HEADS UP This site is currently under heavy development.

THE AI TOOLCHAIN NO. 003

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED AUGUST 21, 2026 · EVERY WEEKDAY

EDITIONS **tail** | grep | head | diff | uniq

Every release worth knowing about, the day it ships.

// HOW THIS ISSUE IS MADE

We read every release from the 119 tools on our watchlist at the source — **GitHub and GitLab release notes**, vendor **release pages** and **changelogs**, project **blogs and feeds**, vendor **press releases**, and the **source code** behind the tag. Bug-fix-only releases and non-product newsroom noise are dropped; what's left is summarized down to the new capability, how to try it, and any **screenshots or videos** the release itself published. Every entry links to the sources it was built from.

// CONTENTS

AI & LLM TOOLING

AI Model & Data Infrastructure	Anthropic · OpenAI · Ollama · Together AI · RunPod
AI Coding Agents	Daytona · Amazon Kiro · Cline · OpenAI Codex CLI · Anthropic Claude Code · Alibaba Qwen Code
Local LLM Runtimes	llama.cpp · vMLX · LocalAI
AI Agent Frameworks	AutoGPT · LangChain · Agno (formerly Phidata)

OTHER / UNCATEGORIZED

AI OBSERVABILITY	LangChain LangSmith · PromptLayer · Langfuse · Braintrust

VECTOR DB RAG	Pinecone

AI & LLM TOOLING

◆ AI Model & Data Infrastructure

Anthropic

1 RELEASE · seen 2026-08-21 NOTES ↗

Anthropic develops Claude, an AI assistant API for developers to build applications with advanced language understanding and reasoning capabilities.

> Claude's API this cycle promoted computer use to general availability with a new toolset name and batch actions, introduced a brand-new browser use toolset for driving an in-app browser, and brought Admin API user-management endpoints to general availability for Enterprise orgs.

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

01 **Computer use toolset reaches general availability** BREAKING

83

The computer use tool is promoted to general availability as `computer_toolset_20260801`, requiring no beta header. It adds batch actions (multiple actions per turn), zoom enabled by default, and per-member configuration via `configs`. Upgrading an existing integration to `computer_toolset_20260801` changes the request shape and tool handling, so migration from `computer_20251124` is required.

— Names exact toolset id, config key, and migration requirement. snapshot-20260821

02 **New browser use toolset** NEW

77

`browser_toolset_20260801` is a new client toolset for driving an application-hosted browser viewport. It reads the page accessibility tree, elements, forms, and tabs, and adds element references, form input, tab management, download reporting, and opt-in file upload on top of screenshot-and-click control.

— Names the toolset id and lists concrete capabilities added. snapshot-20260821

└──▷ BREAKING ON UPGRADE

! Upgrading an existing computer use integration to `computer_toolset_20260801` changes the request shape and tool handling; migration from `computer_20251124` is required (see 'Migrate from computer_20251124').

WAS THIS USEFUL? 👍 👎

OpenAI

1 RELEASE · seen 2026-08-21 NOTES ↗

OpenAI provides APIs and tools for developers to integrate advanced AI models like GPT into applications for natural language processing and generation.

> OpenAI added transparent background generation for gpt-image-2 models and shipped a new Prompt Caching dashboard for monitoring cache performance.

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

01 Transparent background for gpt-image-2 NEW

90

Adds `background: transparent` support (preview) for `gpt-image-2` and `gpt-image-2-2026-04-21` in the `v1/images/generations`, `v1/images/edits`, and `v1/responses` endpoints. Output format must be `png` or `webp`; `jpeg` does not support transparent backgrounds.

Generate a product image with a transparent background for compositing — useful when you need to layer the result over a custom background without post-processing.

```
$ curl https://api.openai.com/v1/images/generations \
  -H 'Authorization: Bearer $OPENAI_API_KEY' \
  -H 'Content-Type: application/json' \
  -d '{
    "model": "gpt-image-2",
    "prompt": "A sleek wireless headphone on a clean surface",
    "background": "transparent",
    "output_format": "png"
  }'
```

— Names endpoints, models, flag and format constraints with a runnable example.

snapshot-20260821

02 Prompt Caching dashboard NEW

60

Releases a Prompt Caching dashboard on the OpenAI API platform showing cache hit rate over time, cache reads per write, and a breakdown of cache-read, cache-write, and uncached tokens; filterable by model and service tier.

— Describes dashboard metrics and filters but no exact UI path or API. snapshot-20260821

WAS THIS USEFUL? 👍 👎

Ollama

1 RELEASE · 2026-08-19 NOTES ↗ CODE ↗

Get up and running with Kimi-K2.6, GLM-5.2, MiniMax, DeepSeek, gpt-oss, Qwen, Gemma and other models.

> Ollama v0.32.15 introduces a guided first-launch onboarding flow for the desktop app and a model metadata cache that roughly halves time-to-first-token.

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

THINNER COVERAGE BELOW

01 **Desktop onboarding flow** NEW 58

On first launch, Ollama now guides new users through sign-in, local-only, or skip options before presenting the `ollama` command.

— Names the exact options but no command example. v0.32.15

02 **Model metadata cache speeds up first token** IMPROVED 20

A new model metadata cache cuts time-to-first-token roughly in half, per the release summary.

— No mechanism or config detail beyond the headline claim. v0.32.15

WAS THIS USEFUL? 👍 👎

Together AI

1 RELEASE · 2026-08-18 NOTES ↗

Together AI provides an API platform for running and fine-tuning open-source large language models at scale.

> Together AI added a fully automatic confirmation policy for node auto repair, letting operators choose which fault types run unattended.

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

01 **Fully automatic policy for node auto repair** NEW 75

A new Fully automatic option under Auto-remediation policy on the Repairs tab lets auto node repair run end-to-end without manual approval. Per-fault scoping via Repair actions lets operators choose which fault groups — Migrate to new host, Reprovision, VM reboot — run unattended under Fully automatic, so destructive repairs can stay gated while transient ones self-clear; a Wait policy set to Grace period can protect in-flight jobs during repair.

Enable fully automatic repair for transient faults while keeping destructive actions gated, protecting running jobs with a grace period.

> 📍 In the console, go to the Repairs tab › Auto-remediation policy › set Confirmation policy to 'Fully automatic' › open Repair actions and enable 'VM reboot' for unattended execution, leave 'Reprovision' and 'Migrate to new host' requiring approval › set Wait policy to 'Grace period' to protect in-flight jobs.

— Names exact UI path and fault types, but no API/CLI equivalent given.

Fully automatic confirmation policy for …

WAS THIS USEFUL? 👍 👎

RunPod

1 RELEASE · 2026-08-01 NOTES ↗

RunPod is a cloud platform providing serverless GPU computing and pod infrastructure for AI model training, inference, and development.

> RunPod introduced Batch Jobs (BETA), a full REST API and Console workflow for submitting, tracking, and managing large asynchronous inference workloads on Serverless endpoints without impacting interactive traffic.

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

01 Batch Jobs (BETA) API for async serverless inference NEW

96

RunPod added a dedicated batch processing system for Serverless endpoints with a full REST API: `POST /v2/{endpoint_id}/batch` creates a batch (optionally seeded with an initial array of `/run` `-shaped` requests), `POST /v2/{endpoint_id}/batch/{id}/requests` appends up to 10 MiB of additional requests to a DRAFT batch across multiple calls, `POST /v2/{endpoint_id}/batch/{id}/finalize` locks the batch to `FINALIZED` and makes it eligible for execution, `GET /v2/{endpoint_id}/batch/{id}` returns `requestTotal`, `requestInProgress`, `requestCompleted`, and `requestFailed` counts for polling, `GET /v2/{endpoint_id}/batch/{id}/requests` returns paginated per-request `status`, `output`, `error`, `startedAt`, and `completedAt` fields via `offset` / `limit` query parameters with a `hasMore` flag, `POST /v2/{endpoint_id}/batch/{id}/cancel` cancels a batch (queued requests are cancelled and unbilled, in-progress requests finish normally), `GET /v2/{endpoint_id}/batch` lists all batches newest first, `PUT /v2/{endpoint_id}/batch/{id}` updates attributes like display name, and `DELETE /v2/{endpoint_id}/batch/{id}/requests/{requestId}` removes a single request from a DRAFT batch. Batches run on dedicated workers isolated from standard `/run` traffic, with limits of 10 active batches per endpoint, 5,000 requests per batch, and 50,000 queued requests per endpoint (up to 1,000,000 daily for enterprise); a Console Batch tab shows per-batch progress and per-request rows sorted failures-first, webhook and Console Inbox notifications fire once on terminal state (`FAILED` / `CANCELLED`) with batch ID, endpoint name, status, and counts, and batch jobs are billed at standard serverless request rates with flex worker discounts for enterprise.

Kick off a nightly embedding run by creating a batch with an initial set of inputs, then finalizing it so dedicated batch workers begin processing asynchronously.

```
$ # 1. Create batch with initial requests
curl -X POST https://api.runpod.io/v2/{endpoint_id}/batch \
  -H 'Authorization: Bearer {api_key}' \
  -H 'Content-Type: application/json' \
  -d '[{"input":{"text":"The quick brown fox"}},{"input":
{"text":"Jumped over the lazy dog"}}]'

# 2. Finalize to start processing
curl -X POST
https://api.runpod.io/v2/{endpoint_id}/batch/batch_01j9abc123/fi
nalize \
  -H 'Authorization: Bearer {api_key}'
```

Poll a running batch until all requests have finished, using the request counts rather than a COMPLETED status.

```
$ curl -X GET
https://api.runpod.io/v2/{endpoint_id}/batch/batch_01j9abc123 \
  -H 'Authorization: Bearer {api_key}'
# Batch is done when requestCompleted + requestFailed ==
requestTotal
```

Page through completed results after a batch finishes, inspecting outputs and error messages for failed child requests.

```
$ curl -X GET
'https://api.runpod.io/v2/{endpoint_id}/batch/batch_01j9abc123/r
equests?offset=0&limit=50' \
  -H 'Authorization: Bearer {api_key}'
```

— Names every endpoint, limit, field, and billing detail with runnable curl examples.

New ReleaseBatch Jobs (BETA)

◆ AI Coding Agents

Daytona

1 RELEASE · seen 2026-08-21 NOTES ↗

Daytona is a cloud development environment platform that provides standardized, reproducible coding workspaces for teams and remote development.

> Daytona 0.201.0 adds pre-signed sandbox file URLs, typed SDK error codes, and a new time-based filter for sandbox listing.

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

01 **autoDestroyAt range filter for sandbox listing** NEW 62

The sandbox list SDK call gains an `autoDestroyAt` range filter, letting callers query sandboxes by their scheduled destruction time.

— Names the exact SDK filter but no code sample shown snapshot-20260821

WAS THIS USEFUL? 👍 👎

Amazon Kiro

1 RELEASE · seen 2026-08-21 NOTES ↗

> Kiro's CLI adds automatic recovery for stalled AI streams, AI-generated titles to distinguish similar chat sessions, and mouse support in the spec review screen.

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

01 **Automatic stream recovery with idle watchdog** NEW 85

Introduces a stream idle watchdog that warns after 60 seconds of silence and cancels at 300 seconds, with automatic retries for throttling, 5xx errors, and connection drops. New config settings `api.streamIdleSoftTimeout`, `api.streamIdleHardTimeout`, and `api.timeout` let users tune the watchdog and a 60-minute streaming timeout for stalled or dropped connections.

— Names exact config keys and timeout values, no example usage given

Spec Review Mouse Support, AI Session Ti…

02 **AI-generated session titles in resume picker** NEW 67

Adds AI-generated session titles to the V3 session resume picker, surfaced via `kiro-cli chat --resume-picker` or `/chat resume`, so similar sessions are distinguishable at a glance.

Resume a past session by name when you have many similar sessions and need the AI-generated titles to tell them apart.

```
$ kiro-cli chat --resume-picker
```

— Names exact command and includes runnable example

Spec Review Mouse Support, AI Session Ti…

THINNER COVERAGE BELOW

03 **Mouse support in spec review screen** NEW 50

Adds mouse support to the spec review screen — scroll to navigate and click to position the cursor — with `m` toggling mouse support on or off.

— Describes behavior and toggle key but no full command context

Spec Review Mouse Support, AI Session Ti…

WAS THIS USEFUL? 👍 👎

Cline

5 RELEASES · 2026-08-20 → 2026-08-21 NOTES ↗ CODE ↗

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

> Across desktop, CLI, and SDK releases, Cline added inline model-generated images, agent-managed todos and schedules, a unified Plugins hub, and eight new model providers, while introducing a `/handoff` command that moves local sessions into Cline Cloud.

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

01 `/handoff` command moves sessions to Cline Cloud NEW

85

Adds the `/handoff` command to transfer a local session — conversation, attached images, and an optional follow-up prompt — to a Cline Cloud workspace that continues running after the app closes. Integrates Cloud into the existing Local / Remote environment menu, gated behind the Cloud sessions feature flag (shows 'Coming soon' when off), and adds interrupted-handoff recovery: if the app restarts, the network drops, or the branch moves mid-transfer, reopening the session resumes or cleanly retries the handoff and restores typed drafts and attachments.

— Names exact command and recovery flow, missing feature-flag name `desktop-v0.0.15-beta.1`

02 Model picker tiers and non-chat model filtering IMPROVED

78

Model selector in the composer and provider settings now lead with Recommended and Free tiers (labeled 'Subscribed' and 'Free' on ClinePass), showing display names with descriptions instead of an alphabetized list of raw model IDs. Image, voice, and other non-chat models are excluded from onboarding, model pickers, and ACP model listings, and the CLI's `--model` flag now rejects them outright.

— Names `--model` flag and tier labels but no full command example `desktop-v0.0.15` `cli-v3.0.56` `sdk/sdk/v0.0.76`

03 Inline model-driven image generation NEW

75

Models that support image generation can now produce images during a task, rendered inline in the conversation. The CLI/TUI saves each generated file to a temporary path, HTML session exports embed images inline, and ACP clients receive them as image content; the SDK persists generated images in session history and exports.

— Explains mechanism across products but no flag to enable it `v4.1.11` `cli-v3.0.56` `sdk/sdk/v0.0.76`

04 Eight new model providers added to catalog NEW

65

Adds AMD, Arcee, Echo, Jalapeno, Kosmik, LLM Gateway, RunInfra, and SCNet as new model providers across desktop, CLI, and SDK, with updated model lists, pricing, and per-provider default models in the refreshed model catalog.

— Names all eight providers but no selection command shown `sdk/sdk/v0.0.76` `cli-v3.0.56` `v4.1.11` `desktop-v0.0.15`

05 **Provider `scx` renamed to `scx-ai`** BREAKING 65

The provider `scx` is renamed to `scx-ai` in the model catalog; any configuration or code referencing `scx` by name will need to be updated.

— Clear rename with migration note, no automated migration path given `sdk/sdk/v0.0.76`

06 **Skill slash command routing fix** IMPROVED 65

Skill slash commands now load through the skills tool instead of being expanded into the user message, so instructions reach the model once instead of twice; the persisted transcript/history records the typed command (e.g. `/my-skill ...`) rather than the full SKILL.md body.

— Explains before/after mechanism, no config key involved `cli-v3.0.56` `sdk/sdk/v0.0.76`

THINNER COVERAGE BELOW

07 **Agent-managed todos and schedules** NEW 55

Agents can now create and manage durable todos and one-time or recurring schedules directly within sessions; in the SDK, schedules are scoped to the workspace that registered them.

— Describes scope but no config key or command shown `desktop-v0.0.15` `sdk/sdk/v0.0.76`

08 **AI SDK telemetry spans gain context fields** IMPROVED 50

AI SDK telemetry spans now carry user, session, conversation, run, provider, and model context under a stable `cline-sdk` service name.

— Names span fields and service name, no usage instructions `sdk/sdk/v0.0.76`

09 **Unified Plugins hub with Marketplace** NEW 45

Plugins, MCP servers, and Skills are unified into a single Plugins hub with a dedicated Marketplace page.

— Names the hub and marketplace but no navigation path given `desktop-v0.0.15` `desktop-v0.0.15-beta.1`

10 **Local/Cloud toggle removed from environment menu** BREAKING 45

The separate Local / Cloud toggle is removed; Cloud environment selection now lives exclusively in the Local / Remote environment menu.

— States removal but gives no further migration steps `desktop-v0.0.15-beta.1`

---

11 **PowerShell commands fail fast on error** IMPROVED 40

PowerShell commands now fail fast on the first error instead of continuing to flood output and reporting success.

— Clear behavior change but no flag or setting named `desktop-v0.0.15`

---

12 **Detached command output streams live** IMPROVED 40

Detached command output now streams live and survives hub restarts instead of being reaped.

— Describes behavior change, no config or command named `sdk/sdk/v0.0.76`

---

13 **Billed gateway cost shown in usage displays** IMPROVED 30

Usage displays across desktop, CLI, and SDK now show the billed gateway cost.

— Bare description, no numbers or location specified `desktop-v0.0.15` `v4.1.11` `cli-v3.0.56` `sdk/sdk/v0.0.76`

---

14 **Refreshed first-run onboarding** IMPROVED 20

Refreshed first-run onboarding with an interactive welcome graphic.

— Purely cosmetic description with no specifics `desktop-v0.0.15`

└──▷ BREAKING ON UPGRADE

! The provider `scx` is renamed to `scx-ai` in the model catalog; any configuration or code referencing `scx` by name will need to be updated.

! The separate Local / Cloud toggle is removed; Cloud environment selection now lives exclusively in the Local / Remote environment menu.

WAS THIS USEFUL? 👍 👎

OpenAI Codex CLI

4 RELEASES · 2026-08-20 → 2026-08-21 NOTES ↗

CODE ↗

Lightweight coding agent that runs in your terminal

> Codex CLI's alpha builds add Amazon Bedrock multi-agent V1 support with a breaking change to remote compaction, macOS app bundle signature verification, and shell snapshot caching, while the stable 0.149.0 release ships an interactive `codex agents` dashboard, a `codex queue` command, TUI working-directory shortcuts, and expanded `codex doctor` diagnostics.

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

01 Additional developer instructions in managed requirements NEW

85

Adds `additional_developer_instructions` to managed requirements, exposed via `configRequirements/read` as `additionalDeveloperInstructions`, with a 10,000 estimated-token size limit, and support across compaction, resume, and agent forks.

— Names exact config key, endpoint field and token limit. `rust-v0.150.0-alpha.2` `rust-v0.150.0-alpha.1`

02 Amazon Bedrock multi-agent V1 and compaction protocol change

80

BREAKING

Enables the multi-agent V1 protocol for Amazon Bedrock models, normalizing Bedrock model catalogs to advertise `MultiAgentVersion::V1` to work around Bedrock's lack of support for response items required by multi-agent V2. Also switches Amazon Bedrock remote compaction to use `compaction_trigger` items sent through `/v1/responses` instead of the legacy dedicated compaction protocol, and explicitly enables remote compaction for OpenAI, Azure Responses, and Amazon Bedrock providers; the legacy dedicated compaction protocol is removed from provider capabilities, so remote compaction now defaults to unsupported and must be explicitly enabled per provider.

— Names protocol version, endpoint, and removed legacy protocol precisely. `rust-v0.150.0-alpha.2` `rust-v0.149.0-alpha.4.1`

03 macOS Codex desktop app signature verification NEW

80

Verifies the Codex desktop app bundle signature using Apple's `codesign --verify --deep --strict` against OpenAI's bundle identifier `com.openai.codex` and Apple Team ID `2DC432GLL2` before launching or installing on macOS.

— Names exact codesign command, bundle ID and team ID. `rust-v0.150.0-alpha.2`

`rust-v0.150.0-alpha.1`

04 **Expanded codex doctor diagnostics** IMPROVED 75

Expands `codex doctor` to diagnose endpoint protection, network/proxy failures, desktop app state, Windows sandbox provisioning, and update connectivity.

— Names command and lists all new diagnostic areas. `rust-v0.149.0`

05 **Shell snapshot caching for exec sessions** NEW 75

Adds a `shellSnapshotV2` executor capability and an optional shell snapshot request to `ExecParams`, caching and restoring Unix shell state and profile exports for `bash`, `zsh`, and `sh` in an in-memory, attachment-scoped cache.

— Names capability flag, ExecParams field and cached shells. `rust-v0.150.0-alpha.1` `rust-v0.150.0-alpha.2`

06 **codex queue command for session messaging** NEW 70

Adds `codex queue` command for sending messages to existing local or remote sessions.

Send a follow-up instruction to an already-running local or remote session without attaching to it interactively.

```
$ codex queue <session-id> 'Run the full test suite and report failures'
```

— Comes with a runnable example command. `rust-v0.149.0`

07 **TUI working directory commands** NEW 70

Adds `/cd`, `/pwd`, and `/cwd` commands for managing the working directory inside TUI sessions.

— Names exact slash commands usable directly in TUI. `rust-v0.149.0`

08 **Expanded Vim editing in TUI composer** IMPROVED 65

Expands Vim editing in the TUI composer with character replacement and change motions including `cw`, `c$`, and `cc`.

— Names the specific Vim motions added. `rust-v0.149.0`

THINNER COVERAGE BELOW

09 **SDK config overrides and reasoning effort levels** NEW 55

SDK users can now pass exact CLI config overrides and select `max` or `ultra` reasoning effort.

– Names the two new effort levels but no code sample. `rust-v0.149.0`

10 **Bounded filesystem directory walks** IMPROVED 55

Requires filesystem backends to implement directory walks natively, with bounded local walks including symlink cycle detection, deterministic ordering, error collection, and response-size limits.

– Describes mechanism but no config surface to invoke. `rust-v0.150.0-alpha.2` `rust-v0.150.0-alpha.1`

11 **Hostname in TUI status line** NEW 30

Adds hostname to the configurable TUI status line.

– Brief description with no configuration detail. `rust-v0.150.0-alpha.2` `rust-v0.150.0-alpha.1`

12 **Host-accepted exec-server WebSockets** NEW 30

Supports host-accepted exec-server WebSockets.

– Names the surface but no usage detail. `rust-v0.150.0-alpha.1`

13 **App-server MCP event streaming** NEW 25

Adds app-server MCP event streaming support.

– Only a bare feature name, no mechanism given. `rust-v0.150.0-alpha.2` `rust-v0.150.0-alpha.1`

14 **Turn cost telemetry for custom model providers** NEW 25

Supports turn cost telemetry for custom model providers.

– Bare capability name without mechanism or metrics detail. `rust-v0.150.0-alpha.2` `rust-v0.150.0-alpha.1`

15 **Standalone named function call outputs** NEW 25

Supports standalone named function call outputs handled as external context.

– Vague description without concrete usage or example. `rust-v0.150.0-alpha.2` `rust-v0.150.0-alpha.1`

16 **Registry connection retry on startup** IMPROVED 25

Retries transient registry failures during initial exec server connection.

– Describes behavior but no retry count or config. `rust-v0.150.0-alpha.2`

rust-v0.150.0-alpha.1

17 **Plugin install suggestion IDs** NEW 20

Includes suggestion IDs in plugin install metadata.

– No explanation of how suggestion IDs are used. rust-v0.150.0-alpha.2 rust-v0.150.0-alpha.1

18 **History and notes tools for token-budget sessions** NEW 20

Adds history and notes tools for token-budget sessions.

– No detail on tool interface or invocation. rust-v0.150.0-alpha.2

19 **Bundled model definitions refresh** IMPROVED 10

Refreshes bundled model definitions.

– No specifics on which models or changes. rust-v0.150.0-alpha.2 rust-v0.150.0-alpha.1

└──▷ **BREAKING ON UPGRADE**

! The legacy dedicated compaction protocol is removed from provider capabilities; remote compaction now defaults to unsupported and must be explicitly enabled per provider.

WAS THIS USEFUL? 👍 👎

Anthropic Claude Code

1 RELEASE · 2026-08-20 NOTES ↗ CODE ↗

Claude Code is an agentic coding tool that lives in your terminal, understands your codebase, and helps you code faster by executing routine tasks, explaining complex code, and handling git workflows - all through natural language commands.

> Claude Code v2.1.238 adds operational controls for self-hosted runners (graceful shutdown, proxy authorization), authenticated plugin marketplace fetches, and tightens MCP headersHelper trust and credential handling, alongside smaller UX and messaging reliability fixes.

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

01 **Graceful shutdown for self-hosted runners** NEW 85

Adds `claude self-hosted-runner --defer-shutdown-max-min <minutes>`: on SIGTERM the runner keeps serving already-attached sessions, parks any remaining sessions once the specified number of minutes elapses, then exits.

Allow a self-hosted runner to finish active sessions gracefully before exiting after receiving SIGTERM — useful for rolling restarts without dropping users.

```
$ claude self-hosted-runner --defer-shutdown-max-min 5
```

— Names exact flag and behaviour with a runnable example. v2.1.238

02 **Proxy authorization for self-hosted runner egress** NEW 85

Adds `claude self-hosted-runner --proxy-authorization-command` and `--proxy-authorization-file` so runners behind egress proxies that require a freshly issued `Proxy-Authorization` header on every connection can supply one dynamically.

Authenticate to an egress proxy that requires a fresh token on each connection by supplying a command that generates the `Proxy-Authorization` header value.

```
$ claude self-hosted-runner --proxy-authorization-command 'vault
read -field=token secret/proxy-auth'
```

— Two named flags plus a runnable example command. v2.1.238

03 **Plugin marketplace headersHelper for authenticated catalogs** NEW 83

Adds `headersHelper` to plugin marketplace URL entries and catalog entries: it runs a command to mint HTTP headers (e.g. a short-lived token) for catalog and same-origin archive fetches. A catalog entry's `headersHelper` runs only at `claude plugin install` or `claude plugin update` time, after its command is shown, behind a `[y/N]` prompt (or `-y` to skip).

— Names config field, commands and prompt flag, but no runnable example. v2.1.238

---

04 **MCP headersHelper trust and credential isolation** BREAKING

76

MCP `headersHelper` in a project `.mcp.json`, and inline MCP servers in project or `--add-dir` agent files, now require the folder's trust dialog to have been accepted (also enforced under `claude -p`). MCP `headersHelper` from a project `.mcp.json`, plugin, or agent file now runs without inherited credential environment variables; user-, managed-, and claude.ai-scope helpers instead run from the Claude config directory.

— Names exact config surfaces and new restrictions but no command to try. v2.1.238

---

05 **Cross-session messaging failure notifications** IMPROVED

65

Cross-session messaging now reports 'refused' back to the sender when the target session has `crossSessionInbound: "refuse"` set, instead of silently succeeding, and notifies the sending session when the target inbox drops messages due to a rate limit or full queue instead of messages vanishing silently.

— Names the config field driving the behaviour change. v2.1.238

---

06 **Readline-style keybinding flavor** NEW

65

Adds a `keybindingFlavor` setting; set to `"readline"` to make Ctrl+W delete back to the previous whitespace (Bash-style), while the default `"classic"` behavior is unchanged.

— Names the exact setting and its two values. v2.1.238

---

07 **Disabled server status in mcp list/get** IMPROVED

60

Changes `claude mcp list` and `claude mcp get` to show disabled servers as `⊘ Disabled` instead of connecting to them for a health check.

— Names exact commands and new display state. v2.1.238

THINNER COVERAGE BELOW

---

08 **Removed double-press fullscreen clear shortcut** BREAKING

53

Removes the double-press Ctrl+L / Cmd+K `/clear` shortcut in fullscreen; both keys now always repaint the screen, and 1-row nvim terminals no longer trigger automatic `/clear` loops.

— Explains the removed behaviour but no migration step. v2.1.238

09 **Managed Agents claude-api skill updates** IMPROVED 40

Updates the bundled `claude-api` skill for Managed Agents: adds web search/fetch domain settings and memory stores on self-hosted sandboxes.

— Thin description without named settings or commands. v2.1.238

└──▷ **BREAKING ON UPGRADE**

! The double-press Ctrl+L / Cmd+K shortcut to trigger `/clear` in fullscreen has been removed; those keys now only repaint.

WAS THIS USEFUL? 👍 👎

Alibaba Qwen Code

1 RELEASE · 2026-08-20 NOTES ↗

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

> Qwen Code's v0.21.15 release centers on resumable PR reviews, a new stable qwen3.8-max model, and a round of Web Shell attachment and goal-management upgrades, alongside extension installation improvements and deeper daemon observability.

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

01 **Resume interrupted PR reviews** NEW 80

Adds a `--resume` flag to `/review`, `review run`, and CI retries so an interrupted review can continue rather than restart, as long as the PR head commit has not moved since the last run.

Resume an interrupted PR review in CI without re-running from scratch, as long as the PR head commit has not changed.

```
$ qwen review run --resume
```

— Names exact flag and command, includes runnable example. v0.21.15

02 **Stable qwen3.8-max model** NEW 75

Adds the stable `qwen3.8-max` model to the Token Plan model list, selectable via `/model` alongside the existing preview version.

Switch to the stable qwen3.8-max model mid-session to use the production-grade reasoning model instead of a preview build.

```
$ /model qwen3.8-max
```

— Named model and command, with runnable switch example. v0.21.15

03 **Extension install and activation management** NEW 68

Adds authenticated HTTPS Git extension installs with configurable credential persistence, enabling secure cloning of private repositories via the daemon, plus batch APIs to update extension activation states for up to 100 extensions at once, either globally or for specific trusted workspaces.

— Names both surfaces and the 100-extension limit, no exact command. v0.21.15

04 **Web Shell attachments and Goal v3 controls** IMPROVED 68

Web Shell now supports inserting file attachments into active turns via the composer or `@` selection, with preview and queue management; file uploads are unified to offer reference-or-upload options with persistent storage and duplicate name handling; and Web Shell adopts Goal v3 controls, letting goals be managed independently of chat messages via a compact composer row.

— Names every constituent Web Shell change but only UI navigation, no command. v0.21.15

05 **Aone Code integration for /review** NEW 65

The `/review` skill now supports posting comments and approvals to Aone Code via the `a1` CLI when using the `--comment` flag.

— Names CLI and flag but no full usage example. v0.21.15

06 **ACP heap metrics in daemon status** NEW 60

ACP child processes now track and report V8 old-generation heap metrics, including peak committed memory and major-GC counts, in daemon status.

— Names specific metrics surfaced in daemon status output. v0.21.15

THINNER COVERAGE BELOW

07 **Thinking toggle for hybrid models** BREAKING 50

Qwen hybrid models now expose a simple Thinking toggle for reasoning control, removing the previous complex effort tiers for supported versions.

— Describes the change but no exact UI path or config key. v0.21.15

08 **PTY worker support for Agent View** NEW 50

Adds PTY worker support to enable managed Agent View sessions with local terminal hosts and authenticated stream forwarding.

— Names the mechanism but gives no way to invoke it. v0.21.15

09 **Bridge session watermark in live-state API** NEW 50

The bridge now exposes a strictly monotonic per-session watermark via `BridgeSessionSummary.updatedAt` in the live-state API.

— Names exact API field consumers can read directly. v0.21.15

10 **Visible context-file attachment notice in CLI** IMPROVED 50

Adds a one-time INFO message to the CLI that lists attached context files above the first user prompt, making previously invisible system prompt attachments visible in chat.

— Names UI location but no exact command or setting. v0.21.15

11 **Distributed tracing for daemon requests** IMPROVED 45

Links daemon HTTP request spans to inbound W3C `traceparent` headers to maintain continuous distributed tracing context.

— Names the header standard but no config or endpoint to act on. v0.21.15

12 **Mutation probes required for Autofix commits** IMPROVED 45

Autofix now requires mutation probes to verify that new guards or branches added in a round are covered by tests before committing changes.

— States the new verification rule but no configuration surface. v0.21.15

13 **DingTalk quoted-message media attachment** NEW 40

Enables DingTalk to download and attach media from quoted messages, allowing agents to inspect referenced images and files.

— Describes behavior but no config or invocation detail. v0.21.15

14 **Standalone conversation isolation primitives** NEW 30

Introduces primitives for standalone conversation isolation, enabling deterministic session identity and validated lineage checks.

— Vague description with no named surface or usage path. v0.21.15

15 **Privacy-preserving prompt outcome ledger** NEW 30

Sessions now persist a privacy-preserving ledger of prompt outcomes to enable accurate cold-load reconciliation without storing user content.

— Explains purpose but no mechanism or interface named. v0.21.15

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

llama.cpp ⓘ 6 RELEASES · 2026-08-20 → 2026-08-21 NOTES ↗ CODE ↗

LLM inference in C/C++

> llama.cpp shipped a wave of performance tuning across CUDA and Metal backends alongside new multimodal device-offload controls, LFM2 speculative decoding, and hardening of server API authentication.

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

01 Per-device multimodal projector offloading NEW

90

Adds the `--mmproj-device` CLI flag (short form `-mmdev`) and the `MTMD_BACKEND_DEVICE` environment variable to select which compute device runs the multimodal projector (mmproj), enabling setups like offloading mmproj to an iGPU while the main model runs on a dGPU. Passing `none` disables GPU offload for the projector entirely.

Offload the multimodal projector to a specific GPU device (e.g. iGPU) while the main model uses a different device — useful on systems with mixed GPU setups.

```
$ llama-server -m model.gguf --mmproj mmproj.gguf --mmproj-device CUDA1
```

Use the `MTMD_BACKEND_DEVICE` env var to set the mmproj device without changing command-line args — handy for preset or scripted deployments.

```
$ MTMD_BACKEND_DEVICE=CUDA1 llama-cli -m model.gguf --mmproj mmproj.gguf
```

— Named flag, env var, and runnable example commands `b10541`

02 Metal flash attention dequantizes quantized KV caches IMPROVED

85

The Metal backend now runs a preprocessing pass for `GGML_OP_FLASH_ATTN_EXT` that dequantizes quantized KV caches (`Q4_0`, `Q4_1`, `Q5_0`, `Q5_1`, `Q8_0`) into contiguous F16 scratch buffers via a new `kernel_flash_attn_ext_dequant_to_f16` kernel, instead of dequantizing in-kernel, improving accuracy and performance on Apple Silicon. It skips redundant V dequantization when V is a view of K (as in MLA-based models), and new `test-backend-ops` cases cover the MLA shape, pad path (`kv=113`), and large context (`kv=16384`).

Run perplexity evaluation on Apple Silicon with a quantized KV cache — the Metal backend now dequantizes to F16 automatically before flash attention, matching F16 KV accuracy.

```
$ llama-perplexity -m model-q8_0.gguf -fa 1 -ctk q8_0 -ctv q8_0
```

— Named kernel and test cases with a runnable example b10532

---

03 **DSpark speculative decoding for LFM2** NEW 80

Adds DSpark speculative decoding support for LFM2 models, including partial rollback for LFM2 recurrent state, enabling draft-model-accelerated inference for the LFM2.5 series via flags like `--spec-draft-n-max` and `--spec-draft-n-min`.

Run LFM2 with DSpark speculative decoding to achieve multi-fold token generation speedup on supported hardware.

```
$ llama-server -m LFM2.5-1.2B-Instruct-F16.gguf -md LFM2.5-1.2B-Instruct-DSpark-5L-draft-f16.gguf -ngl 99 -ngld 99 -fa on --spec-draft-n-max 7 --spec-draft-n-min 0
```

— Mechanism and runnable command given, no benchmark numbers b10541

---

04 **Tuned CUDA MMVQ-to-MMQ dispatch thresholds** IMPROVED 75

Adds per-hardware and per-quantization-type batch crossover thresholds for the CUDA `mul_mat_vec_q` to MMQ (int8 tensor-core) dispatch path, with tuned switch points for Blackwell (RTX 5090, DGX Spark) and Ada (RTX 4090) GPUs, delivering +23-41% decode throughput at batch size 8 for K-quants (`Q2_K`, `Q3_K`, `Q4_K`, `Q5_K`, `Q6_K`) on dense models with no low-batch regression.

— Concrete GPUs, quant types and percentages, but automatic and unconfigurable b10534

---

05 **Server auth enforced on /v1/models endpoint** BREAKING 60

The `/v1/models` endpoint is now private when server authentication is enabled, blocking unauthenticated access to model file paths that may contain PII such as home directory names. Unauthenticated clients that previously queried `/v1/models` without a key will now receive an auth error.

— Named endpoint and breaking behaviour, no migration command b10519

---

06 **IBM Granite sliding-window attention model support** NEW 60

Adds conversion and inference support for `GraniteSWAForCausalLM` and `GraniteMoeSWAForCausalLM` architectures, enabling IBM Granite models with interleaved Sliding Window Attention (SWA) and Attention Sinks. Also adds per-layer RoPE/NoPE determination via `llama_hparams::has_rope`, allowing models with mixed attention layers to load and run correctly.

— Named model classes and internal hparam, no run example `b10514`

THINNER COVERAGE BELOW

07 **Multi-GPU tensor split for LFM2 architectures** NEW 55

Adds `--split-mode tensor` support for the LFM2 and LFM2MOE model families, enabling multi-GPU tensor parallelism for these architectures.

— Named flag and model families, no example run `b10549`

08 **/metrics endpoint reachable during sleep mode** NEW 45

The `/metrics` endpoint is now accessible while llama-server is in sleep/idle state, allowing monitoring systems to scrape metrics without waking the server.

— Named endpoint and behaviour, no example given `b10519`

09 **RoPE offset support extended to more backends** IMPROVED 40

Extends `ggml_rope_set_offset` support to the OpenCL, SYCL, WebGPU, and Hexagon backends.

— Named function and backends but internal, no usage path `b10549`

10 **SYCL backend degrades gracefully with no devices** IMPROVED 40

The SYCL backend now reports zero devices instead of aborting when no SYCL-capable hardware is present, allowing tools like `llama-quantize` to run on non-SYCL hosts without crashing.

— Names affected tool but limited detail on scope `b10514`

└──▷ BREAKING ON UPGRADE

! When authentication is enabled, unauthenticated clients that previously could query `/v1/models` without a key will now receive an auth error — any tooling or scripts relying on unauthenticated access to `/v1/models` will break.

WAS THIS USEFUL? 👍 👎

vMLX

1 RELEASE · 2026-08-20 NOTES ↗

vMLX - JANGTQ Uber Compressed MLX Models - L2 Disk Cache (survives restart) + L1 Paged (super fast ttft) + Hybrid SSM Scheduler + Cont Batching + etc!

> vMLX 1.6.34 overhauls native speculative decoding (MTP) and DFlash2 session reuse for large multiturn speedups, removes the RAM preflight that blocked big model loads, and adds video input for Qwen3.5-family models.

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

01 Native MTP overhaul eliminates draft replay IMPROVED

75

Native MTP (speculative decoding) for Qwen3.8 and dots3-note now uses an aligned draft-head context cache plus generalized skip-replay so rejected drafts no longer replay through the main model at any depth. Warm in-app decode on Qwen3.8-27B rises from ~22 to 38–48 t/s, and dots3-note reaches ~41 t/s.

— Clear mechanism and before/after numbers but no user-facing flag v1.6.34

02 DFlash2 session prefix reuse for multiturn TTFT IMPROVED

75

DFlash2 now reuses session prefixes via end-of-turn cache checkpoints, draft hidden-state gap splice, and prompt-boundary snapshots, dropping warm multiturn TTFT by roughly 20x and sustaining 60+ t/s on Qwen3.8 (68.7 t/s measured); the DFlash runtime now ships in the bundle.

— Detailed mechanism and measured numbers, but activation is automatic v1.6.34

03 Prefill admission wired-limit advisory IMPROVED

73

Prefill admission rejections now include a wired-limit advisory naming the exact `sysctl`, the macOS ~84% `-of-RAM` default, and the reset-on-reboot behavior.

— Names exact sysctl and default, directly actionable for diagnosis v1.6.34

04 RAM preflight removed for large model loads IMPROVED

65

The RAM preflight check that refused large model loads has been removed; memory estimates now advise rather than block, enabling a 101GB dots3-note bundle to load on a 128GB Mac.

— Concrete before/after with real numbers, actionable outcome v1.6.34

THINNER COVERAGE BELOW

05 **Native MTP engages by default** IMPROVED 57

Bundle-temperature auto-detection, which had kept native MTP permanently off, is replaced with compatible-only detection using a deterministic-defaults sampling policy so native MTP now actually engages by default.

— Explains behavior change but no config surface named v1.6.34

06 **Stream interval defaults to 8** IMPROVED 55

Stream interval now defaults to `8`, preventing the renderer from backpressure-stalling the engine emit loop on long conversations; legacy sessions are lifted automatically to the new default.

— Names the config default value and its automatic migration v1.6.34

07 **Video input for Qwen3.5-family models** NEW 46

Qwen3.5-family models now support video input, with videos routed as sampled frames so temporal questions answer correctly.

— States capability and mechanism briefly, no usage example given v1.6.34

WAS THIS USEFUL? 👍 👎

LocalAI

1 RELEASE · 2026-08-20 NOTES ↗ CODE ↗

LocalAI is the open-source AI engine. Run any model - LLMs, vision, voice, image, video - on any hardware.

> LocalAI 4.9.0 makes authentication deny-by-default across every HTTP route, adds a new KNN-based prompt router with corpus management endpoints, and ships chat context compression, PII pseudonymization, MiniMax-H3 video+audio generation, Qwen3-TTS, and broadened CUDA hardware support alongside UI consolidation and 85 new gallery models.

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

01 KNN router with managed prompt corpus NEW

95

Introduces a `classifier: knn` router type that does similarity-weighted voting over a persisted JSONL corpus stored under `<data path>/router-corpus`, governed by a configurable `knn.similarity_threshold`; prompts below threshold return an undecidable fallback instead of a forced guess. The corpus is managed via `POST /api/router/{name}/corpus`, `GET /api/router/{name}/corpus/stats`, and `DELETE /api/router/{name}/corpus`, all admin-gated and also exposed as MCP tools.

Seed the KNN router corpus with a labelled example so the router can classify similar prompts by similarity-weighted voting without a classifier model.

```
$ curl -X POST http://localhost:8080/api/router/my-router/corpus
\
  -H 'Authorization: Bearer <token>' \
  -H 'Content-Type: application/json' \
  -d '{"text": "What is my account balance?", "label":
"finance"}'
```

— Runnable curl example plus exact endpoints and config key v4.9.0

02 Deny-by-default authentication for all routes BREAKING

85

Every HTTP route now requires credentials unless explicitly listed in a public registry, closing bypasses for previously unauthenticated aliases such as `/moderations`, `/models`, `/backends`, and `/mcp/chat/completions`. With database auth or legacy API keys configured, `/version` and generated audio, image, video, and 3D asset URLs also now require credentials by default; narrow public paths can be restored via `ApplicationConfig.PathWithoutAuth`.

— Names affected endpoints and the exact config key to fix them v4.9.0

03 **MiniMax-H3 video and audio generation** NEW 85

Adds MiniMax-H3 video+audio generation to the `vllm-cpp` backend (ABI v12): a second engine handle loads the H3 checkpoint set, with `parameters.model` naming the DiT and the text encoder and two VAEs named under `options:`. `GenerateVideo` renders jointly into an MP4 with a real AAC audio track.

— Named config keys but no complete runnable snippet v4.9.0

04 **Shared UDP mux for Realtime WebRTC** NEW 80

A new `--web-rtc-udp-port` CLI flag and `LOCALAI_WEBRTC_UDP_PORT` environment variable let all Realtime WebRTC calls reuse a single shared Pion ICE UDP mux, simplifying container and firewall setup by needing only one open UDP port for every call.

Expose a single fixed UDP port for all Realtime WebRTC sessions so one firewall rule covers all calls.

```
$ docker run -ti --name local-ai -p 8080:8080 -p 3478:3478/udp -
e LOCALAI_WEBRTC_UDP_PORT=3478 localai/localai:latest
```

— Exact flag, env var, and a runnable docker command v4.9.0

05 **PII pseudonymization with reversal** NEW 70

An opt-in `pii.reverse_in_response` config turns masked PII values into request-scoped deterministic pseudonyms (e.g. `EMAIL_001`, `EMAIL_002`) that are restored when the backend echoes them back, including across SSE tokens split over multiple writes.

— Named config key and pattern but no runnable example v4.9.0

06 **Chat context compression** NEW 65

An opt-in per-model `compression` config compresses older complete conversation turns through a configured LocalAI model before inference, while preserving system prompts, the newest messages, and whole tool-call/result units intact. The resulting compression ratio and duration are returned as response metadata and metrics.

— Names config key but gives no exact metadata field names v4.9.0

07 **Expanded vllm-cpp CUDA architecture coverage** IMPROVED 65

`vllm-cpp` CUDA architecture coverage on amd64 grows from `120a;121a` to `80;86;89;90a;100a;103a;120a;121a`, and on arm64 from `121a` to

`87;90a;100a;110;121a`, adding support for A100, L4, 4090, H100/H200, B200, Jetson Orin, and Jetson Thor.

— Exact before/after architecture lists but no user action v4.9.0

08 **Unified models and backends UI** IMPROVED 65

`/app/models` now owns the Explore and Installed views, `/app/backends` owns the Catalog and Installed views, and `/app/manage` redirects to the new pages while preserving legacy query state.

— Exact navigation paths given, but limited behavioural detail v4.9.0

09 **Qwen3-TTS in llama-cpp backend** NEW 60

Adds Qwen3-TTS support to the `llama-cpp` backend across CUDA, ROCm, SYCL, Vulkan, Metal, and L4T using upstream GGUF conversion, with gallery entries `qwen3-tts-llamacpp` and `qwen3-tts-llamacpp-q4` provided.

— Named gallery entries give a starting point, little mechanism v4.9.0

THINNER COVERAGE BELOW

10 **85 new model gallery entries** NEW 55

Adds 85 new model gallery entries including Qwen3 (9B, 27B), Gemma 4 Scotoma 2, DeepSeek V4 Pro 0813, Nemotron 3.5 Lightning 30B, MiniMax-H3 Ref2VA (`minimax-h3-fl2va-q4` , `minimax-h3-ref2va-q4`), Higgs Audio v3 TTS, and others.

— Names specific gallery ids as usable starting points v4.9.0

11 **Admission control and backend activity visibility** NEW 50

Adds global process-wide HTTP admission control bounding in-flight backend operations, with running backend traces now visible in the UI alongside direct log links.

— UI location shown but no config key for the limit itself v4.9.0

12 **Durable job-based cold model loading** IMPROVED 50

Cold model loads now run as durable jobs rather than holding the per-model advisory lock across multi-GB transfers, preventing loads from appearing permanently broken during staging.

— Explains the fix but gives no user-facing control v4.9.0

13 **Portuguese and Indonesian UI translations** NEW 45

Adds Portuguese (Brazil) (`pt-BR`) UI translation with full 14-namespace key parity, and an Indonesian translation covering admin, media, and navigation surfaces.

— Names locales and namespace count, no further mechanism `v4.9.0`

┗—▷ BREAKING ON UPGRADE

- ! With database auth or legacy API keys configured, `/version` and generated audio, image, video, and 3D asset URLs now require credentials by default due to the deny-by-default authentication change.
- ! Embedded deployments that previously relied on unauthenticated access to routes like `/moderations` , `/models` , `/backends` , or `/mcp/chat/completions` must explicitly add those paths to `ApplicationConfig.PathWithoutAuth` .

WAS THIS USEFUL? 👍 👎

◆ AI Agent Frameworks

AutoGPT

1 RELEASE · 2026-08-21 NOTES ↗

AutoGPT is the vision of accessible AI for everyone, to use and to build on. Our mission is to provide the tools, so that you can focus on what matters.

> AutoGPT Platform v0.7.2 adds a Microsoft Teams bot adapter and device code OAuth for integrations, expands expert memory management, hiring, and work-surface flows, ships new Stripe and third-party integration blocks, and turns AutoPilot transport into an explicit user choice instead of a silent default.

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

01 **Stripe Link and subscription webhook blocks** NEW 60

Adds Stripe Link wallet blocks for wallet-based payments and Stripe subscription webhook trigger blocks for automating workflows on Stripe subscription events.

— Names two specific block types usable in workflows `autogpt-platform-beta-v0.7.2`

02 **Expert memory management** NEW 60

Adds a user-facing memory settings page for managing what each expert remembers, isolated expert memory so each expert's memory store is kept separate from others, and an expert memory admin viewer for inspecting and managing per-expert memory as an administrator.

— Three concrete surfaces described but no exact UI paths `autogpt-platform-beta-v0.7.2`

03 **All Quiet and DataForB2B integration blocks** NEW 60

Adds All Quiet incident management blocks for integrating AutoGPT workflows with All Quiet alerting, and a DataForB2B provider block for B2B data enrichment within workflows.

— Names both providers but not block configuration details `autogpt-platform-beta-v0.7.2`

04 **Raise your own expert v1** NEW 60

Adds 'Raise your own expert v1', a flow for creating custom experts with attachments, budget, and skills configuration.

— Names configurable fields but no exact UI navigation `autogpt-platform-beta-v0.7.2`

05 **Expert hiring flow enhancements** NEW 60

Adds a day-one kickoff flow that activates automatically after hiring an expert, writing-style capture during the hire flow so a new expert can match a user's communication

style, and a launch roster with real workflow bundles for bootstrapping expert assignments.

— Three named hiring-flow additions described concretely `autogpt-platform-beta-v0.7.2`

THINNER COVERAGE BELOW

06 **Team home and sidebar UI additions** NEW 55

Adds per-expert spend tracking displayed on home team cards, sidebar team-member presence indicators, and a create menu for quick expert actions.

— Names UI surfaces but not exact navigation paths `autogpt-platform-beta-v0.7.2`

07 **Device code OAuth flow for integrations** NEW 45

Adds a device code OAuth flow for authenticating integrations that cannot use a browser redirect.

— Explains mechanism but no config or endpoint named `autogpt-platform-beta-v0.7.2`

08 **AutoPilot transport becomes explicit choice** BREAKING 45

Makes the AutoPilot transport an explicit user choice rather than an automatic default.

— States before/after behaviour change clearly `autogpt-platform-beta-v0.7.2`

09 **Microsoft Teams bot adapter** NEW 40

Adds a Microsoft Teams bot adapter, enabling AutoGPT agents to send and receive messages through Microsoft Teams.

— Names the integration but no setup steps given `autogpt-platform-beta-v0.7.2`

10 **Pods v1 expert groups** NEW 40

Adds Pods v1 — named expert groups for organizing experts into labeled collections.

— Names the feature but not how to create or manage pods `autogpt-platform-beta-v0.7.2`

11 **Expert work surface v1** NEW 40

Adds expert work surface v1 with work cards and a dedicated work tab per expert.

— Names the surface but not its full behaviour `autogpt-platform-beta-v0.7.2`

12 **Copilot tool chain UI** NEW 30

Adds a copilot tool chain UI for composing and managing copilot operations.

— Thin description with no operational detail `autogpt-platform-beta-v0.7.2`

13 **AI-voice narrative in morning briefing** NEW 20

Adds AI-voice narrative in the morning briefing.

— Bare description with no mechanism or scope `autogpt-platform-beta-v0.7.2`

WAS THIS USEFUL? 👍 👎

LangChain ⓘ 2 RELEASES · 2026-08-20 NOTES ↗

The agent engineering platform.

> LangChain shipped a small pair of updates this window: document reranking support in the Fireworks integration, and a customizable token counter for context editing middleware.

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

THINNER COVERAGE BELOW

01 Custom token_counter in ContextEditingMiddleware IMPROVED

50

`langchain` 1.3.16 supports a custom `token_counter` argument in `ContextEditingMiddleware` for fine-grained token counting control.

— Names exact argument and class, but no usage example. `langchain==1.3.16`

02 Document reranking in Fireworks integration NEW

35

`langchain-fireworks` 1.6.0 adds document reranking support to the Fireworks integration.

— Names the capability but no API details or mechanism. `langchain-fireworks==1.6.0`

WAS THIS USEFUL? 👍 👎

Agno (formerly Phidata)

1 RELEASE · 2026-08-20 NOTES ↗

Build, run, and manage agent platforms.

> Agno's v3.0.0a2 alpha ships a new FinanceTools toolkit, a governed Studio 3.0 control plane, and reliable background execution for AgentOS, alongside a wide set of breaking changes to session/memory params, HITL configuration, and Team/Workflow constructors.

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

01 **Removed session and memory parameters** BREAKING 75

Removes the `enable_user_memories`, `search_session_history`, `num_history_sessions`, and `num_past_session_runs` parameters; any code passing these will break on upgrade.

— Lists all four removed params for direct migration. v3.0.0a2

02 **Human-in-the-loop config via HumanReview** BREAKING 70

Human-in-the-loop configuration now requires human_review=HumanReview(...), replacing the previous flat HITL kwargs, which are removed and will break existing code on upgrade.

— Gives exact before/after syntax for migration. v3.0.0a2

03 **Reasoning shortcut removed** BREAKING 60

The `reasoning=True` shortcut is removed; callers must now pass an explicit `reasoning_model` argument.

— Names old and new argument for direct code fix. v3.0.0a2

THINNER COVERAGE BELOW

04 **FinanceTools unified finance toolkit** NEW 55

Adds `FinanceTools`, a unified finance toolkit with swappable data providers, replacing scattered individual finance tool integrations.

— Names the toolkit but not the providers or config. v3.0.0a2

05 **Keyword-only Team and Workflow constructors** BREAKING 55

Makes `Team` and `Workflow` constructors keyword-only, enforcing explicit argument passing; positional arguments will now raise errors on upgrade.

— Names both constructors and the upgrade impact. v3.0.0a2

06 **User isolation for evals, schedules, knowledge and metrics** IMPROVED 50

Adds user-isolation to evals, so evaluation runs are scoped per user, and extends the same isolation to schedules, metrics, knowledge, and vector DB resources.

— Names five scoped areas but no config keys or mechanism. v3.0.0a2

07 **Reliable background execution for AgentOS** NEW 45

Introduces reliable background execution for AgentOS — bounded, observable, and durable agent job processing.

— Describes properties but no API or config to invoke it. v3.0.0a2

08 **Toolkit id field** NEW 40

Adds an `id` field to `Toolkit`, allowing toolkits to be referenced and tracked by identifier.

— Names the field and purpose but no usage example. v3.0.0a2

09 **Improved agno create onboarding** IMPROVED 40

Improves the `agno create` onboarding flow for new platform setup.

— Names the command but not what specifically changed. v3.0.0a2

10 **RampRouter model class** NEW 35

Adds `RampRouter` model class for the Ramp Router (router.com) provider.

— Bare name of a new model class, no usage shown. v3.0.0a2

11 **Denormalized sessions table** IMPROVED 35

Denormalizes the sessions table in the database for improved query performance.

— States the change and benefit but no schema or metrics. v3.0.0a2

12 **Studio 3.0 control plane** NEW 30

Introduces Studio 3.0, a governed control plane for agents that build agents.

— Conceptual description only, no navigation or usage. v3.0.0a2

13 **Consolidated AgentOS metadata routes** IMPROVED 30

Consolidates AgentOS metadata routes into a unified surface.

— No endpoint names or before/after route list given. v3.0.0a2

14 **MistralAI v1 compatibility layer removed** BREAKING 30

The MistralAI v1 compatibility layer is removed; code relying on it will break.

— Names the layer but gives no migration guidance. v3.0.0a2

15 **Culture feature removed** BREAKING 25

The `culture` feature (experimental) is removed with no replacement.

— States removal only, no replacement or migration path. v3.0.0a2

16 **Deprecated API surface removed** BREAKING 10

Removes the deprecated v3.0 API surface and an unannounced compatibility surface.

— No specifics on which endpoints or surfaces are affected. v3.0.0a2

└──▷ BREAKING ON UPGRADE

! The `enable_user_memories`, `search_session_history`, `num_history_sessions`, and `num_past_session_runs` parameters are removed; any code passing these will break on upgrade.

! Flat HITL kwargs are removed; callers must now pass `human_review=HumanReview(...)` instead.

! The `reasoning=True` shortcut is removed; callers must now pass an explicit `reasoning_model` argument.

! The `culture` feature (experimental) is removed with no replacement.

! The MistralAI v1 compatibility layer is removed; code relying on it will break.

! `Team` and `Workflow` constructors are now keyword-only; positional arguments will raise errors on upgrade.

! Deprecated v3.0 API surface and unannounced compat surface are removed.

OTHER / UNCATEGORIZED

◆ AI OBSERVABILITY

LangChain LangSmith

1 RELEASE · 2026-08-10 NOTES ↗

LangSmith is a platform for debugging, testing, and monitoring LangChain applications and language model workflows in production.

> LangSmith added a validation endpoint for testing thread evaluators, switched bulk export compression to zstd by default, and removed legacy dataset comparison SDK helpers in favor of a new experiment-runs endpoint, alongside OpenTelemetry span-ordering fixes, resource-attribute tracing, and several run-ingestion reliability improvements.

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

01 **Thread evaluator test validation endpoint** NEW 88

The `/runs/rules/validate` endpoint now supports testing a multi-turn thread evaluator against a real conversation, using `test_thread_id` and `session_id` parameters, before saving the evaluator rule.

Test a multi-turn thread evaluator against a real conversation before saving it, using the updated `/runs/rules/validate` endpoint.

```
$ curl -X POST 'https://<your-langsmith-
host>/runs/rules/validate' \
  -H 'Content-Type: application/json' \
  -H 'X-API-Key: <api-key>' \
  -d '{"test_thread_id": "<thread-id>", "session_id": "<session-
id>"}'
```

— Runnable curl against a named endpoint with exact parameters. snapshot-20260821

02 **zstd default for bulk export compression** BREAKING 84

Bulk export compression now defaults to `zstd` instead of `gzip` on cloud deployments; self-hosted environments retain the `gzip` default via the `FF_BULK_EXPORT_DEFAULT_COMPRESSION` environment variable.

Retain gzip compression for bulk exports on a self-hosted LangSmith deployment instead of the new zstd default.

```
$ export FF_BULK_EXPORT_DEFAULT_COMPRESSION=gzip
```

— Names default change and the exact env var to retain old behavior. snapshot-20260821

03 **Legacy dataset comparison helpers removed** BREAKING 84

Legacy dataset comparison helpers are removed from the public OpenAPI spec and generated SDKs; code using those SDK methods must migrate to `POST /v2/datasets/{dataset_id}/experiment-runs`. Existing HTTP routes continue to work for LangSmith UI clients.

— Gives the exact replacement endpoint to migrate to. snapshot-20260821

04 **OpenTelemetry resource attributes on traces** NEW 81

Setting `OTEL_RESOURCE_ATTRIBUTES` (e.g. `user.id`, `deployment.environment`) now surfaces those values on LangSmith traces under `otel.resource.*` fields without modifying span emission.

Attach OpenTelemetry resource attributes (e.g. user ID, environment) so they appear on LangSmith traces under `otel.resource.*` without modifying span emission.

```
$ export
OTEL_RESOURCE_ATTRIBUTES="user.id=usr_123,deployment.environment
=production"
```

— Exact env var and resulting field naming, runnable as-is. snapshot-20260821

05 **run_verbs log format change** BREAKING 67

The batched-run ingestion log now emits `run_verbs` as a list of `run_id` and `verbs` objects instead of a map keyed by run UUID; structured-log aggregators parsing the old map format need to be updated.

— Names the exact field and structure change, no migration command. snapshot-20260821

THINNER COVERAGE BELOW

06 **OpenTelemetry span ordering fix** IMPROVED 54

Native OpenTelemetry child spans that arrive before their SDK-attributed parent span are no longer dropped; they are now buffered and correctly nested regardless of arrival order.

— Automatic backend fix; nothing for a user to configure. snapshot-20260821

07 **Oversized field handling in multipart run ingestion** IMPROVED 48

LangSmith now preserves traces in multipart ingestion batches when one run has oversized inputs or outputs, replacing the oversized fields with a placeholder instead of rejecting the entire batch.

— Clear before/after behavior but no user-facing control. `snapshot-20260821`

---

08 **Dataset split chips in Examples table** IMPROVED 45

Each example's dataset splits now render as chips in the dataset Examples table, with a clickable overflow menu for examples belonging to multiple splits.

— Clear UI location but purely cosmetic. `snapshot-20260821`

---

09 **Clearer 409 Conflict messages for duplicate runs** IMPROVED 45

Duplicate run create or update payloads now return clearer 409 Conflict messages indicating whether the duplicate was a create or update request.

— Names the status code and distinction but no example payload. `snapshot-20260821`

---

10 **Streaming thread stats ordering** IMPROVED 42

Thread stats requests that opt into streaming now return main stats first and append feedback stats when ready, rather than waiting for both together.

— Behavior change described but no interface to act on. `snapshot-20260821`

---

11 **Error recording for failing custom code evaluators** IMPROVED 42

Custom code evaluators that time out or fail now record an error on the run instead of silently leaving it without feedback.

— Behavior change stated but no config or code surface given. `snapshot-20260821`

---

12 **Timeout banner in runs table** IMPROVED 33

When a runs query times out, the runs table now shows a timeout banner instead of failing silently.

— Small UI signal with minimal named detail. `snapshot-20260821`

└──▷ BREAKING ON UPGRADE

! Legacy dataset comparison helpers are removed from the public OpenAPI spec and generated SDKs; code using those SDK methods must migrate to `POST /v2/datasets/{dataset_id}/experiment-runs`.

! Bulk export compression `gzip` to `zstd` on cloud deployments; set `FF_BULK_EXPORT_DEFAULT_COMPRESSION` to retain `gzip` on self-hosted environments.

defaults changed from

! The batched-run ingestion log now emits `run_verbs` as a list of `run_id` and `verbs` objects instead of a map keyed by run UUID; structured-log aggregators parsing the old map format will need to be updated.

PromptLayer ⓘ 1 RELEASE · seen 2026-08-21 NOTES ↗

PromptLayer is a platform that logs, manages, and analyzes LLM API calls for debugging and optimization.

> PromptLayer's trace view gets richer with automatic input/output previews, a redesigned tabbed span details panel, and entity highlighting, alongside new support for Gemini 3.7 Flash.

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

01 **Trace input/output previews** NEW 60

The trace detail view now shows input/output previews at the top level, automatically extracted from the first user message and final assistant response, displayed above the span tree. This is especially useful for multi-step traces involving tool calls.

– Names mechanism and location but no config/API surface `snapshot-20260821`

02 **Redesigned span details panel with tabs** IMPROVED 60

The span details panel is redesigned with a tabbed interface separating metadata, inputs/outputs, and entity references, plus hover cards that let users preview a span quickly without opening the full panel.

– Clear UI mechanism described, no deeper config detail `snapshot-20260821`

THINNER COVERAGE BELOW

03 **Gemini 3.7 Flash model support** NEW 45

Adds support for the Gemini 3.7 Flash model with updated pricing and reasoning capabilities.

– Names model but no usage detail or limits `snapshot-20260821`

04 **Trace entity highlighting** NEW 35

Trace entities such as prompts, workflows, and evaluations are now highlighted in the trace view to provide better context.

– Brief description without mechanism or scope details `snapshot-20260821`

WAS THIS USEFUL? 👍 👎

Langfuse

1 RELEASE · 2026-08-21 NOTES ↗

Open source AI engineering platform: LLM evals, observability, metrics, prompt management, playground, datasets. Integrates with OpenTelemetry, LangChain, OpenAI SDK, LiteLLM, and more. YC W23

> Langfuse v4.16.0 brings its Ask AI assistant to self-hosted deployments, adds more filtering to session span lists, and extends signup attribution tracking to additional ad platforms.

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

THINNER COVERAGE BELOW

01 **Expanded signup attribution tracking** IMPROVED 40

Extends signup attribution tracking to capture LinkedIn, Reddit, and X click IDs.

– Names specific platforms but no config or usage detail v4.16.0

02 **Ask AI on self-hosted instances** NEW 35

Ask AI support is now enabled on self-hosted Langfuse deployments, not just Langfuse Cloud.

– States the change but no mechanism or setup detail v4.16.0

03 **More filters in session span lists** IMPROVED 30

Adds more filter options in the span list within session views.

– Bare description, no named filter types or UI path v4.16.0

WAS THIS USEFUL? 👍 👎

Braintrust

1 RELEASE · 2026-08-01 NOTES ↗

Braintrust is an AI evaluation platform that helps developers test, benchmark, and monitor AI applications and models in production.

> Braintrust's August 2026 release adds write access to its MCP server, two new built-in open-source models, an AWS Lambda extension for lower-latency tracing, and broad SDK instrumentation and trace-grouping improvements, alongside several breaking changes to Go SDK span formats.

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

01 Breaking span-format changes across Go SDK provider integrations

90

BREAKING

Go SDK v0.11.x changes trace span shapes across several provider integrations. For Google GenAI, provider metadata changed from `'gemini'` to `'google'`. For Eino, `ChatModel` span output is now an OpenAI-compatible choices array (`[{"index": 0, "finish_reason": "...", "message": {...}}]`) instead of a flat message map; embedding input is now `{"inputs": [{"content": "..."}]}` and output is `{"count": N}`, removing `embedding_length` and renaming `embeddings_count`, and provider metadata is now lowercase (e.g. `'openai'` instead of `'OpenAI'`). For Anthropic, span metadata no longer includes `endpoint`, the `output` field is now a single message object instead of an array, and non-streaming spans no longer emit `time_to_first_token`. For Bedrock, span metadata renames `stop_sequences` to `stop`, removes `additional_model_request_fields`, and aligns image, document, and tool block shapes with Bedrock's native wire format. Trace queries relying on the previous field names or formats need updating.

— Enumerates every renamed/removed field across four providers verbatim. snapshot-20260821

02 Firebase Genkit middleware options and traced tool wrappers in Go SDK

NEW

80

Go SDK v0.11.1 adds `WithProvider` and `WithModel` options on `NewMiddleware` for explicit model attribution in Firebase Genkit, plus traced tool wrappers `DefineTool`, `DefineToolWithInputSchema`, and `DefineMultipartTool`; auto-instrumentation now replaces `genkit.DefineTool`, `genkit.DefineToolWithInputSchema`, and `genkit.DefineMultipartTool` calls with their traced equivalents.

— Names every function and option added, verbatim. snapshot-20260821

03 Write tools in the Braintrust MCP server NEW

75

The Braintrust MCP server now exposes write tools so coding agents can create and update prompts, scorers, classifiers, monitor views, alerts, scheduled jobs, dataset rows, and Topics pipeline configuration, not just read them. Write tools operate under the authenticated account's permissions and several can replace or remove objects, so Braintrust recommends configuring the client to require confirmation before use.

— Names every writable object type and the permission model. `snapshot-20260821`

---

04 **New built-in open-source models kimi-k3 and deepseek-v4-flash-0731**

NEW

70

Adds `kimi-k3` and `deepseek-v4-flash-0731` as built-in open-source models served directly by Braintrust with no AI-provider setup required. They can be selected under the Braintrust provider in playgrounds, prompts, and scorers, or requested by name through the Braintrust Gateway; usage draws from monthly model credits shared with Topics.

— Names both models and the shared-credit mechanism. `snapshot-20260821`

---

05 **New auto-instrumentation targets in Python SDK v0.33.0** NEW

70

Python SDK v0.33.0 adds Vercel AI SDK for Python auto-instrumentation, enabled by default in auto_instrument(), and Cursor SDK Python instrumentation for tracing agent runs, model turns, and tool calls.

Enable Vercel AI SDK auto-instrumentation in a Python service to trace all AI calls without manual span creation.

python

```python
import braintrust

braintrust.auto_instrument()  # Vercel AI SDK for Python instrumented by default in v0.33.0+
```

— Includes a runnable code example enabling the feature. `snapshot-20260821`

---

06 **Automatic conversation grouping for multi-turn traces** IMPROVED

65

Traces that share a `metadata.conversation_id` now surface as related traces automatically for end-to-end multi-turn conversation review without requiring grouping configuration. The 'Group by' and 'Cluster by' controls have moved into the row-type selector in the toolbar.

— Names the metadata field and the moved UI controls. `snapshot-20260821`

---

07 **Trace-group references in dataset rows** NEW

60

Adds trace-group references to dataset rows so a multi-turn session or related log set becomes one example, with up to 64 traces per row rendered inline.

— Gives concrete per-row trace limit of 64. `snapshot-20260821`

THINNER COVERAGE BELOW

08 Braintrust Lambda Extension for Python and TypeScript NEW

55

Adds the Braintrust Lambda Extension for Python and TypeScript/JavaScript Lambda functions, giving the Braintrust SDK's flush() method a local handoff path for traces to reduce latency in the request path.

— Names the mechanism but not install steps. `snapshot-20260821`

09 SDK version context on Ruby spans NEW

30

Ruby SDK v0.4.1 adds `braintrust.context_json` carrying SDK name and version to all spans.

— Names the field but describes no further use. `snapshot-20260821`

└──▷ BREAKING ON UPGRADE

! Python SDK v0.32.0: LiveKit Agents audio attachments on `agent_speaking` spans are now disabled by default. Set `BRAINTRUST_CAPTURE_AGENT_AUDIO_ATTACHMENTS=true` to restore the previous behavior.

! Go SDK v0.11.1 (Google GenAI): Provider metadata changed from `'gemini'` to `'google'`. Update trace queries that filter on the previous provider value.

! Go SDK v0.11.1 (Eino): `ChatModel` span output is now an OpenAI-compatible choices array (`[{"index":0,"finish_re`) instead of a flat message map. Embedding input is now `{"inputs":[{"content":` and output is `{"count":N}` , removing `embedding_length` and renaming `embeddings_count` . Provider metadata is now lowercase (e.g.

```
a
s
o
n
"
:
"
.
.
.
"
,
"
m
e
s
s
a
g
e
"
:
{
.
.
.
}
}
]
"
.
.
.
"
}
]
}
```

! Go SDK v0.11.0 (Anthropic): Span metadata no longer includes `endpoint`. The `output` field is now a single message object instead of an array. Non-streaming spans no longer emit `time_to_first_token`.

! Go SDK v0.11.0 (Bedrock): Span metadata renames `stop_sequences` to `stop` and removes `additional_model_request_fields`. Image, document, and tool block shapes now align with Bedrock's native wire format.

WAS THIS USEFUL? 👍 👎

◆ VECTOR DB RAG

Pinecone ⓘ 1 RELEASE · 2026-08-01 NOTES ↗

Pinecone is a vector database service that stores and searches high-dimensional embeddings for AI and machine learning applications.

> Pinecone added Terraform support for managing organization membership, letting teams handle invites and removals as code.

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

THINNER COVERAGE BELOW

01 **Terraform resources for organization membership** NEW

45

New Terraform resources allow managing organization invites and removing organization members directly through Terraform configuration.

— Names Terraform and the two operations but no resource identifiers. snapshot-20260821

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · August 21, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe