinct UI/manifest additions named but only briefly v1.17.0

03 **Conversation tags and filtering** NEW 30

Adds conversation tags and filtering in the conversation panel.

— Names the panel but no detail on how tags work v1.17.0

04 **LLM provider connections on cloud deployments** NEW

25

Enables LLM provider connections on cloud deployments.

— Only a one-line description, no mechanism given v1.17.0

05 **Message timestamps on hover** NEW

15

Displays message timestamps on hover.

— Single-line description with no further detail v1.17.0

WAS THIS USEFUL? 👍 👎

Superset

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

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

> Superset's desktop v1.28.0 release adds MCP plugin integrations for agents, cloud workspace creation, iOS Live Activities showing agent status, and a round of polish across Pages, mobile terminals, and the paywall flow.

└──▷ **WHAT SHIPPED · 9 FEATURES** most completely described first ? what's the number?

01 **Host version display and remote updates** NEW 50

The desktop app now shows host versions and supports updating standalone hosts directly from the app.

— Describes both viewing and updating hosts desktop-v1.28.0

02 **MCP plugin integrations skill for agents** NEW 45

Adds an integrations skill that lets the agent call plugin MCP tools directly from within its workflow.

— Names MCP mechanism but no usage steps given desktop-v1.28.0

03 **PR summary opens as workspace pane** IMPROVED 45

The PR summary can now be opened as a workspace pane directly from the Changes pane.

— Clear UI navigation path but no further mechanism desktop-v1.28.0

04 **iOS Live Activities for agent status** NEW 45

Shows agent status on the iOS Lock Screen and Dynamic Island.

— Names specific iOS surfaces but no interaction detail desktop-v1.28.0

05 **Pages product polish: onboarding, comments, theming** IMPROVED 40

Adds Pages onboarding and polishes the page creation flows, splits comment mode out from the comments panel, and adds theming for published pages.

— Groups thin Pages increments but lacks mechanism detail desktop-v1.28.0

06 **Shareable link for automations** NEW 40

Users can copy a shareable link to an automation.

— Concrete action but no detail on link scope or permissions desktop-v1.28.0

07 **Cloud workspace creation from sidebar** NEW 40

A cloud workspace can be created directly from the sidebar.

— Clear entry point but no further mechanism described `desktop-v1.28.0`

08 **Mobile terminal and session persistence** IMPROVED 40

On mobile, switched-to terminals are kept warm with a cached buffer, and the app now remembers the last session tab per workspace.

— Groups two mobile session behaviours, both described briefly `desktop-v1.28.0`

09 **Paywall flow closed via Stripe outcomes** IMPROVED 30

Closes the paywall funnel through Stripe payment outcomes.

— Names Stripe but no mechanism or flow detail `desktop-v1.28.0`

WAS THIS USEFUL? 👍 👎

Zed

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

Zed is an open-source code editor that runs collaborative development workflows.

> Zed v1.19.2 adds call hierarchy navigation, a tabular data preview for CSV/TSV/SSV/PSV files, and a batch of new panel and settings-UI options, while flipping project search on type to enabled by default.

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

01 **Call hierarchy navigation** NEW 85

Adds call hierarchy support via `call hierarchy: show incoming calls` and `call hierarchy: show outgoing calls` commands, with default keybindings `cmd-k cmd-h` on macOS and `ctrl-k ctrl-h` on Linux and Windows.

Navigate a codebase call graph to trace who calls a function and who it calls, without leaving the editor.

> 📍 Open the command palette and run `call hierarchy: show incoming calls` (macOS: cmd-k cmd-h) to see all callers of the symbol under the cursor.

— Full commands, keybindings, and a runnable usage example. v1.19.2

02 **New panel and file-tab settings** NEW 85

Adds `outline_panel.multi_buffer_hide_symbols` setting and `outline panel: toggle symbols` action to show only files and directories in multi-buffer Outline Panel views; `reveal_if_open` setting to reveal buffers already open in another pane instead of opening them in the active pane; `folder_indicator` setting for showing disclosure chevrons and folder icons together in the Project Panel, Outline Panel, and Git Panel; `close_panel_on_toggle` setting added to the Settings UI; and Open File on Remote and Copy Remote File URL options added to file tabs and the Project Panel.

Hide symbols in the Outline Panel when working across a multi-buffer view to reduce noise and focus on file structure.

```json
{ "outline_panel": { "multi_buffer_hide_symbols": true } }
```

— Names every setting key and action, one has a runnable config example. v1.19.2

03 **Tabular data preview for delimited files** NEW 75

Adds `cmd-shift-v` and `cmd-k v` on macOS and `ctrl-shift-v` and `ctrl-k v` on Linux and Windows to open a tabular data preview for CSV, TSV, SSV, and PSV files.

— Exact keybindings and supported formats given, no deeper mechanism. v1.19.2

04 **Project search on type enabled by default** BREAKING 70

Project search on type is now enabled by default, automatically triggering search as you type; set `{ "search": { "search_on_type": false } }` in `~/.config/zed/settings.json` to restore the previous manual-invocation behavior.

Suppress search triggering automatically as you type in a project-wide search, restoring manual invocation.

```json
{ "search": { "search_on_type": false } }
```

— Exact config key and file path to revert the default behavior. v1.19.2

05 **Modal editing selection and text objects** NEW 60

Adds Visual Line selection with `V` in terminal vi mode, and Helix mode support for closest-pair text objects with `m i m` and `m a m`.

— Exact keys named but no further mechanism or scope described. v1.19.2

THINNER COVERAGE BELOW

06 **Command palette recency-based ordering** IMPROVED 40

Improves command palette ordering by sorting commands by recency of use, ranking previously used commands above unused ones while filtering.

— Describes the sorting mechanism but no configurable surface. v1.19.2

07 **LSP 3.18 Markdown diagnostics rendering** NEW 35

Adds support for rendering LSP 3.18 Markdown diagnostic messages from language servers.

— Names the protocol version but is an automatic, non-actionable change. v1.19.2

08 **Renaming Terminal Threads** NEW 30

Adds support for renaming Terminal Threads from the Threads Sidebar.

— Names the UI location but no further detail. v1.19.2

09 **Effort-based reasoning for OpenRouter** NEW 25

Adds support for effort-based reasoning when using OpenRouter models.

— Names the provider but not the mechanism or configuration. v1.19.2

10 **Automatic language detection for untitled buffers** NEW

20

Adds automatic language detection for untitled buffers.

— Bare description with no mechanism or scope given. v1.19.2

└──▷ **BREAKING ON UPGRADE**

! Project search on type is now enabled by default; set `{ "search": { "search_on_type": false } }` to restore the previous behavior.

WAS THIS USEFUL?  

Trailhq Graft

Feedback ⓘ 1 RELEASE · 2026-09-10 CODE ↗ RANK

Graft builds a code knowledge graph that gives coding agents compact, repository-specific context through a CLI and MCP server.

> Graft v0.18.0 introduces Trail Brain integration that injects repo-mined rules into every `ask`, plus checkout verification before mining, expanded `graft init` wiring for Claude Code, and dollar-value savings reporting in Claude Code sessions.

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

01 **`graft init` writes Claude Code wiring to `~`** IMPROVED

75

`graft init` now writes Claude Code wiring into `~` so worktrees and shells created off the same home keep graft available even when repo-level files are excluded by `.gitignore`.

— Names exact command and file location, actionable v0.18.0

02 **Trail Brain integration for `ask`** NEW

60

Graft can now build a Trail Brain from a repo and carry its rules into every `ask`, so retrieval is shaped by accumulated team context rather than the code graph alone.

— Explains mechanism but no exact command flags shown v0.18.0

THINNER COVERAGE BELOW

03 **Pre-mining checkout verification** NEW

50

A brain now checks the working copy against the repo it expects before mining, preventing rules from being extracted from the wrong tree.

— Describes safeguard mechanism but no config surface named v0.18.0

04 **Dollar-value savings on Claude Code statusline** IMPROVED

45

Claude Code sessions now report the dollar value of what graft saved, not just token counts, making the cost payoff of graph-routed lookups visible on the statusline.

— States what changed but no metric or config detail v0.18.0

WAS THIS USEFUL? 👍 👎

Alibaba Qwen Code

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗

RANK

An open-source AI coding agent that lives in your terminal.

> Qwen Code v0.23.3 introduces ACP-based subagent delegation to external agents such as Claude Code, adds an OpenAI Responses API provider backend, expands reasoning presets, and raises the default workspace limit for `qwen serve`, alongside session registry and web-shell interface improvements.

└──▷ **WHAT SHIPPED · 7 FEATURES** most completely described first ? what's the number?

01 **Higher workspace limit in qwen serve** IMPROVED 60

Raises the default workspace limit in `qwen serve` to 256 workspaces.

— Names exact command and new numeric limit v0.23.3

THINNER COVERAGE BELOW

02 **Subagent delegation over ACP protocol** NEW 55

Qwen Code can now delegate a subagent turn to an external agent over ACP (Agent Communication Protocol), with Claude Code as the first supported peer.

— Names protocol and first peer but gives no usage command v0.23.3

03 **Session registry improvements for daemon sessions** IMPROVED 50

Registers daemon-managed sessions in the session registry and enables them to send peer messages over ACP; also records session kind in the registry and adds documentation of the registry and wire contract.

— Describes registry mechanics but no concrete API names v0.23.3

04 **Web-shell interface UX improvements** IMPROVED 50

Adds unified session sources to the web-shell interface, improves session overview navigation and details, makes the product name and logo configurable, keeps answered questions visible outside processed summaries, and renders URLs in user messages as clickable links.

— Lists several UI changes without config keys or flags v0.23.3

05 **OpenAI Responses API provider backend** NEW 35

Adds the OpenAI Responses API content generator as a supported provider backend.

— Names the API but no config or setup detail v0.23.3

06 **Expanded reasoning presets for models** IMPROVED 30

Expands reasoning presets for Kimi, Qwen, and DeepSeek models.

— Names affected models but not the presets themselves v0.23.3

07 **Newest models shown first in provider list** IMPROVED 25

Shows newest discovered models first in the provider model list.

— Bare description of a minor ordering change v0.23.3

WAS THIS USEFUL? 👍 👎

Sourcegraph Amp

Feedback ⓘ 1 RELEASE · 2026-09-10 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 introduced a fully customizable Mode Dial, letting users tune per-role models and reasoning effort, build custom dials from plugin agents, switch modes from the CLI, and share dial configs across a team.

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

01 **Customizable Mode Dial for models and agents** NEW 81

The Settings → Mode Dial screen gains a Tune Modes tab to independently choose the model and reasoning effort for a builtin mode's main agent, Oracle, or subagents, and a Build Dial tab to place custom plugin agents alongside builtin modes with drag-to-order and a 2–4 mode limit per dial. Dial modes can be switched from the CLI with Ctrl+S, and workspace admins can set a shared dial for their team, with personal dial choices taking precedence over the team default.

— Names exact settings tabs, shortcut, and mode limits from one release.

Customize Your Dial (2026-09-10)

WAS THIS USEFUL? 👍 👎

herdr

Feedback ⓘ 1 RELEASE · 2026-09-07 BLOG ↗ RANK

Herdr is a coding-agent runtime that manages persistent terminal sessions across local and SSH machines.

> Herdr 0.9 introduces multi-machine management, letting a single TUI client connect to and control remote servers over SSH alongside local sessions.

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

01 Multi-machine support via `herdr machine add` NEW

87

Herdr adds the `herdr machine add <name>` subcommand to connect remote machines over SSH, bringing their workspaces, tabs, and agents into the same local TUI client. This is enabled by a redesigned client/server architecture where the outer UI renders on the client while each server independently owns its own sessions, letting one TUI aggregate multiple independent servers across machines.

Add a remote machine (VPS, Mac mini, etc.) so its agents and workspaces appear alongside local ones without opening a separate terminal tab.

```
$ herdr machine add workbox
```

— Names exact subcommand and explains client/server mechanism behind it.

launch-20260907-274f517e

WAS THIS USEFUL? 👍 👎

AGENT

◆ MCP Servers & Tooling

gridctl

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗ RANK

Gridctl is a command-line tool that manages and routes Model Context Protocol servers, skills, and tools.

> gridctl reached v1.0.0-rc.1, turning into a full stack-as-code MCP gateway: `gridctl apply` deploys a YAML stack while new commands add client auto-linking, catalog search, token-reduction analysis, trust-on-first-use schema pinning, a SKILL.md skill library, containerized Python MCP servers from PyPI or Git, model routing, and Sigstore-signed release provenance.

└──▷ **WHAT SHIPPED · 21 FEATURES** most completely described first ? what's the number?

01 **Client auto-linking with `gridctl link`** NEW 95

`gridctl link`, `gridctl link <client>`, and `gridctl link --all` auto-detect installed LLM clients (Claude Desktop, Claude Code, Cursor, Antigravity, Grok) and inject gateway config pointing at the gateway's MCP endpoint, exposed at `http://localhost:8180/mcp` as streamable HTTP with an SSE endpoint, connectable by any MCP client. A `link:` block in `stack.yaml` idempotently relinks the listed clients on every `gridctl apply`, and `gridctl destroy --unlink` tears down the stack while removing injected client config entries.

— Commands, config key, and endpoint all named with mechanism. product docs

02 **Encrypted variable store and scoped delivery** NEW 93

`gridctl var explain <key>` shows a variable's resolution, declaration, and consumers without exposing its value, and `gridctl var scan --staged` checks for exact secret values before commit. Variables are referenced as `${var:<KEY>}` in a `secrets.sets` block in `stack.yaml`, delivered from an encrypted-at-rest store, with reserved-name enforcement for `GRIDCTL_*`, `OP_CONNECT_TOKEN`, and `OP_SERVICE_ACCOUNT_TOKEN`; scoped variable delivery now enables per-scope secrets and variables across stacks.

— Commands, config block, syntax, and reserved names all named. v1.0.0-rc.1

03 **Token-reduction findings via `gridctl optimize`** NEW 93

`gridctl optimize`, with `--format json` and `--severity warn,critical` flags, surfaces actionable token-reduction findings — unused servers/tools, schema overhead, format-conversion shortfalls — with projected weekly token impact and paste-ready YAML remediations; `--format json` emits a machine-readable `OptimizeReport`.

Audit a running gateway for wasted tokens before a long agent run — narrow to only warn and critical findings and get machine-readable output for downstream processing.

```
$ gridctl optimize --severity warn,critical --format json
```

Surface high-impact token-reduction opportunities across the running gateway and pipe the structured report into a CI artifact.

```
$ gridctl optimize --format json --severity warn,critical >
optimize-report.json
```

— Exact flags, output format, and remediation mechanism all named. product docs

---

04 **Trust-on-first-use schema pinning** NEW 93

`gridctl pins verify` (exit `1` on drift), `gridctl pins diff <server>` (per-tool before/after plus injection scan findings), and `gridctl pins approve <server>` implement trust-on-first-use pinning of tool definitions, flagging drift on apply and scanning for injection signals such as hidden instructions, sensitive-file references, hidden Unicode, and cross-server tool shadowing.

Verify that no upstream MCP server has silently changed a tool schema or injected hidden instructions since the last pin — fail CI if drift is detected.

```
$ gridctl pins verify
```

Detect prompt-injection drift in a GitHub MCP server's tool definitions before an agent ever sees the changed schema.

```
$ gridctl pins diff github
```

— Commands, exit codes, and specific injection signals all named. product docs

---

05 **Output format conversion for tool results** NEW 90

The `output_format` config key, settable at the `gateway:` level and per-server under `mcp-servers:`, converts tool call results to `toon`, `csv`, `json`, or `text` before they reach the client, cutting token consumption by 25–61% for tabular and key-value data; payloads over 1 MB pass through unchanged.

Cut token consumption on a high-volume analytics server by converting its JSON responses to CSV at the gateway level, while leaving other servers on the default format.

```yaml
gateway:
  output_format: json
mcp-servers:
  - name: analytics
    output_format: csv
```

Reduce downstream token cost for a server returning large tabular payloads by converting its output to CSV before it reaches the client.

```yaml
gateway:
  output_format: toon
mcp-servers:
  - name: analytics
    output_format: csv
```

— Config key, all four formats, and exact reduction numbers named. product docs

06 **SKILL.md skill library and projection** NEW 88

`gridctl skill list`, `gridctl activate <skill>`, and `gridctl skill project sync` manage a `SKILL.md` registry exposed to upstream MCP clients as prompts, with projection to Claude Code, OpenCode, Copilot, and Gemini CLI dialects tracked by a shared lockfile; `gridctl skill pins` adds per-file digest tracking of skill documents, holding drift for human approval with injection heuristics.

— Commands, file format, target dialects, and lockfile all named. product docs

07 **Containerized Python MCP servers from PyPI/Git** NEW 88

`source.type: pypi` and `source.type: git` config keys in `stack.yaml` generate containerized Python MCP servers, run as non-root on Python/uv base images, built from a public PyPI package or a Git repository (with `ref` pinning and optional `path` / `runtime` fields) — no Dockerfile required.

Run a PyPI package as a containerised MCP server at a pinned version, without writing a Dockerfile.

```yaml
mcp-servers:
  - name: fetch
    source:
      type: pypi
      package: mcp-server-fetch
      ref: 2026.8.18
```

Run a Python MCP server from a specific git commit and subdirectory, pinned for reproducibility.

```yaml
mcp-servers:
  - name: time
    source:
      type: git
      url: https://github.com/modelcontextprotocol/servers.git
      ref: d73f99efbfd40c3aa1b61e88728b3d49fb52608f
      path: src/time
      runtime: python
```

— Config keys and runnable examples pin exact package/commit. v1.0.0-rc.1

08 **Stack-as-code deployment via `gridctl apply`** NEW 85

`gridctl apply stack.yaml` deploys a full MCP gateway from a YAML stack spec in one command, linking downstream servers (OAuth remote, containerized stdio, or local command) to upstream LLM clients.

— Command, YAML spec, and three downstream server types named. product docs

09 **Tool surface control via groups and whitelists** NEW 81

`gridctl groups` lists tool groups, member counts, and their `/groups/{name}/mcp` endpoints. Per-server and per-group `tools:` whitelists, `exclude:` lists, `groups:` bundles, and `clients:` restrictions in `stack.yaml` control which tools each linked client may access.

— Command, endpoint pattern, and config keys named. product docs

10 **Import existing MCP client configs** NEW 73

`gridctl import` scans existing client MCP configs, dedupes discovered servers, and appends selections to `stack.yaml`, offering any plaintext secrets it finds into the encrypted variable store.

– Command and dedupe/secrets mechanism named, single source. product docs

11 **OAuth authentication for remote MCP servers** NEW 73

The per-server `auth: type: oauth` config key in `stack.yaml` handles OAuth-protected remote MCP servers, including encrypted-on-disk token storage, refresh, and a `needs auth` deployment state.

– Config key and states named, no CLI command given. product docs

12 **Stack export for safe sharing** NEW 67

`gridctl export` re-reads the running deployment's stack file with `${var:<KEY>}` variable references preserved and inline credentials blocked, for safe sharing; the web UI's Stack spec view offers the same output via an Export YAML action.

– Command and safeguard named, UI path noted but not detailed. product docs

13 **Per-server rate limiting** NEW 67

`gridctl limits` displays every configured rate limit and its current state; a `limits.rate_limits` config block per server sets `calls_per_minute` and `burst` fields.

– Command and config fields named, no runtime behavior described. product docs

14 **Stack diff and validation commands** NEW 65

`gridctl plan stack.yaml` diffs a stack spec against the running deployment state before applying, and `gridctl validate stack.yaml` lints and schema-checks a stack spec, exiting with codes `0`, `1`, or `2`.

– Named commands and exit codes but no deeper mechanism. product docs

15 **Catalog search and add commands** NEW 63

`gridctl search <term>` finds MCP servers in the built-in catalog and the MCP Registry, and `gridctl add <server>` appends a matched entry to `stack.yaml` by name.

– Commands named but minimal behavioral detail. product docs

16 **Code mode for tool-capped clients** NEW 62

The `gateway.code_mode: on` config key reduces the tool surface presented to clients that cap the number of MCP servers or tools they can see, such as Antigravity's 100-tool limit, letting large stacks work with those clients.

– Config key and concrete client limit named, mechanism thin. product docs

17 **Gateway API authentication** NEW 60

The `gateway.auth` config key enables bearer token or API-key authentication on the `/mcp` endpoint and grouped `/groups/{name}/mcp` endpoints.

— Config key and endpoints named, minimal further detail. product docs

THINNER COVERAGE BELOW

18 **Signed release provenance with Sigstore** NEW 59

Binary releases are signed with a `provenance.sigstore.json` bundle, making v1.0.0-rc.1 the first provenance-covered production release, verifiable via the release verification procedure.

— Artifact name and version given, verification steps not detailed here. v1.0.0-rc.1

19 **Token-usage observability in Metrics workspace** NEW 59

A token-usage observability layer counts tokens per server, replica, client, and tool, and charts throughput and format-conversion savings in the Metrics workspace.

— Metrics dimensions named but no CLI or API access point. product docs

20 **Model routing policy via `gridctl models`** NEW 54

The `gridctl models` subcommand projects model routing policy from a running gateway, and the web UI adds a model routing surface for viewing and managing that policy.

— Command named but routing mechanism itself not described. v1.0.0-rc.1

21 **Web UI with Tools, Metrics, and Library workspaces** NEW 48

A web UI accessible via `gridctl` includes a Tools Workspace, Metrics Workspace, a Stack spec view, and a Library workspace for skill authoring.

— Workspace names given but no navigation or mechanism detail. product docs

WAS THIS USEFUL?  

◆ AI Agent Frameworks

Skyvern

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

Skyvern automates browser-based workflows with vision models and agents, completing form-driven tasks on sites it has not seen before.

> Skyvern v1.0.53 adds Parquet workflow exports and extraction-block data export, expands Copilot with web search, in-chat credential requests, and reasoning visibility on Task V3 action rows, and exposes the Skyvern 3.0 engine in the workflow editor.

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

01 **Copilot credential request and management** NEW 60

Lets Copilot ask for a login credential with a tool that opens an in-chat credential card, shows each saved credential's tested site and authenticator in Copilot account state, and lets Copilot use explicit access-link destinations.

— Groups three related Copilot credential increments with names v1.0.53

THINNER COVERAGE BELOW

02 **Skyvern 3.0 engine on validation blocks and editor** IMPROVED 50

Exposes the Skyvern 3.0 engine on validation blocks and the default engine list in the workflow editor.

— Names surfaces but not selection mechanism v1.0.53

03 **Reasoning visibility on Task V3 action rows** NEW 50

Surfaces the provider's reasoning summary on Task V3 action rows where available, and persists the model's turn reasoning on Task V3 action rows.

— Names surface twice but no usage path v1.0.53

04 **Billing error surfaced on Discover 402** NEW 50

Shows a billing error and a Go to Billing link when Discover hits a `402` response.

— Names status code and UI action clearly v1.0.53

05 **Parquet and extraction-block data exports** NEW 45

Adds schema-directed Parquet workflow exports, and adds data export to the extraction block.

— Names formats but no mechanism or UI path given v1.0.53

06 **Web search helper for Copilot and code blocks** NEW

45

Adds a web search helper for code blocks and the Copilot to support lead-discovery workflows without scraping Google or Bing.

— Purpose clear, no exact invocation shown v1.0.53

07 **Getting started track for new organizations** NEW 35

Shows new organizations a Getting started track in the cloud sidebar.

— Clear UI location, thin on content v1.0.53

08 **Keypress recording in Record Browser** NEW 25

Records keypress gestures in the Record Browser.

— Bare description with no further detail v1.0.53

WAS THIS USEFUL? 👍 👎

CrewAI

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

Open-source Python framework for building multi-agent workflows using role-based AI agent Crews and event-driven Flows.

> CrewAI's only shipment this window was internal telemetry to track checkpoint runtime and CLI usage.

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

01 **Telemetry for checkpoint runtime and CLI usage** NEW

20

CrewAI now adds telemetry to track checkpoint runtime and CLI usage.

— Only a bare description; no mechanism, data collected, or opt-out named 1.15.21

WAS THIS USEFUL? 👍 👎

AWS Strands Agents

Feedback ⓘ 1 RELEASE · 2026-09-09

NOTES ↗ RANK

Strands Agents is AWS's open-source SDK for production agents in Python and TypeScript, with a model-driven loop, tool integration, and deployment targets.

> Strands Agents Python v1.55.1 adds background task scheduling to the Agent class, ports the notebook tool to Python, and simplifies bidirectional streaming model configuration for Google and OpenAI.

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

01 **Background tasks on Agent class** NEW 45

Adds `backgroundTasks` to the Agent class, enabling agents to schedule and run background tasks.

— Names the API surface but no usage detail or example python/v1.55.1

02 **Notebook vended tool in Python SDK** NEW 35

Ports the notebook tool to Python as a vended tool, making it available in the Python SDK's tool library.

— States what changed but no mechanism or usage python/v1.55.1

03 **Simplified bidi model configs for Google and OpenAI** IMPROVED 35

Simplifies Google and OpenAI model configurations for bidirectional streaming (`bidi`) mode.

— Names the feature area but not the specific config changes python/v1.55.1

WAS THIS USEFUL? 👍 👎

CopilotKit

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

CopilotKit connects AI agents to application interfaces with shared state, tools, and generative UI components.

> CopilotKit's v1.71.0 release adds provider-level agent configuration and unifies React Native's render-tool hooks with the shared react-core hook surface.

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

01 **React Native render-tool hooks unified with react-core**

IMPROVED 45

Converges React Native render-tool hooks onto `react-core`, unifying the hook surface across React and React Native.

— Names the module but no API or migration detail v1.71.0

02 **Provider-level agent configuration** NEW 20

Adds the ability to configure agents at the provider level, per the v1.71.0 release summary; no further mechanism or config keys were specified.

— Only mentioned in summary, no mechanism or config named v1.71.0

WAS THIS USEFUL? 👍 👎

Vercel AI SDK

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

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

> Vercel AI SDK 7.0.96 adds batch cancel and list APIs for managing asynchronous AI job batches.

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

01 **Batch cancel and list APIs** NEW 30

Adds batch cancel and list APIs for managing batches of AI requests.

— No endpoint names, params, or usage shown. `ai@7.0.96`

WAS THIS USEFUL? 👍 👎

LangGenius Dify

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗

RANK

Dify is an LLM application platform for building agentic workflows and RAG pipelines through a visual workspace, deployable to cloud or self-hosted.

> Dify 1.17.1 ships dataset-scoped Knowledge Base API keys, a configurable Markdown field-name length limit, keyboard-driven canvas navigation, marketplace creator profile pages, and a mandatory staged Weaviate upgrade for self-hosted deployments.

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

01 Configurable Markdown form field-name length limit NEW

85

Adds `MARKDOWN_FORM_FIELD_NAME_MAX_LENGTH` (Docker) and `NEXT_PUBLIC_MARKDOWN_FORM_FIELD_NAME_MAX_LENGTH` (direct Web deployments) environment variables to configure the Markdown form field-name character limit, defaulting to `128`.

Raise the Markdown form field-name limit beyond 128 characters in a Docker-based self-hosted deployment to accommodate longer generated field names.

```
$ MARKDOWN_FORM_FIELD_NAME_MAX_LENGTH=512
```

Raise the Markdown form field-name limit in a direct (non-Docker) Web deployment by setting the public environment variable before building or starting the frontend.

```
$ NEXT_PUBLIC_MARKDOWN_FORM_FIELD_NAME_MAX_LENGTH=512
```

— Exact env vars, default value, and runnable config examples given 1.17.1

02 Dataset-scoped Knowledge Base API keys NEW

70

Knowledge Base API keys can now be bound to a specific knowledge base at creation time via the knowledge base API access panel, using a workspace / this-knowledge-base scope selector. A key bound to one dataset returns `403` for any other dataset.

— Names the panel, scope selector and 403 behavior but no API call shown 1.17.1

03 Staged Weaviate upgrade required for self-hosted BREAKING

68

Self-hosted deployments using the bundled Weaviate must complete a manual staged upgrade before starting 1.17.1: the bundled Weaviate moves from `1.27.0` to `1.39.2` (12 minor versions), and skipping minors is unsupported. Pulling and restarting without the staged upgrade can silently and permanently break vector search.

— Names exact versions and risk but no upgrade command steps given `1.17.1`

THINNER COVERAGE BELOW

04 **Marketplace creator profile pages** NEW 53

Adds public marketplace creator profile pages at `/marketplace/creator/<handle>`, listing a publisher's plugins, reachable from a new Creator Center entry in the sidebar help menu.

— Names the URL pattern and navigation entry, no further mechanism `1.17.1`

05 **Keyboard arrow-key canvas movement** NEW 45

Adds keyboard arrow-key movement for workflow canvas nodes, multi-node selections, comment markers, and draft comments, with Shift held for larger steps.

— Describes behavior clearly but no key bindings or config surface `1.17.1`

└──▷ **BREAKING ON UPGRADE**

! Self-hosted deployments using the bundled Weaviate must complete a manual staged upgrade before starting 1.17.1: the bundled Weaviate moves from `1.27.0` to `1.39.2` (12 minor versions) and skipping minors is unsupported — pulling and restarting without the staged upgrade can silently and permanently break vector search.

WAS THIS USEFUL? 👍 👎

Microsoft Agent Framework

Feedback ⓘ

1 RELEASE · 2026-09-10 NOTES ↗ RANK

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

> This release adds a shared vector-store layer with four new connectors, extends AG-UI/MCP conversation handling, and ships a run of breaking changes touching secret handling, file APIs, GitHub Copilot hooks, and Foundry checkpoint loading.

└──▷ **WHAT SHIPPED · 11 FEATURES** most completely described first ? what's the number?

01 Shared vector-store abstractions and connectors NEW

80

Adds shared vector-store abstractions, portable filters, and an in-memory vector store to `agent-framework-core`; an Azure AI Search implementation via `agent-framework-azure-ai-search`; Redis HASH and JSON vector stores via `agent-framework-redis`; the alpha Qdrant vector-store connector via `agent-framework-qdrant`; and the alpha PostgreSQL/pgvector connector via `agent-framework-postgres`.

— Names every package and store type but no usage snippet `python-1.18.0`

02 SecretString becomes a masked value wrapper BREAKING

75

`SecretString` is now a masked value wrapper rather than a `str` subclass, affecting credential handling across `agent-framework-anthropic`, `agent-framework-azure-ai-search`, `agent-framework-azure-cosmos`, `agent-framework-bedrock`, `agent-framework-copilotstudio`, `agent-framework-core`, `agent-framework-foundry`, `agent-framework-gemini`, `agent-framework-mem0`, and `agent-framework-openai`.

— Names all affected packages but no code migration example `python-1.18.0`

03 AG-UI message snapshot suppression NEW

70

Adds `emit_messages_snapshot` configuration to `agent-framework-ag-ui` for suppressing the terminal message snapshot.

— Exact config key named, purpose clear, no example call `python-1.18.0`

04 Ranged file reads and line-numbering on AgentFileStore

70

BREAKING

Adds ranged file reads and moves the line-numbering contract onto `AgentFileStore` (experimental), changing the existing file-read API.

— Named class and behavior change but no migration steps `python-1.18.0`

05 **Lab dependency isolation from core[all]** BREAKING 65

Lab dependency resolution is now isolated and `agent-framework-lab` is removed from `core[all]`; install it separately.

— Exact package and install action named `python-1.18.0`

06 **Opt-in workspace file hooks in GitHub Copilot integration** 65

BREAKING

Workspace file hooks in `agent-framework-github-copilot` are now opt-in; existing setups must set `enable_file_hooks` to restore prior behavior.

— Exact flag and remediation step named `python-1.18.0`

THINNER COVERAGE BELOW

07 **Restricted checkpoint deserialization in Foundry hosting** 55

BREAKING

Checkpoint deserialization in `agent-framework-foundry-hosting` is restricted by default (beta); applications must register additional checkpoint types explicitly.

— Names package and required action, no registration API shown `python-1.18.0`

08 **Tool invocation loop duration bound and stop-reason signal** 45

NEW

Adds a maximum-duration bound and stop-reason signal to the tool invocation loop.

— Describes what changed but no config surface named `python-1.18.0`

09 **Centralized path normalization for files, memory, sessions, skills** 40

BREAKING

File, memory, session, and skill path normalization is now centralized (experimental), which may change resolved paths in existing setups.

— Impact noted but no specifics on new path behavior `python-1.18.0`

10 **MCP Host-history conversion for AG-UI** NEW 25

Adds MCP Host-history conversion for persisted AG-UI conversations.

— Bare one-line description with no mechanism `python-1.18.0`

---

11 **Mixed workflow invocation keyword arguments** NEW 25

Supports mixed workflow invocation keyword arguments.

— One line with no example or parameter names `python-1.18.0`

└──▷ **BREAKING ON UPGRADE**

! Lab dependency resolution is now isolated and `agent-framework-lab` is removed from `core[all]`; install it separately.

! `SecretString` is now a masked value wrapper rather than a `str` subclass, affecting credential handling across `agent-framework-anthropic`, `agent-framework-azure-ai-search`, `agent-framework-azure-cosmos`, `agent-framework-bedrock`, `agent-framework-copilotstudio`, `agent-framework-core`, `agent-framework-foundry`, `agent-framework-gemini`, `agent-framework-mem0`, and `agent-framework-openai`.

! Ranged file reads and the line-numbering contract have moved onto `AgentFileStore` (experimental), changing the existing file-read API.

! File, memory, session, and skill path normalization is now centralized (experimental), which may change resolved paths in existing setups.

! Workspace file hooks in `agent-framework-github-copilot` are now opt-in; existing setups must set `enable_file_hooks` to restore prior behavior.

! Checkpoint deserialization in `agent-framework-foundry-hosting` is restricted by default (beta); applications must register additional checkpoint types explicitly.

WAS THIS USEFUL? 👍 👎

LangChain LangGraph

Feedback ⓘ 1 RELEASE · 2026-09-09

BLOG ↗ RANK

Low-level orchestration framework for building, managing, and deploying long-running, stateful AI agents.

> LangGraph's Managed Deep Agents (v0.7.0+) introduced Connections, a full credential-management system covering agent-owned secrets, per-caller OAuth identities from a 23-service catalog, self-registering MCP OAuth, shared deployment-wide grants, runtime credential resolution, and local-development fallback.

└──▷ WHAT SHIPPED · 7 FEATURES most completely described first ? what's the number?

01 User-owned OAuth connections from built-in catalog NEW

90

`mda connections create <slug> --oauth <catalog-service> --client-id --secret-from-env --scope` registers user-owned OAuth connections from a 23-service catalog (including `github`) without writing authorization URLs, token URLs, or refresh logic. `--allowed-scope` caps what later authorizations may request, `--authorize-url` / `--token-url` support OAuth providers outside the built-in catalog, and `mda connections catalog` lists the providers available.

Register a GitHub OAuth connection scoped to `repo` so the agent files issues under each caller's own GitHub identity rather than a shared service account.

```
$ uv run mda connections create github-issues \
  --oauth github \
  --client-id "$GITHUB_CLIENT_ID" \
  --secret-from-env GITHUB_CLIENT_SECRET \
  --scope repo
```

— Verbatim flags plus a runnable example command. launch-20260909-449aba15

02 Runtime credential resolution API NEW

85

connections.get(<slug>, {"type": "agent"}) and connections.get(<slug>, {"type": "user"}) in `managed_deepagents` resolve credentials at run time, fetching a cached OAuth token or pausing the run to request a new grant for the caller.

Read a user-owned OAuth token at run time inside a tool — triggering an authorization interrupt automatically if the caller has not yet granted access.

```python
from managed_deepagents import connections

access_token = await connections.get("github-issues", {"type": "user"})
```

— Runnable code example demonstrates the exact call. launch-20260909-449aba15

03 **Agent-owned secret storage in Connections** NEW 80

`mda connections create <slug> --secret-from-env <ENV_VAR>` stores agent-owned static secrets in a LangSmith workspace, keeping credentials out of `.env` files and build artifacts.

Store a Tavily API key as an agent-owned connection so every caller shares it without embedding it in the codebase or deployment config.

```
$ uv run mda connections create tavily-agent --secret-from-env TAVILY_API_KEY
```

— Runnable command shown with clear mechanism and storage location.

launch-20260909-449aba15

04 **OAuth connections to self-registering MCP servers** NEW 80

`mda connections create <slug> --mcp <url>` creates a user-owned OAuth connection to any MCP server that self-registers its OAuth client, requiring only the server URL. The `define_mcp` helper in `managed_deepagents` wires MCP servers with a connection credential, e.g. connections.get("linear-mcp", {"type": "user"}) passed as the `connection` field.

— Flag and helper named but no full runnable snippet. launch-20260909-449aba15

05 **Local development connection resolution** NEW 70

Under `mda dev`, agent-owned connections resolve from `MDA_DEV_<SLUG>` environment variables (hyphens uppercased to underscores) in `.env`, and user-owned connections resolve the signed-in developer as a real principal.

— Config key pattern named but no example shown. launch-20260909-449aba15

06 **Shared deployment-level OAuth connections** NEW 60

The `--authorize` flag on `mda connections create` pre-stores a single shared OAuth grant used by all callers — a fourth ownership mode (deployment-level shared account) alongside agent-owned and per-caller user-owned connections.

— Exact flag named but no runnable example given. launch-20260909-449aba15

THINNER COVERAGE BELOW

07 **List configured connections** NEW 45

`mda connections list` inspects all connections added to a LangSmith workspace.

— Bare command with no further mechanism described. launch-20260909-449aba15

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ Agentic Infrastructure

Perplexity API

Feedback ⓘ 1 RELEASE · SEEN 2026-09-10 NOTES ↗

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 shipped a hosted remote MCP server with OAuth 2.1 and API-key authentication, a new gateway router endpoint exposing moderation and reasoning data, and added Google's Gemini 3.8 Flash model to the Agent API at promotional pricing.

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

01 Hosted remote MCP server with OAuth and API-key auth NEW

91

Adds a hosted remote MCP server over Streamable HTTP at `https://api.perplexity.ai/mcp` requiring no installation, working with any client that supports remote MCP servers (Claude Code, Cursor, VS Code, claude.ai). Supports OAuth 2.1 with PKCE and dynamic client registration, opening a browser sign-in on first use with billing organization selectable at sign-in, or API-key authentication via `--header "Authorization: Bearer YOUR_API_KEY"` for clients without OAuth support or server-side setups such as the Anthropic API MCP connector. Run `/mcp` inside Claude Code to connect and follow the sign-in prompt, with optional API-key override via the bearer-token header.

Connect to the Perplexity MCP server from Claude Code using an API key instead of OAuth.

```
$ /mcp --header "Authorization: Bearer YOUR_API_KEY"
```

Connect Cursor to the Perplexity MCP server via OAuth by navigating to the Tools & MCP settings.

> 📍 1. Open Cursor Settings › Tools & MCP 2. Add the Perplexity MCP server URL 3. Click Connect next to the server entry and complete browser sign-in

Use the Perplexity remote MCP server with OAuth in an MCP client without managing API keys — just point the client at the server URL and sign in.

> 📍 Add the remote MCP server URL to your MCP client (e.g. Claude Code, Cursor, VS Code) and sign in with your Perplexity account when prompted to authenticate

via OAuth.

Connect an OAuth-capable MCP client to the Perplexity MCP Server without managing API keys — billing is tied to the org you select at sign-in.

📍 Add `https://api.perplexity.ai/mcp` as the MCP server URL in your client (claude.ai, Claude Code, Cursor, or VS Code), then follow the OAuth prompt to sign in with your Perplexity account and select a billing organization.

— Names protocol, server URL, auth flows and a runnable connect command `snapshot-20260910`

02 **Gateway router endpoint with moderation and reasoning fields**

NEW 90

Adds a `POST /router/v1/chat/completions` gateway endpoint for chat completions, whose responses include a `moderation` object with `input` and `output` moderation results — `flagged`, `categories`, `category_scores`, and `category_applied_input_types` — for both prompt and completion sides. Also adds a `reasoning_content` field on each choice's message object surfacing chain-of-thought/scratchpad reasoning, and a `cache_write_tokens` field under `usage.prompt_tokens_details` for tracking tokens written to the prompt cache.

Check whether the model flagged your prompt or its own output for policy violations using the new moderation fields in the gateway endpoint response.

```
$ curl --request POST --url
https://api.perplexity.ai/router/v1/chat/completions --header
'Authorization: Bearer <token>' --header 'Content-Type:
application/json' --data '{"model": "sonar", "messages":
[{"role": "user", "content": "<prompt>"}]}' | jq '{input_flagged:
.moderation.input.moderation_results[0].flagged, output_flagged:
.moderation.output.moderation_results[0].flagged, categories:
.moderation.output.moderation_results[0].categories}'
```

Extract chain-of-thought reasoning alongside the final answer to audit model decision-making in security analysis workflows.

```
$ curl --request POST --url
https://api.perplexity.ai/router/v1/chat/completions --header
'Authorization: Bearer <token>' --header 'Content-Type:
application/json' --data '{"model": "sonar-reasoning",
"messages": [{"role": "user", "content": "<prompt>"}]}' | jq
'{reasoning: .choices[0].message.reasoning_content, answer:
.choices[0].message.content}'
```

— Endpoint path and every response field named with runnable curl+jq examples

product docs

---

03 **Gemini 3.8 Flash model in Agent API** NEW 65

Adds `google/gemini-3.8-flash` to the Agent API, priced at $0.75/M uncached-input tokens, $0.075/M cached-input tokens, and $3.75/M output and reasoning tokens through December 31, 2026.

Call `google/gemini-3.8-flash` via the Perplexity API for fast, cost-efficient inference with promotional pricing through December 31, 2026.

```
$ 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 OWASP Top 10."}]}'
```

Use the cost-efficient Gemini 3.8 Flash model for Agent API calls during its promotional pricing window.

```
$ 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 intel feeds."}]}'
```

— Model name and pricing given, but no mechanism beyond model swap snapshot-20260910

WAS THIS USEFUL? 👍 👎

Microsoft ONNX Runtime

Feedback ⓘ 1 RELEASE · 2026-09-10

NOTES ↗ RANK

ONNX Runtime runs machine-learning models across hardware and operating systems with an open-source inference engine.

> ONNX Runtime v1.29.1 extends attention-related operators with bidirectional GroupQueryAttention, a new BNHS KV-cache layout, and CPU sliding-window attention_bias support, alongside a performance-test tooling update that removes redundant memory copies.

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

01 **Plugin EP allocators in perf test tool** IMPROVED 80

Updates `onnxruntime_perf_test` to use plugin Execution Provider device allocators for generated inputs, loaded test data, and pre-allocated outputs, eliminating unnecessary per-run host/device copies.

— Names the tool and exact behavior change, directly usable in benchmarking. v1.29.1

02 **BNHS Value KV-cache layout support** NEW 70

Adds a session option and Execution Provider metadata contract for using the BNHS Value KV-cache layout, with graph transformations preserving compatibility with the existing BNSH operator schema.

— Mechanism and compatibility scope described, no exact config key given. v1.29.1

03 **CPU attention_bias with sliding-window cache** NEW 70

Adds CPU support for `attention_bias` with a sliding-window KV cache, including explicit position IDs and post-eviction bias indexing.

— Names the exact operator feature and mechanism but no runnable snippet. v1.29.1

04 **Bidirectional GroupQueryAttention support** NEW 60

Adds bidirectional GroupQueryAttention support on CPU and CUDA via a backward-compatible `causal` attribute.

— Names the attribute and providers but no usage example. v1.29.1

WAS THIS USEFUL? 👍 👎

Hugging Face TRL

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗

RANK

TRL trains transformer language models with reinforcement learning, covering supervised fine-tuning, reward modeling, PPO, DPO, and GRPO.

> TRL v1.13.0 adds support for training on million-token contexts and vendors fused linear losses into a new `trl.losses` module, while removing the legacy PPO trainer stack and raising several minimum dependency versions.

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

01 **1M-token long-context training for Qwen3-8B** NEW 93

Adds a long-context training guide and runnable example demonstrating 1,048,576-token training of Qwen3-8B on a single 8×H100 node, using `loss_type="chunked_nll"`, gradient-checkpointing `offload`, and YaRN RoPE levers; requires `transformers >= 5.16`.

Train a model on million-token sequences using chunked NLL loss and gradient-checkpointing offload to fit a single 8xH100 node.

```
$ trl sft \
  --model_name_or_path Qwen/Qwen3-8B \
  --dataset_name <your-long-context-dataset> \
  --loss_type chunked_nll \
  --per_device_train_batch_size 1 \
  --output_dir Qwen3-8B-1M-SFT
```

— Names exact loss type, config levers, dependency and includes runnable command. v1.13.0

02 **PPO trainer stack removed** BREAKING 60

`PPOTrainer`, `PPOConfig`, `PreTrainedModelWrapper`, `AutoModelForCausalLMWithValueHead`, and `AutoModelForSeq2SeqLMWithValueHead` are removed from TRL; `from trl import PPOTrainer` now fails at import time.

— Names removed classes exactly but gives no migration path. v1.13.0

THINNER COVERAGE BELOW

03 **Minimum dependency versions raised** BREAKING 58

TRL now requires `peft >= 0.13.0` (breaking installs on peft 0.12.x) and `deepspeed >= 0.18.6` (breaking installs on deepspeed 0.16.4); vLLM 0.19.0 is no longer supported.

— Names exact version floors but no migration guidance. v1.13.0

---

04 **Fused linear losses vendored into trl.losses** NEW 35

TRL vendors fused linear losses into a new `trl.losses` module, with chunked cross-entropy reported to run 6× faster via bf16 tensor cores.

— Names module and speedup but no usage detail given. v1.13.0

---

05 **vLLM 0.28.0 support** NEW 31

Adds support for vLLM 0.28.0.

— One-line addition with no further detail. v1.13.0

└──▷ **BREAKING ON UPGRADE**

! `PPOTrainer`, `PPOConfig`, `PreTrainedModelWrapper`, `AutoModelForCausalLMWithValueHead`, and `AutoModelForSeq2SeqLMWithValueHead` are removed from TRL; `from trl import PPOTrainer` now fails at import time.

! `peft >= 0.13.0` is now required; installations using peft 0.12.x will break.

! `deepspeed >= 0.18.6` is now required; installations using deepspeed 0.16.4 will break.

! vLLM 0.19.0 is no longer supported.

WAS THIS USEFUL? 👍 👎

Eigen Labs Darkbloom

Feedback ⓘ 1 RELEASE · 2026-09-09

NOTES ↗ CODE ↗ RANK

Private Inference Network on Idle Macs

> Darkbloom's v0.9.1 release focuses on coordinator operability, adding an isolated migration mode and more accurate request-outcome accounting, alongside a mention of per-model cache outcome reporting.

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

01 **Isolated migration mode for coordinator** NEW 85

The coordinator binary gains a `--migrate-only` flag that runs database migrations in isolation without starting listeners, workers, admin seeding, or MDM clients. It requires `EIGENINFERENCE_DATABASE_URL` to be set, and can be used as a pre-deploy migration step in CI/CD pipelines, e.g. `EIGENINFERENCE_DATABASE_URL=postgres://user:pass@host/db coordinator --migrate-only`.

Run database migrations before a coordinator deployment without starting the full service — useful in a pre-deploy migration step in CI/CD pipelines.

```
$ EIGENINFERENCE_DATABASE_URL=postgres://user:pass@host/db coordinator --migrate-only
```

— Names exact flag, env var and runnable command. v0.9.1

THINNER COVERAGE BELOW

02 **Independent request-outcome recording** IMPROVED 40

Each incoming inference request outcome is now recorded independently of provider-attempt profiles, so a request that survives multiple provider refusals before completing is counted as one outcome rather than several failures.

— Explains the behavior change but no config surface or command. v0.9.1

03 **Per-model cache outcome reporting** NEW 15

Darkbloom adds per-model cache outcome reporting, per the release summary, though no further mechanism or fields are described.

— Only named in summary; no detail on mechanism or fields. v0.9.1

WAS THIS USEFUL?  

NVIDIA Model Optimizer

Feedback ⓘ 1 RELEASE · 2026-09-09

NOTES ↗ RANK

NVIDIA Model Optimizer provides quantization, pruning, distillation, and other model-compression techniques for deployment.

> NVIDIA Model Optimizer added opt-in support for the TensorRT-RTX ABI Execution Provider for ONNX calibration on Windows arm64.

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

01 TensorRT-RTX ABI Execution Provider for ONNX calibration NEW

75

New `--calibration_eps=NvTensorRtRtx` and `--trt_rtx_backend=abi` flags enable the TensorRT-RTX ABI Execution Provider for ONNX calibration on Windows arm64; this is opt-in, with the legacy backend remaining the default.

— Names exact flags and platform but lacks broader behavioural detail. 0.46.1

WAS THIS USEFUL?  

Hugging Face Transformers

Feedback ⓘ

1 RELEASE · 2026-09-09 NOTES ↗ RANK

Transformers: the model-definition framework for state-of-the-art machine learning models in text, vision, audio, and multimodal models, for both inference and training.

> Transformers v5.17.0 ships six new model architectures spanning a 780B-parameter MoE LLM, speech synthesis, multimodal encoders and multiple ASR models, alongside quantization, kernel-registration and generation-performance improvements, plus a breaking change to vision RoPE handling.

└──▷ **WHAT SHIPPED · 13 FEATURES** most completely described first ? what's the number?

01 **HYV4 mixture-of-experts LLM** NEW 75

Adds HYV4 (Hy4-Preview), a 780B-parameter mixture-of-experts LLM with Multi-head Latent Attention (`kv_lora_rank` , `kv_b_proj`), DeepSeek Sparse Attention (`index_topk` , `indexer_types`), and Independent Hyper-Connections (`hc_mult`), with a 1M-token context window.

— Names architecture internals but no runnable usage snippet v5.17.0

02 **Canary multilingual ASR model** NEW 65

Adds Canary, a multilingual ASR and speech-to-text translation model using `ParakeetEncoder` / `ParakeetEncoderConfig` and `CanaryProcessor` for task-selecting prompt prefixes.

— Names concrete classes a reader could import directly v5.17.0

03 **Vision RoPE migration to centralized utils** BREAKING 65

Custom vision models relying on attention-layer-level or model-specific RoPE grid interleaving logic must migrate to the new centralized `modeling_rope_utils.py` implementation for vision (2D/3D) rotary embeddings.

— Names exact file and migration requirement for affected code v5.17.0

04 **NeoMME multimodal encoders** NEW 60

Adds NeoMME and NeoMME-Retriever: 260M/800M multimodal-native multilingual encoders processing text and raw image patches in a single bidirectional Transformer, with NeoMME-Retriever supporting MeanMaxSim late-interaction and dense cosine-similarity retrieval.

— Concrete sizes and retrieval methods, no code example v5.17.0

05 **Fun-ASR-Nano ASR model** NEW 60

Adds Fun-ASR-Nano, an 800M-parameter ASR model covering Chinese (including 7 dialects and 26 accents), English, and Japanese with hotword customization and native punctuation output.

— Rich detail on coverage but no invocation shown v5.17.0

06 **KimiLinear hybrid attention architecture** NEW 60

Adds KimiLinear, a hybrid linear attention architecture featuring Kimi Delta Attention (KDA) with per-channel forget gates, interleaved MLA full-attention layers, and DeepSeek-V3-style MoE feed-forward blocks.

— Mechanism-heavy but no usage example provided v5.17.0

THINNER COVERAGE BELOW

07 **Standalone function registration in KernelConfig** NEW 55

Enables standalone functions (e.g., RoPE) to be registered in `KernelConfig` with optional non-inheritance of default mappings.

— Names config object and example function, no full snippet v5.17.0

08 **AutoRound low-bit quantization** IMPROVED 50

Supports 5-, 6-, and 7-bit quantization in AutoRound.

— Concrete bit-widths named but no config example given v5.17.0

09 **VibeVoice speech synthesis model** NEW 45

Adds VibeVoice, a next-token diffusion speech synthesis model for high-fidelity long-form multi-speaker audio generation such as podcasts and audiobooks.

— Describes purpose but no API details given v5.17.0

10 **Context parallelism support flag** NEW 40

Adds `supports_context_parallel` property to `PreTrainedModel` for signalling context-parallelism compatibility.

— Named property but no usage detail beyond signalling v5.17.0

11 **Paged attention no-cache error** IMPROVED 30

Raises an error when a paged attention forward pass is called with no cache, surfacing silent failures instead of failing quietly.

— States behaviour change but no config or flag named v5.17.0

12 **Hub-kernel fallback warning** IMPROVED 30

Warns once when a hub-kernel function falls back to its reference PyTorch path instead of silently running slower code.

– Behaviour described but no configuration surface named v5.17.0

13 **Reduced generation-step overhead** IMPROVED 30

Reduces per-step generation overhead by avoiding unnecessary accelerator synchronization on every decode step.

– Mechanism named briefly, no measured benchmark given v5.17.0

└──▷ **ALSO FROM THESE RELEASES**

└──▷ **BREAKING ON UPGRADE**

! Custom vision models relying on attention-layer-level or model-specific RoPE grid interleaving logic must migrate to the new centralized `modeling_rope_utils.py` implementation for vision (2D/3D) rotary embeddings.

WAS THIS USEFUL? 👍 👎

OpenAI Python SDK

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

RANK

The OpenAI Python SDK provides typed Python clients for OpenAI model APIs, streaming, and asynchronous requests.

> OpenAI Python SDK v3.11.0 adds expiration controls for service account keys.

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

01 **Expiration controls for service account keys** NEW 35

The API client adds expiration controls for service account keys.

– Names the capability but no flag, field, or mechanism given v3.11.0

WAS THIS USEFUL? 👍 👎

OpenAI TypeScript SDK

Feedback ⓘ 1 RELEASE · 2026-09-09

NOTES ↗ RANK

The OpenAI TypeScript SDK provides typed JavaScript and TypeScript clients for OpenAI APIs and streaming responses.

> OpenAI TypeScript SDK v7.13.0 adds API key expiration controls to the API surface.

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

01 **API key expiration controls** NEW 20

Adds API key expiration controls to the OpenAI API surface, allowing keys to be configured with expiration.

— No named flag, endpoint, or parameter given, only a brief description. v7.13.0

WAS THIS USEFUL? 👍 👎

Unsloth AI Unsloth

Feedback ⓘ 1 RELEASE · 2026-09-09

NOTES ↗ RANK

Unsloth runs and fine-tunes language and diffusion models locally through a Python library and interface.

> Unsloth v0.1.808-beta ships native Docker images for running the full Studio stack, adds a Vulkan backend for AMD iGPUs, expands Apple Silicon MLX support, and speeds up diffusion inference while shrinking the Python package.

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

01 **Docker images for Unsloth Studio** NEW 80

Publishes native AMD64 and ARM64 `unsloth/unsloth` Docker images, enabling ARM-native deployments. The full Studio stack can be run on an NVIDIA GPU host with `docker run -d --gpus all --ipc=host -p 8000:8000 -p 8888:8888 -e UNSLOTH_STUDIO_PASSWORD="mypassword" -e JUPYTER_PASSWORD="mypassword" -v "$PWD":/workspace/host unsloth/unsloth`, exposing Studio on port 8000 and JupyterLab on port 8888.

Run the full Unsloth Studio stack on an NVIDIA GPU host using the new Docker image, exposing Studio on port 8000 and JupyterLab on port 8888.

```
$ docker run -d --gpus all --ipc=host \
  -p 8000:8000 -p 8888:8888 \
  -e UNSLOTH_STUDIO_PASSWORD="mypassword" -e
JUPYTER_PASSWORD="mypassword" \
  -v "$PWD":/workspace/host \
  unsloth/unsloth
```

— Runnable command with exact ports and env vars given. v0.1.808-beta

THINNER COVERAGE BELOW

02 **PyTorch 2.11 default with new install extras** IMPROVED 55

Updates default PyTorch to 2.11 across supported installers, with `cu128`, `cu126`, and `cu130` extras (`torch2110`) and `xformers 0.0.35`.

— Names exact install extras and versions, no full command. v0.1.808-beta

03 **Expanded MLX support on Apple Silicon** IMPROVED 45

Supports DoRA fine-tuning and additional DPO loss types on Apple Silicon via MLX, and batched MLX generation now streams and samples each chat session independently.

— Names DoRA and DPO but no command or config shown. v0.1.808-beta

04 **Faster diffusion inference** IMPROVED 35

Accelerates diffusion inference 1.2x–1.7x for INT8/FP8 pathways across all models.

— Gives concrete speedup numbers but no usage detail. v0.1.808-beta

05 **Text-only fine-tuning for multimodal models** NEW 25

Enables more multimodal models to be fine-tuned using text-only datasets.

— One-line description with no named models or steps. v0.1.808-beta

06 **Smaller Python package size** IMPROVED 25

Reduces the Unsloth Python package size by more than 60%.

— States a number but no mechanism given. v0.1.808-beta

WAS THIS USEFUL? 👍 👎

Fireworks AI

Feedback ⓘ 2 RELEASES · 2026-09-09 NOTES ↗ RANK

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

> Fireworks AI's biggest additions this window are a training cost estimator that previews costs across Managed, Serverless and Dedicated tiers, and a coding-agent training skill that lets agents like Claude Code, Cursor and Codex plan and cost a training run before spend; it also added per-user spending limits, SSO/SCIM, a pre-deployment model evaluation workflow, FireRouter provider-key routing, and deployment tagging.

└──▷ **WHAT SHIPPED · 7 FEATURES** most completely described first ? what's the number?

01 Training cost estimator NEW

75

Adds a training cost estimator that previews the cost of a training job before it runs, using published per-token rates for Managed and Serverless deployments and allocated GPU-hour rates for Dedicated deployments.

Before submitting a fine-tuning job, check the estimated cost to avoid surprises on large datasets or GPU-intensive runs.

📍 In the Fireworks AI console, navigate to Training, configure your job parameters, then open the Training cost estimator to view the projected cost before submitting.

— Names tiers and rate mechanism plus a UI path to try it

Training skill for coding agents | Training cost estimator

02 Per-user spending limits in Fireworks Nexus NEW

70

Adds Per-User Usage Limits to set per-user spending limits on serverless inference, with account default limits, group limits, and per-user overrides managed via Fireworks Nexus.

Enforce a per-user spending cap so no single user can exhaust your serverless inference budget.

📍 In the Fireworks console, go to Fireworks Nexus › Usage Limits, set an account default spending limit, then add per-user overrides or group limits as needed.

— Concrete console navigation path given, no API/config key product docs

03 Training skill for coding agents NEW

60

Adds Agent Skills to install Fireworks training skills (research, configure, debug) for coding agents, and a Fireworks training skill for Claude Code, Cursor, Codex, and other compatible coding agents that lets you describe a training goal in plain language to plan a run, estimate its cost, and wait for approval before spend.

— Names specific agents and behavior but no install command

Training skill for coding agents

THINNER COVERAGE BELOW

04 **Bring-your-own provider keys via FireRouter** NEW 55

Adds Provider Keys in Fireworks Nexus so requests can be routed through FireRouter to a third-party provider using your own API key instead of Fireworks credentials.

Route requests through FireRouter to a third-party provider using your own API key instead of Fireworks credentials.

📍 In the Fireworks console, go to Fireworks Nexus › Provider Keys and add your provider API key so FireRouter can call that provider on your behalf.

— Console path given but no bullet-level detail product docs

05 **Deployment tags for dedicated deployments** NEW 55

Adds Deployment Tags on dedicated deployments to attach key-value metadata, such as `team=security` and `env=prod`, for cost attribution and filtering.

Tag a dedicated deployment with environment or team metadata for cost attribution and filtering.

📍 In the Fireworks console, open your dedicated deployment, navigate to Deployment Tags, and attach key-value metadata (e.g. `team=security`, `env=prod`).

— UI path and example key-values given, no API endpoint product docs

06 **Custom SSO and SCIM provisioning** NEW 35

Adds Custom SSO support for Single Sign-On authentication and SCIM user and group provisioning.

— Named feature but no setup steps provided product docs

07 **Evaluate trained models before deployment** NEW 35

Adds Evaluating Trained Models workflow to evaluate a fine-tuned model before creating a production deployment.

— Named workflow but no steps or endpoint given product docs

WAS THIS USEFUL? 👍 👎

Ollama

Feedback ⓘ 1 RELEASE · 2026-09-05 NOTES ↗ CODE ↗ RANK

Local large language model runner that downloads, serves, and chats with models through a CLI and REST API.

> Ollama v0.34.0 adds ChatGPT Desktop integration on macOS, OpenAI-compatible tool search and response compaction endpoints, and faster structured output on Apple Silicon via speculative decoding.

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

01 **ChatGPT Desktop model selection integration** NEW 60

Adds Ollama model selection to ChatGPT Desktop on macOS, accessible via the Ollama app, letting users run open models while keeping their existing chats, plugins, MCPs, and skills.

— UI path given but no config or API detail v0.34.0

02 **OpenAI-compatible tool search support** NEW 60

Adds OpenAI-compatible client tool search support, using the `tsc_` prefix for tool search call IDs in both streaming and non-streaming responses.

— Names ID prefix but no endpoint or command shown v0.34.0

03 **Response compaction endpoint** NEW 60

Adds an OpenAI-compatible response compaction endpoint, with retry on context overflow and image preservation through compaction.

— Describes behavior but no exact route or flag named v0.34.0

04 **Faster structured output on Apple Silicon** IMPROVED 60

Improves structured output performance on Apple Silicon by enabling speculative decoding under structured output via xgrammar structural tags.

— Explains mechanism but no flag or benchmark given v0.34.0

WAS THIS USEFUL? 👍 👎

Intel AutoRound

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗

RANK

AutoRound quantizes language and multimodal models through weight-rounding optimization and exports them for inference engines.

> AutoRound v0.15.1 makes a small internal optimization to its quantization orchestrator, letting callers skip redundant reference-output collection.

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

01 reference_output parameter in quantize_block IMPROVED

75

The orchestrator's `quantize_block` function now accepts a `reference_output` parameter, allowing callers to supply pre-collected reference outputs and skip the `collect_reference` step, e.g. autoround.quantize_block(block, reference_output=cached_ref_output) to avoid re-running reference collection when quantizing multiple blocks with shared activations.

Pass previously collected reference outputs into `quantize_block` to avoid re-running `collect_reference` when quantizing multiple blocks with shared activations.

```python
autoround.quantize_block(block,
reference_output=cached_ref_output)
```

— Names exact parameter and shows runnable code, but scope is narrow v0.15.1

WAS THIS USEFUL? 👍 👎

Axolotl AI Axolotl

Feedback ⓘ 1 RELEASE · 2026-09-10

NOTES ↗ RANK

Axolotl fine-tunes open language models from YAML configuration, supporting full fine-tuning, LoRA, QLoRA, and multi-GPU training recipes.

> Axolotl v0.19.0 adds ternary quantization-aware training for BitNet-style models, new optimizers and backend selection for FlashAttention/FP8/Liger, GRPO entropy controls, S3-compatible dataset loading, and fine-tuning support for seven new model families including the 176.94B Qwen3.8-Flash-Next.

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

01 **Ternary QAT for BitNet b1.58** NEW 90

Adds `qat: weight_dtype: ternary` to enable BitNet b1.58 ternary quantization-aware training with weights restricted to {-1, 0, 1}, requiring no extra dependency, with a two-stage recipe provided in `examples/neutrino/`.

Run ternary QAT to produce a BitNet b1.58 checkpoint with no extra dependencies, using the two-stage Neutrino recipe as a starting point.

yaml

```yaml
qat:
  weight_dtype: ternary
```

— Exact config key, allowed values, and example path given. v0.19.0

02 **Qwen3.8-Flash-Next fine-tuning support** NEW 85

Adds fine-tuning support for Qwen3.8-Flash-Next (176.94B multimodal MoE) with QLoRA, vision QLoRA, and NVFP4 MoE-LoRA configs in `examples/qwen3.8-flash-next/`, plus `ple_cpu_offload: true` for single-B300 use.

— Model size, config variants, and flag with example path named. v0.19.0

03 **S3-compatible dataset loading via AWS_ENDPOINT_URL** NEW 81

Adds support for loading datasets from S3-compatible stores (MinIO, Tigris, Cloudflare R2) by setting the `AWS_ENDPOINT_URL` environment variable while keeping the existing `s3://` path format.

Load a private dataset from a Cloudflare R2 bucket without changing the s3:// path format already in your config.

```
$ export
AWS_ENDPOINT_URL=https://<account>.r2.cloudflarestorage.com
```

— Named env var, path format, and a runnable shell example. v0.19.0

04 **TorchAO FP8 scaling recipe selection** NEW 80

Adds `fp8_config.recipe` to select TorchAO FP8 scaling recipes: `tensorwise` (default), `rowwise`, or `rowwise_with_gw_hp`; incompatible `fp8_enable_fsdp_float8_all_gather` plus rowwise combinations are rejected at validation.

— Config key, all recipe values, and validation rule named exhaustively. v0.19.0

05 **GRPO entropy and importance-sampling controls** NEW 78

Surfaces GRPO entropy regularization and importance-sampling clipping knobs in `cfg.trl`: `entropy_coef`, `use_adaptive_entropy`, `entropy_target`, `entropy_coef_delta` / `min` / `max`, `vllm_importance_sampling_clip_min` / `max`, and `log_multimodal`; `scale_rewards` now accepts `group`, `batch`, or `none` in addition to the previous boolean.

— All new config fields named exhaustively, no usage example provided. v0.19.0

06 **PoLoRA and AdamC optimizers** NEW 75

Adds `optimizer: polora` (Muon-like LoRA optimizer) and `optimizer: adamc` (Adam with learning-rate-scaled weight decay), both usable with LoRA or QLoRA on single GPU, DDP, or FSDP2.

Use the PoLoRA optimizer for LoRA fine-tuning on a single GPU to get Muon-style orthogonal updates on adapter weights.

```yaml
optimizer: polora
```

— Both config values and supported training modes named, example runnable. v0.19.0

07 **Liger kernel backend selection** IMPROVED 71

Adds `liger_kernel_impl: cutile` or `cutedsl` to select Liger 0.8.1 Blackwell-tuned alternative kernel backends; `ipo` under `dpo_use_liger_kernel` now fails at config time rather than mid-run.

— Config keys and Liger version named, no runnable example given. v0.19.0

08 **Explicit FlashAttention-4 and FA-Torch backends** NEW — 68

Adds `flash_attention_4` and `flash_attention_torch` as explicit `attn_implementation` values, both supporting varlen sample packing; `flash_attention_torch` validates against the installed transformers version at preflight.

— Config values and validation behavior named, no usage example shown. v0.19.0

THINNER COVERAGE BELOW

09 **Declarative model-support profiles** NEW — 55

Adds declarative model-support profiles (`VANILLA_CAUSAL_LM` , `IMAGE_TEXT_TO_TEXT`) so new architectures declare only their differences from a family template, with no config-file changes required.

— Explains mechanism but has no user-facing config surface. v0.19.0

10 **Chat-template tokenization speedup** IMPROVED — 50

Delivers a 16.4x chat-template tokenization speedup for long multi-turn conversations (80-turn Qwen2.5: 1577.6 ms → 96.4 ms).

— Concrete benchmark numbers but nothing for reader to act on. v0.19.0

11 **Support for six new model families** NEW — 45

Adds fine-tuning support for six new model families: Ling 3.0 Flash and Tiny, Muse Glimmer 30B, Cohere North Micro Vision Instruct, Shieldstral, Qwen3.8 27B, and Nemotron 3.5 Lightning 30B-A3B, each with example configs.

— Names models but gives no config keys or mechanism. v0.19.0

12 **Raised PyTorch version ceiling** IMPROVED — 45

Raises the PyTorch ceiling to 2.13.0 with GPU container builds provided for 2.12.1 and 2.13.0.

— Named versions but minimal further detail on impact. v0.19.0

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

llmfit

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗ CODE ↗ RANK

Hundreds of models & providers. One command to find what runs on your hardware.

> llmfit v1.1.15 adds Ferrum as a named benchmark provider and expands its community benchmark dataset with several new hardware profiles.

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

01 **Ferrum provider support in bench** NEW 80

Adds `--provider ferrum` to `llmfit bench` and the `FERRUM_HOST` environment variable, enabling benchmarking against Ferrum endpoints using the existing OpenAI-compatible transport.

Benchmark a model running on a Ferrum endpoint to capture real tok/s and TTFT measurements on your hardware.

```
$ llmfit bench --provider ferrum
```

— Names exact flag, env var, and runnable command. v1.1.15

THINNER COVERAGE BELOW

02 **New community benchmark hardware profiles** NEW 45

Adds community benchmark results for `nvidia-gb10`, `nvidia-geforce-rtx-5070-ti`, `nvidia-geforce-rtx-3080`, `intel-raptor-lake-p-iris-xe-graphics-integrated`, and `intel-coffeelake-s-gt2-uhd-graphics-630-integrated` hardware profiles.

— Names specific hardware profiles but no mechanism or usage. v1.1.15

WAS THIS USEFUL? 👍 👎

Comfy Org ComfyUI

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

RANK

ComfyUI is a node-graph interface, API, and backend for running diffusion models, chaining generation steps into reusable visual workflows.

> ComfyUI v0.35.0 introduces a new Comfy Compiler for graph execution optimization, adds 3D and video editing nodes, AVIF image export, a sparse attention node, and a broad wave of new partner model integrations and LoRA support.

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

01 **New partner model integrations across partner nodes** NEW 60

Adds MiniMax-H3 Max and Max Turbo model options to the H3 text-to-video and first-last-frame partner nodes; Recraft V4 Styles and V4 Styles Pro to the Create Style partner node; WAN3-Prime via the WAN partner node; Google Omni 1.1 support; SenseNova U1.5 support (CORE-411); Pixal3D multiview support (CORE-421); Meta Muse Image partner nodes; Claude Fable 5.1 to the Anthropic Claude partner node; GPT Image 2.5 Flare and Sunburst models to the OpenAI partner node; the GPT-6 Astra model and a `reasoning` parameter to the OpenAI partner node; and Comfy-Cloud partner nodes to run curated workflows on Comfy Cloud GPUs.

— Names every partner node and model added, each thinly described v0.35.0

THINNER COVERAGE BELOW

02 **3D file import node** NEW 55

Adds the `File3DToMesh` node to parse GLB, GLTF, OBJ, and STL files into a MESH type for use in 3D workflows.

— Names node and supported formats, no example given v0.35.0

03 **In-graph video trim and crop nodes** NEW 55

Adds `VideoTrim` and `VideoCrop` nodes with `VIDEO_EDIT` widget inputs for in-graph video editing.

— Names nodes and widget type, no usage example v0.35.0

04 **LTXV video generation nodes** NEW 55

Adds LTXV generated-keyframe nodes and a Freeze Latent node, plus `LTXVAddLatentGuide` for pinning a pre-encoded latent as a guide frame.

— Names three nodes with brief mechanism for one v0.35.0

05 **AMD GPU attention and FP8 support** NEW — 55

Enables PyTorch SDPA attention and FP8 ops for AMD gfx1170 and gfx1171 GPUs.

— Names GPU archs and ops but no config to enable v0.35.0

06 **AVIF export in Save Image Advanced** IMPROVED — 50

Supports the `avif` format in the Save Image Advanced node.

— Names node and format, no limits or defaults given v0.35.0

07 **Live resolution preview in ResolutionSelector** IMPROVED — 45

Adds a live resolution preview widget to the `ResolutionSelector` node.

— Names node and widget, clear UI location v0.35.0

08 **LoRA support for MiniMax-H3 and HiDream O1** NEW — 45

Supports PDD LoRA for MiniMax-H3, and supports MiniMax-H3 and HiDream O1 LoRAs trained with DiffSynth-Studio / ModelScope.

— Names models and training frameworks, no usage steps v0.35.0

09 **Container memory limit respected** IMPROVED — 45

Now respects the container cgroup memory limit instead of host RAM when running in containers.

— States before/after behaviour, no config key named v0.35.0

10 **Comfy Compiler for graph execution** NEW — 40

Introduces the Comfy Compiler (tracked as CORE-389) for graph execution optimization.

— Mechanism named but no detail on how it optimizes v0.35.0

11 **ImageCompositor emits document canvas** IMPROVED — 40

Emits the document canvas from `ImageCompositor` for use in the layer editor.

— Names node and consumer, no further mechanism v0.35.0

12 **cfgpp_ud10_ab sampler added** NEW — 35

Adds the `cfgpp_ud10_ab` sampler.

— Bare name with no mechanism explained v0.35.0

13 **Sparse Attention node** NEW 30

Adds a Sparse Attention node.

— Bare name, no explanation of behaviour v0.35.0

14 **Color space conversion node** NEW 25

Adds a color space conversion node.

— No node name or supported color spaces given v0.35.0

15 **Save 3D Advanced output reporting** IMPROVED 25

Reports the file saved by Save 3D (Advanced) as a standard 3D output item.

— Bare behavioural change, no further detail v0.35.0

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

Pinecone

Feedback ⓘ 2 RELEASES · SEEN 2026-09-09 → 2026-09-10 NOTES ↗

RANK

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

> Pinecone shipped a new Nexus retrieval layer with BYOC and custom context manifests, a broad set of organization access-control and resource-management APIs, and took full-text search (BM25 keyword ranking) generally available inside the vector index.

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

01 **Service account and access control APIs** NEW 93

Adds `POST /service-accounts` to create a service account with optional initial role bindings (client secret returned only once), `GET /service-accounts/{id}` and `GET /service-accounts` to retrieve or list service accounts, `PATCH /service-accounts/{id}` to rename one, `DELETE /service-accounts/{id}` to delete one and revoke all its tokens within seconds, a rotate-secret endpoint to rotate a service account's OAuth client secret (previous secret and tokens revoked within seconds, new secret returned once), `POST /role-bindings` to grant a role to a principal at organization or project scope, and `GET /invites` to list pending and expired invites in the caller's organization.

Rotate a compromised service account OAuth secret immediately to revoke all outstanding tokens within seconds.

```
$ curl -X POST https://api.pinecone.io/service-accounts/<service-account-id>/rotate-secret \
  -H 'Authorization: Bearer <your-token>'
```

— Every endpoint named with revocation/return-once behaviour and a runnable curl example

snapshot-20260910

02 **Full-text search generally available** NEW 86

Pinecone's full-text search (BM25 keyword ranking) is now generally available via the Documents API, running inside the same Pinecone Database index as dense and sparse vectors with no separate search cluster required. It supports Lucene query syntax including boolean operators, phrase queries, and fuzzy matching with the `~` operator for typo tolerance across multiple text fields per index, plus text-match filters that restrict semantic (dense vector) search results to documents matching a literal keyword

filter so one query combines identifier lookup with meaning-based ranking. Tokenization and stemming cover 18 languages plus language-agnostic n-gram tokenization for substring- and prefix-matching, and the feature supports Dedicated Read Nodes for provisioned read capacity and Bring Your Own Cloud (BYOC) deployment, metered in the same read units and write units as vector indexes with no cluster sizing, sharding, or upgrade maintenance.

— Rich mechanism and named capabilities but no runnable command given

launch-20260909-87279580

03 **Inference and Admin API specs versioned 2026-07** NEW 70

Adds `inference_2026-07.oas` and `admin_2026-07.oas` OpenAPI specs, versioned `2026-07`, for the Inference and Admin APIs, downloadable directly from the pinecone-api GitHub repository.

Fetch the latest 2026-07 Inference API spec to generate a client or validate requests against the new versioned contract.

```
$ curl -O https://raw.githubusercontent.com/pinecone-io/pinecone-api/refs/heads/main/2026-07/inference_2026-07.oas.yaml
```

Fetch the latest 2026-07 Admin API spec to inspect new admin endpoints or integrate them into an API gateway.

```
$ curl -O https://raw.githubusercontent.com/pinecone-io/pinecone-api/refs/heads/main/2026-07/admin_2026-07.oas.yaml
```

— Exact file names and version plus runnable curl download commands product docs

04 **Nexus retrieval layer with context manifests** NEW 60

Pinecone launched Nexus, a curated retrieval layer with Bring Your Own Cloud (BYOC) support, and added a context manifest system letting users define custom artifact and edge types in the console, turning sources into queryable knowledge. Manifests are built via Nexus > Context Design in the console by selecting Design your own manifest, defining artifact and edge types, then triggering curation to index sources.

Design a Nexus context manifest with custom artifact and edge types to make your sources queryable in Pinecone Nexus.

📍 In the Pinecone console, navigate to Nexus › Context Design, select Design your own manifest, define custom artifact types and edge types, then trigger curation to index your sources.

— Describes mechanism and console path but no API or config surface `product docs`

THINNER COVERAGE BELOW

05 **Resource management APIs for projects, organizations, API keys**

NEW

55

Adds project management APIs to list projects, create a project, get project details, and update a project's name, maximum Pod count, or CMEK, plus delete a project; organization management APIs to list organizations, get organization details, update an organization's name, and delete an organization; and API key management APIs to list all API keys in a project and delete an API key from a project.

— Names fields and resources but no exact endpoint paths given `snapshot-20260910`

06 **Documentation index at /llms.txt** NEW

40

Serves a complete documentation index at `/llms.txt` for programmatic discovery of all available pages.

— Names exact path but no further mechanism or example `snapshot-20260910`

WAS THIS USEFUL? 👍 👎

timgordontg engrim

Feedback ⓘ 2 RELEASES · 2026-09-10

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 shipped a new `engrim merge` subcommand for combining SQLite memory stores across machines and CI, alongside `engrim prune` for log retention and disk reclamation, tag-based filtering, and a `--strict` exit-code gate for agent review hooks.

└──▷ **WHAT SHIPPED · 7 FEATURES** most completely described first ? what's the number?

01 **Prune command for log retention and vacuum** NEW 95

New `engrim prune` command purges stale transcript logs and reclaims SQLite disk space via `VACUUM`. It requires `--keep-days <N>` or `$ENGRIM_PRUNE_KEEP_DAYS` to delete anything, or `--vacuum` alone can reclaim fragmented space without deleting logs.

Reclaim disk space on a schedule without risking log deletion — vacuum only, no retention window needed.

```
$ engrim prune --vacuum
```

Purge transcript logs older than 30 days and vacuum the SQLite store, driven by an env var so CI or cron jobs need no flag changes.

```
$ ENGRIM_PRUNE_KEEP_DAYS=30 engrim prune
```

— Flags, env var and runnable examples all given v1.3.1

02 **Merge SQLite memory stores** NEW 85

New `engrim merge OTHER.db` subcommand folds another SQLite store's records and logs into the current store. It uses idempotent monotonic status resolution so `superseded` / `done` records carry over and are never reactivated, plus content-hash deduplication, enabling parallel CI and multi-machine agent workflows.

— Names subcommand and merge mechanism, no usage example given v1.3.2

03 **Strict exit-code gating for review** NEW 85

Adds `--strict` flag to `engrim review` (and `$ENGRIM_STRICT=1` env var) that exits with code 2 when uncommitted architectural decisions are detected, enabling stop-hook gating for agents and orchestrators.

Gate an agent stop hook so it exits with code 2 — blocking the agent from clearing context — if any uncommitted architectural decisions are detected.

```
$ engrim review --strict
```

— Exact flag, env var and runnable example provided v1.3.1

04 **Tag filtering for recall and list** NEW 65

Adds `--tag` / `--tags` flags to `engrim recall` and `engrim list` for filtering memories by tag, with full support across the CLI and the stdio MCP server tools for Claude Code and Cursor.

— Flags and surfaces named, no runnable example v1.3.1

THINNER COVERAGE BELOW

05 **Negation-aware decision capture** IMPROVED 35

Capture checks now include explicit negation handling so reversed decisions (e.g. 'decided not to use X') no longer falsely match existing positive decisions.

— Describes behavior change but no command or flag v1.3.2

06 **Fixed review status on fresh environments** IMPROVED 30

In a fresh, unlogged environment, review status is now reported as unknown rather than a false-positive safe to clear when no session logs exist.

— Brief bug-fix description, no surface named v1.3.2

07 **Version flag for CLI** NEW 15

Adds `-v` / `--version` flag to the `engrim` CLI parser.

— Bare flag name with no further detail v1.3.1

WAS THIS USEFUL? 👍 👎

InfiniFlow RAGFlow

Feedback ⓘ 1 RELEASE · 2026-09-10

NOTES ↗ RANK

RAGFlow is a retrieval-augmented generation engine built on deep document understanding, combining chunk-level citations with agentic workflows.

> RAGFlow v0.27.2 broadens data ingestion and parsing options, adds new model providers and an agent search tool, refactors the Agentic RAG retrieval framework for speed, and rounds out knowledge-graph, document-preview and search UI details.

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

01 **RERANK_TOKEN_LIMIT_MODE config key** NEW — 65

Adds the `RERANK_TOKEN_LIMIT_MODE` configuration key to control token limit behaviour in the rerank pipeline.

— Named config key with clear purpose but no values documented v0.27.2

THINNER COVERAGE BELOW

02 **Sitemap and WebDAV connector additions** NEW — 50

Adds a Sitemap data source for `sitemap.xml` `-based` web ingestion, and adds WebDAV connector support for custom CA certificates.

— Names both surfaces but lacks configuration detail v0.27.2

03 **New document parsing backends** NEW — 45

Adds MonkeyOCRv2 as a supported document parsing backend and adds support for self-hosted PaddleOCR-VL deployments.

— Names two backends but no setup steps given v0.27.2

04 **Document preview enhancements** NEW — 40

Adds EPUB document preview and adds Excel citation source location in document preview.

— Names two formats but no further mechanism v0.27.2

05 **Search and highlighting UI improvements** IMPROVED — 40

Adds search match highlighting in Team settings tables, highlights results in retrieval testing and search, and displays similarity threshold as a percentage.

— Groups three UI tweaks with no deeper mechanism v0.27.2

06 **Refactored Agentic RAG retrieval framework** IMPROVED

35

The Agentic RAG retrieval framework was refactored, significantly improving reasoning speed and benchmark performance.

— Names benefit but no mechanism or numbers given v0.27.2

07 **Knowledge graph UI enhancements** IMPROVED 35

Adds knowledge graph display of total and current node counts, and highlights the node in graph search when pressing Enter on an entity name.

— Describes UI behaviour but no config or API surface v0.27.2

08 **New model providers: Hubris and llmman** NEW 30

Adds Hubris and llmman as new model providers.

— Named providers but no integration detail v0.27.2

09 **Knowledge compilation runtime settings and rate limiting** IMPROVED 30

Adds knowledge compilation runtime settings and enforces a rate limit on the knowledge compilation pipeline.

— No specific setting names or limit values given v0.27.2

10 **Sofya search tool for agents** NEW 25

Adds a Sofya search tool for use in agents.

— Bare name, no usage detail v0.27.2

WAS THIS USEFUL?  

EVALUATE

◆ AI Observability & Evals

PromptLayer

Feedback ⓘ 1 RELEASE · 2026-09-10 CHANGELOG ↗ RANK

PromptLayer is a platform for managing, evaluating, and observing prompts and LLM applications.

> PromptLayer added a second keyboard shortcut for filters and expanded multimodal support for Gemini 3.8 Flash workflows with a new fullscreen video view.

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

01 **Multimodal inputs for Gemini 3.8 Flash workflows** NEW

50

Supports multimodal inputs — text, images, video, audio, and PDF files — for Gemini 3.8 Flash model workflows. Also adds a fullscreen view for video content playback in conversation display.

— Names input types and model but no config/API detail changelog-20260910-eacb230b

02 **Ctrl+Enter shortcut to apply filters** IMPROVED

40

Adds `Ctrl+Enter` as an additional keyboard shortcut to apply filters, alongside the existing `Cmd+Enter`.

— Exact shortcut named but minor scope changelog-20260910-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 biggest change this window is deprecating its legacy feedback-formula and dataset-comparison APIs in favor of a new paginated experiment-runs endpoint and composite evaluators, alongside new tooling for testing multi-turn evaluators and managing dataset splits in bulk.

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

01 Test multi-turn evaluators against real conversations NEW

85

The `POST /runs/rules/validate` endpoint now accepts `test_thread_id` and `session_id` parameters, letting you validate a thread evaluator against a real conversation before saving the rule. A matching Test action was added to multi-turn evaluators in the evaluator config UI, running the evaluator against a sample thread before it's saved.

Validate a multi-turn thread evaluator against a real conversation before saving the rule, to confirm it scores as expected.

```
$ curl -X POST
'https://api.smith.langchain.com/runs/rules/validate' \
  -H 'Content-Type: application/json' \
  -H 'x-api-key: <api_key>' \
  -d '{"rule": {"evaluator_type": "thread"}, "test_thread_id": "
<thread_id>", "session_id": "<session_id>"}'
```

— Names endpoint, params, and includes a runnable curl example snapshot-20260910

02 Experiment comparison API replaced with paginated endpoint

70

BREAKING

`POST /v2/datasets/<id>/experiment-runs` is now the supported public API for paginated experiment comparison. Legacy dataset comparison helpers have been removed from the public OpenAPI spec and generated SDKs, though the underlying HTTP routes continue to work only for LangSmith UI clients.

— Names replacement endpoint but no usage example snapshot-20260910

THINNER COVERAGE BELOW

03 Mixed thread and run items in annotation queues NEW

50

The annotation queue items API supports adding a batch of conversation threads and runs to a queue in a single request, mixing `item_type: THREAD` (with `thread_id` / `session_id`) and `item_type: RUN` (with `run_id`) entries via `POST /annotation-queues/<queue_id>/items`.

Add a batch of conversation threads to an annotation queue for human review, mixing thread and run items in a single request.

```
$ curl -X POST 'https://api.smith.langchain.com/annotation-queues/<queue_id>/items' \
  -H 'Content-Type: application/json' \
  -H 'x-api-key: <api_key>' \
  -d '{"items": [{"item_type": "THREAD", "thread_id": "<thread_id>", "session_id": "<session_id>"}, {"item_type": "RUN", "run_id": "<run_id>"}]}'
```

— Only known via example, not described in release notes `snapshot-20260910`

04 Dataset split chips in Examples table IMPROVED

38

Each example's dataset splits now render as interactive chips in the dataset Examples table, with a clickable +N overflow menu when an example belongs to more splits than fit the column.

— UI-only description, no navigation path or config given `snapshot-20260910`

05 Deferred rendering for large experiment tables IMPROVED

30

Experiment tables now defer full payload rendering and output diff preparation until those views are requested, improving responsiveness for runs with large agent trajectories.

— States the change but no metrics or trigger detail `snapshot-20260910`

└──▷ 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.

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

WAS THIS USEFUL? 👍 👎

Arize Phoenix

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

Arize Phoenix is an open-source platform that monitors LLM application traces and evaluates their outputs.

> Phoenix Evals adds a new evaluator for checking whether conversational requests were fully completed.

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

01 **Completeness evaluator for conversations** NEW 53

Adds `createCompletenessEvaluator` to judge whether every active user request in a conversation was actually completed.

— Names exact function but no usage or mechanism detail. `@arizeai/phoenix-evals@2.5.0`

WAS THIS USEFUL? 👍 👎

Langfuse

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

Langfuse provides tracing, evaluation, and monitoring for LLM applications.

> Langfuse's v4.33.0 release adds a time-scoped observation lookup to the public API, new filter operators for nullable fields, and a small playground UI improvement.

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

01 Optional startTime parameter on observation lookup NEW

80

The public API endpoint `GET /observations/{id}` now accepts an optional `startTime` query parameter, enabling time-scoped observation lookups so you can fetch a specific observation without scanning the full history when you know approximately when it occurred.

Fetch a specific observation scoped to a start time to avoid scanning the full history when you know approximately when the observation occurred.

```
$ curl -X GET
'https://cloud.langfuse.com/api/public/observations/{id}?
startTime=2025-09-01T00:00:00Z' \
  -H 'Authorization: Basic <base64-encoded-credentials>'
```

— Named endpoint, param, and runnable curl example. v4.33.0

THINNER COVERAGE BELOW

02 has: and -has: filter operators for nullable fields NEW

40

The filter UI now surfaces `has:` and `-has:` filter operators while typing a nullable field name, allowing filtering on presence or absence of a value.

— Named operators but no exact UI path or example. v4.33.0

03 Source prompt name in playground IMPROVED

20

The playground window content now displays the source prompt name.

— Very thin description, no mechanism or location detail. v4.33.0

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 changes centered on Loop, which now runs in a Braintrust-managed runtime with scheduled automations, alongside new AI-driven trace analysis (Patterns, Debugger) and Java SDK concurrency controls that require Task/Scorer/Classifier implementations to become thread-safe. Smaller additions covered new built-in models, dashboard sections, blind human reviews, and SDK-level metadata across Python, Java, and TypeScript.

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

01 Java SDK concurrent eval execution and tuning

BREAKING

95

The Java SDK now runs eval cases concurrently, so Task, Scorer, and Classifier implementations must be thread-safe. Concurrency and OTel batching are controlled via the `BRAINTRUST_DEFAULT_MAX_CONCURRENCY`, `BRAINTRUST_OTEL_MAX_QUEUE_SIZE`, `BRAINTRUST_OTEL_MAX_EXPORT_BATCH_SIZE`, and `BRAINTRUST_OTEL_EXPORT_INTERVAL_MILLIS` environment variables, or programmatically via .maxConcurrency(int) on `Eval.Builder` and `Devserver.Builder` (default: 3 at a time) and .executor(Executor).

Tune Java eval concurrency and OTel export behavior without touching code, useful in high-throughput CI environments.

```
$ export BRAINTRUST_DEFAULT_MAX_CONCURRENCY=10
export BRAINTRUST_OTEL_MAX_QUEUE_SIZE=4096
export BRAINTRUST_OTEL_MAX_EXPORT_BATCH_SIZE=512
export BRAINTRUST_OTEL_EXPORT_INTERVAL_MILLIS=2000
```

Cap concurrent eval case execution and tune OTel export batching without touching code, using environment variables.

```
$ BRAINTRUST_DEFAULT_MAX_CONCURRENCY=10
BRAINTRUST_OTEL_MAX_QUEUE_SIZE=512
BRAINTRUST_OTEL_MAX_EXPORT_BATCH_SIZE=64
BRAINTRUST_OTEL_EXPORT_INTERVAL_MILLIS=2000 your-eval-runner
```

— Runnable env vars and builder methods with an explicit default and migration requirement

snapshot-20260910

02 **GLM-5.3 Flash built-in model** NEW 65

Adds GLM-5.3 Flash as a built-in multimodal reasoning model available under the Braintrust provider in playgrounds, prompts, and scorers, or via the Braintrust Gateway using the identifier `glm-5.3-flash`.

— Exact gateway identifier makes this directly usable snapshot-20260910

03 **Eval.start() async eval execution in Java SDK** NEW 60

Adds Eval.start() to the Java SDK (`v0.3.23`) to start an eval in the background and return an `EvalResult` as soon as the experiment exists, while Eval.run() still awaits full completion.

— Names the exact method and version but no runnable example snapshot-20260910

THINNER COVERAGE BELOW

04 **Debugger and Analyze trace view** NEW 55

Adds the Debugger to the trace viewer, which reports likely failure modes for a single trace citing the spans, tool calls, and model outputs behind each one, alongside an Analyze trace view that groups a run into labeled Work sections.

— Names the UI view and what it groups, giving a starting point snapshot-20260910

05 **Blind human reviews project setting** NEW 55

Adds the Blind human reviews project setting, which hides peer scores, comments, and aggregates until a reviewer submits their own scores; reviewers with the Update permission are always exempt.

— Names the exact setting and permission exemption clearly snapshot-20260910

06 **org_name parameter for name-lookup tools** NEW 55

Adds `org_name` parameter to name-lookup tools to disambiguate patterns, prompts, and scorers across organizations.

— Names the parameter but not the tool or endpoint it modifies product docs

07 **TypeScript SDK scorer score objects without name** IMPROVED 55

TypeScript SDK (`v3.31.0`) scorer functions in the test runner can now return a score object without a `name` field, using `'score'` as the key when `name` is omitted.

— Exact field behavior and version named, easy to apply snapshot-20260910

08 **Dashboard sections for charts** NEW 55

Group a dashboard's charts into named sections that can be reordered, duplicated, and collapsed, with collapsed state persisted per browser; available on Pro and Enterprise plans.

— Describes behavior and plan availability, no exact UI path snapshot-20260910

09 **Patterns AI trace investigator** NEW 50

Patterns is an AI-driven scheduled investigation of traces that surfaces recurring problems, cost trends, and cohort-level failure modes; each finding is saved with supporting traces and a suggested fix.

— Explains mechanism but no config, endpoint, or navigation path snapshot-20260910

10 **Pydantic AI span metadata parameters** IMPROVED 50

Adds `temperature`, `max_tokens`, and similar invocation parameters to Pydantic AI LLM span metadata in the Python SDK (`v0.37.0`) when model settings are provided.

— Names fields and version but no usage example snapshot-20260910

11 **Shareable dashboard chart links** NEW 40

Adds a shareable link to individual dashboard charts that opens the chart in fullscreen mode.

— Clear UI action but thin on mechanism product docs

12 **toolCallId in Vercel AI SDK v6 spans** IMPROVED 40

Vercel AI SDK v6 tool call spans now include `toolCallId` in span metadata (TypeScript SDK `v3.31.0`).

— Names the field and SDK version, thin on further detail snapshot-20260910

└──▷ **BREAKING ON UPGRADE**

! Task and Classifier implementations must now be thread-safe due to concurrent eval case execution.

! Task, Scorer, and Classifier implementations in the Java SDK (`v0.3.23`) must now be thread-safe, as eval cases run concurrently.

WAS THIS USEFUL? 👍 👎

GOVERN

◆ AI/LLM Security

AI-Infra-Guard

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗

RANK

AI-Infra-Guard is an AI red teaming platform that scans AI infrastructure for CVE vulnerabilities, MCP/Agent skill risks, jailbreak weaknesses, and API relay security issues.

> AI-Infra-Guard's v4.6.1 release expands the API Checker's model fingerprinting coverage to include Gemini, Gemma, and GLM models.

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

01 **Gemini, Gemma, and GLM fingerprints in API Checker** IMPROVED

45

The API Checker now identifies Gemini, Gemma, and GLM AI models via fingerprints, and the GLM-5.3-Flash fingerprint baseline was updated.

— Names models added but no mechanism or usage detail v4.6.1

WAS THIS USEFUL? 👍 👎

ai-safe2-framework

Feedback ⓘ 1 RELEASE · 2026-09-10

NOTES ↗ RANK

Ai-Safe2-Framework provides governance, risk, and compliance controls for securing agentic AI systems and non-human identities.

> safe2 CLI 0.2.0 adds offline Challenge evidence bundle generation and verification, a hardened Skill Trust Gate command that replaces a retired script, and stricter AISM ingestion validation, alongside a small adapter flag and expanded CI coverage.

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

01 **Skill Trust Gate CLI command and detection rules** BREAKING

95

Adds `safe2 gate skill PATH --strict` to replace the retired standalone skill-gate script — the old script now exits with an error, so all uses must be replaced with `safe2 gate skill PATH --strict`. The new command enforces expanded file-inspection scope across scripts, extensionless files, and unfamiliar extensions, and ships new Skill Trust Gate detection rules covering injection directives, credential paths, dynamic execution, and other suspicious patterns, with coverage counts reporting what was inspected.

— Names exact command, flag, and migration path in full. 2026-09-09_safe2_CLI_0.2.0

02 **Offline Challenge 001 evidence bundle generation** NEW

90

Adds `safe2 challenge quickstart 001 --output-dir <dir>` to generate an offline Challenge 001 evidence bundle — including Decision Cards, inert scenario results, and optional Ed25519 signatures — without model credentials or paid APIs.

— Exact command and bundle contents given. 2026-09-09_safe2_CLI_0.2.0

03 **Challenge bundle verification command** NEW

90

Adds `safe2 challenge verify-bundle <dir>` to verify a Challenge bundle's internal consistency, checking file inventory, hashes, source binding, and artifact integrity; accepts `--expected-sha256` to pin the manifest against a separately obtained trusted fingerprint.

— Exact command, flag, and verification checks named. 2026-09-09_safe2_CLI_0.2.0

04 **Stricter AISM ingestion validation** BREAKING

60

AISM ingestion now rejects ambiguous JSON, unsafe paths, invalid Challenge runs, and existing output files — use a new output filename when re-running.

— Clear before/after behaviour with migration note. 2026-09-09_safe2_CLI_0.2.0

THINNER COVERAGE BELOW

05 **SkillSpector adapter --executable flag** NEW 55

Adds `--executable` flag to the SkillSpector adapter for configuring a separate provider environment.

— Flag named but mechanism thinly described. 2026-09-09_safe2_CLI_0.2.0

06 **Expanded CI Python version coverage** IMPROVED 30

Extends CI coverage to Python 3.11–3.14 (previously only Python 3.11 was validated).

— Simple internal CI change, minimal reader action. 2026-09-09_safe2_CLI_0.2.0

└──▷ **BREAKING ON UPGRADE**

! The retired standalone skill-gate script now exits with an error; replace all uses with `safe2 gate skill PATH --strict`.

! AISM ingestion now rejects ambiguous JSON, unsafe paths, invalid Challenge runs, and existing output files — use a new output filename when re-running.

WAS THIS USEFUL? 👍 👎

Doberman-Core

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

Doberman-Core is a runtime security tool that blocks unsafe or unintended AI agent inputs, outputs, and tool calls before execution.

> Doberman-Core v0.18.7 adds phone-based Approve/Deny challenges over ntfy, tightens detection against command-wrapper bypasses, and closes a fail-open gap in Claude Code hooks, alongside more auditable decision logs and tighter elevation revocation.

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

01 **Phone-based Approve/Deny challenges via ntfy** NEW 90

`doberman phone setup` pushes 2FA and elevation challenges to a phone via ntfy with Approve and Deny buttons; on 2FA tiers the tap replaces the TOTP code. `--server` points setup at a self-hosted ntfy instance instead of the public one.

Enroll phone approvals against a self-hosted ntfy server so challenge commands never reach the public ntfy instance.

```
$ doberman phone setup --server https://ntfy.example.com
```

— Names command, flag and mechanism with a runnable example. v0.18.7

02 **Timeout pin on Claude Code hook entries** BREAKING 90

Claude Code hook entries now pin a `timeout` that outlasts Doberman's challenge ceiling, preventing a timed-out hook from letting a call through before Doberman denies. Existing hook entries lack this pin and will continue to fail open on timeout until you re-run `doberman install-hooks` (add `--host cursor` for Cursor); `doberman doctor` flags un-pinned entries.

Refresh hook entries after upgrading to pick up the timeout pin that prevents fail-open on challenge timeout.

```
$ doberman install-hooks --host cursor
```

— Exact migration command and flag given, with mechanism and impact. v0.18.7

03 **Wrapper-bypass detection for destructive commands** IMPROVED 70

Destructive or egress commands wrapped in `builtin`, `eval`, `strace`, `flock`, `unshare`, or `taskset` now classify the same as their unwrapped form instead of passing the scanner unnoticed.

— Names all six wrappers and the before/after behaviour; no user action needed. v0.18.7

THINNER COVERAGE BELOW

04 **JSONL decision log output** NEW 55

`doberman log --jsonl` output records which code path resolved an authentication and whether a person actually approved it in every decision-log row.

— Names the flag but gives no usage example. v0.18.7

05 **Re-verification of approved deletes before execution** IMPROVED 45

Deletes approved through a host hook are re-verified before execution, showing the blast radius and denying if the file set changed since approval.

— Mechanism described but no named surface or user action. v0.18.7

06 **Elevation revocation on post-approval BLOCK** IMPROVED 45

A post-approval `BLOCK` now revokes the elevation it just granted instead of leaving it live until its TTL expires.

— Behaviour change explained but no configurable surface or example. v0.18.7

└──▷ **BREAKING ON UPGRADE**

! Existing Claude Code hook entries lack the new `timeout` pin and will continue to fail open on timeout until you re-run `doberman install-hooks` (add `--host cursor` for Cursor); `doberman doctor` flags un-pinned entries.

WAS THIS USEFUL? 👍 👎

garak

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

Garak scans language models and applications for vulnerabilities including prompt injection and data leakage.

> garak v0.17.0 adds EU AI Act risk-category mapping to probe results, extends the Ollama generator with authentication and custom client support, and drops Python 3.10 in favor of 3.13.

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

01 **Ollama generator auth and parameter forwarding** IMPROVED 55

The Ollama generator now supports authentication and custom client kwargs, and forwards generation parameters to Ollama via the `options` dict.

— Names the options dict but no exact flags or config keys v0.17.0

02 **EU AI Act risk-category tags for probes** NEW 45

Adds EU AI Act risk-category reference tags so probe results can be surfaced and grouped by EU AI Act risk categories.

— Describes the capability but no concrete usage path given v0.17.0

03 **Python version support updated** BREAKING 35

Python 3.10 is no longer supported; Python 3.13 is now supported.

— Clear version change but minimal explanation of impact v0.17.0

└──▷ **BREAKING ON UPGRADE**

! Python 3.10 is no longer supported; Python 3.13 is now supported.

WAS THIS USEFUL? 👍 👎

ToolHive

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗ RANK

ToolHive runs MCP servers in isolated containers and enforces identity and access policies for local and Kubernetes deployments.

> ToolHive v0.48.0 focuses on tightening authentication for MCP upstreams and hardening plugin/tool verification, alongside improvements to vMCP session handling and proxy timeout configuration.

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

01 **vMCP session handling and eviction** NEW 55

The vMCP session factory gains a `dial-control` option for fine-grained connection control, and vMCP now automatically evicts sessions when a backend is dropped, preventing stale session accumulation.

– Names the dial-control option but lacks usage detail v0.48.0

02 **Plugin verification and tool discovery hardening** IMPROVED 55

Multiple security hardening changes: ToolHive now fails closed on malformed protected list responses, fails closed on keyed skill upgrade verdicts, honours a pinned cosign key during plugin sync and upgrade operations, requires a discovery capability for key signing, and hardens find-tool response filtering to reduce information exposure.

– Enumerates each hardening behavior but no config surface named v0.48.0

03 **OIDC dynamic client registration** NEW 45

Supports OIDC dynamic client registration for OAuth2/OIDC upstreams.

– Names the mechanism but no config or command detail v0.48.0

04 **SPIFFE client authentication dispatch** NEW 45

Adds SPIFFE client authentication dispatch, enabling workload-identity-based auth routing.

– Names the mechanism without deployment or config specifics v0.48.0

05 **Basic auth for OAuth2 upstream token exchange** IMPROVED 30

Uses Basic auth for OAuth2 upstreams during token exchange flows.

– Brief note on auth method with no further mechanism v0.48.0

WAS THIS USEFUL? 👍 👎

agent-safe-pipeline

Feedback ⓘ 1 RELEASE · 2026-09-10

NOTES ↗ RANK

Agent Safe Pipeline provides a security-focused framework for building AI agent workflows with policy checks and safety controls.

> Agent Safe Pipeline's v0.1.4 release adds a CommerceGate MCP package, a wire-contract harness with isolated shadow mode, managed Presence escalation testing, and provider outcome auditing with a maintenance policy.

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

01 **CommerceGate MCP package** NEW 45

Adds the `@decionis/commerce` CommerceGate MCP package, housed under `packages/com...`, with directory bundle preparation support.

— Names package and path but truncated, no usage example v0.1.4

02 **Wire-contract harness with shadow mode isolation** NEW 35

Adds a wire-contract harness and isolates shadow mode, conforming to the Decionis execution contract.

— Describes mechanism but no command or config to act on v0.1.4

03 **Managed Presence escalation testing** NEW 35

Adds managed Presence escalation with evidence demos and local Presence escalation testing backed by a truthful fixture ledger.

— Names feature and fixture ledger but no concrete usage v0.1.4

04 **Auditable provider outcomes and maintenance policy** NEW 15

Models auditable provider outcomes and publishes a maintenance policy.

— Generic description with no named surfaces or mechanism v0.1.4

WAS THIS USEFUL? 👍 👎

Certiv

Feedback ⓘ 1 RELEASE · 2026-09-04 BLOG ↗ RANK

Certiv provides endpoint visibility and control for locally installed AI agents, helping organizations discover, understand, and safeguard AI actions.

> Certiv launched Certiv Cost in public preview, giving teams session-level token attribution and configurable spend controls for locally installed AI agents.

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

01 Budget simulation and enforcement in Certiv Cost NEW

74

Certiv Cost adds budget simulation that replays historical usage data to preview how often a proposed dollar threshold would have been hit before switching an endpoint from simulation to enforcement, plus support for entering negotiated per-model rates so thresholds are computed against actual contracted pricing rather than list pricing. Endpoints can be set to Notify or Block enforcement mode; in Block mode, future prompts on that endpoint are stopped before the request is sent once the ceiling is reached.

Contain a potentially runaway agent by enforcing a hard spending ceiling on a single endpoint — stopping future prompts before they leave.

> 📍 In Certiv Cost, select the target endpoint, set a dollar budget for your chosen period, and set the enforcement mode to Block. Future prompts on that endpoint are stopped before the request is sent once the ceiling is reached.

Validate a proposed budget ceiling against real historical traffic before enforcing it, to avoid blocking legitimate work.

> 📍 In Certiv Cost, open the budget simulation view for the endpoint, enter a proposed dollar threshold, and review how many times that ceiling would have been triggered in the historical usage period. Adjust the value until the hit rate reflects only anomalous sessions, then switch to Notify or Block mode.

— Names enforcement modes and simulation workflow but no API/CLI surface

launch-20260904-934ae61c

THINNER COVERAGE BELOW

02 Certiv Cost token attribution NEW

55

Certiv Cost is a new public-preview capability that provides session-level token attribution broken out by agent, endpoint, user, and model, along with historical usage trends.

— Describes attribution dimensions but no config or command to try it

launch-20260904-934ae61c

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Frontier Models

Anthropic

Feedback ⓘ 2 RELEASES · SEEN 2026-09-10 CHANGELOG ↗

RANK

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

> Anthropic's CLI gained a declarative `ant apply` workflow for managing agents, environments, skills and more, while per-message effort changes reached Google Cloud for three Claude models.

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

01 `ant apply` for declarative resource management NEW

88

CLI v1.30.0 adds the `ant apply` subcommand to create and update agents, environments, skills, memory stores, and deployments from files in your repository, printing a plan for approval before making changes. Each run writes a `claude-lock.json` lockfile so subsequent runs — locally or in CI — update existing resources rather than creating duplicates.

Declare an agent resource in a file and apply it to create or update the agent in your Anthropic environment, then commit the lockfile for reproducible CI runs.

```
$ ant apply
```

— Names exact subcommand, lockfile, and workflow with runnable example.

changelog-20260910-c86646a9

02 Per-message effort changes on Google Cloud IMPROVED

65

Per-message effort changes (beta) is now extended to Google Cloud for Claude Fable 5.1, Claude Mythos 5.1, and Claude Opus 5, accessed via the `mid-conversation-output-config-2026-07-01` beta header.

— Names beta header and models but no usage example. snapshot-20260910

WAS THIS USEFUL? 👍 👎

◆ Open-Weight Models

DeepSeek Models

Feedback ⓘ 1 RELEASE · 2026-09-10 NOTES ↗

RANK

The DeepSeek API uses an API format compatible with OpenAI/Anthropic. By modifying the configuration, you can use the OpenAI/Anthropic SDK or softwares compatible with the OpenAI/Anthropic API to access the DeepSeek API.

> DeepSeek launched V4.1-Flash, a new smallest-in-family model with native multimodal visual understanding and lower pricing, while retiring older V4 Flash variants in favor of automatic routing to the new model.

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

01 **Legacy V4 Flash model names rerouted to V4.1 Flash** BREAKING

70

After 12:00 Beijing Time on September 14, 2026, all requests to `deepseek-v4-pro` will be routed to V4.1 Flash and billed at the V4.1 Flash price. The model names `deepseek-v4-flash` and `deepseek-v4-flash-vision-exp` are now temporarily routed to V4.1 Flash, and the original V4 Flash and V4 Flash Vision Exp models have been retired.

— Exact model names and cutover date given, no migration steps Date: 2026-09-10

02 **DeepSeek-V4.1-Flash model launch** NEW

67

DeepSeek released DeepSeek-V4.1-Flash, callable via the model name `deepseek-flash` on the DeepSeek API. It is the smallest model in a new architecture family designed for a higher capability ceiling, faster inference, and higher throughput, and it includes native multimodal visual understanding built in. API pricing for DeepSeek-V4.1-Flash was also reduced relative to prior generation models.

— Names model, call name, and pricing change but no numbers Date: 2026-09-10

└──▷ BREAKING ON UPGRADE

! After 12:00 Beijing Time on September 14, 2026, all requests to `deepseek-v4-pro` will be routed to V4.1 Flash and billed at the V4.1 Flash price.

! The model names `deepseek-v4-flash` and `deepseek-v4-flash-vision-exp` are now temporarily routed to V4.1 Flash; the original V4 Flash and V4 Flash Vision Exp models have been retired.

WAS THIS USEFUL? 👍 👎

Cohere Models

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

Cohere develops Command, Aya, Embed, Rerank, transcription, and document-understanding models.

> Cohere launched North Small Translate, a dedicated machine-translation model available through the API, as open weights, and for private enterprise deployment.

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

01 **North Small Translate machine-translation model** NEW

90

`north-small-translate-1-0` is a 218B-parameter (25B active) mixture-of-experts model purpose-built for machine translation across more than 50 languages and locale variants. It is available via the Chat V2 API on the free tier with a 16K context length, as FP8 open weights on Hugging Face under CC BY-NC 4.0 for non-commercial use, and for private enterprise deployment through Model Vault under a commercial license with a suggested hardware configuration of two H100 GPUs or one B200 GPU.

— Names model, params, API, tiers, weights license, and hardware specs September 9, 2026

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · September 10, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue