THE AI TOOLCHAIN

NO. 050

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED OCTOBER 8, 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 142 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 38 of 39 tools

▾ // THE BRIEF HIDE

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

Three unrelated things stand out. smolagents now swaps the sandbox a CodeAgent executes in — local, e2b, modal, docker — behind one ace, and can save or load a whole agent from the Hub. Cline moved

vim ? help

coding sessions into isolated cloud sandboxes that land as GitHub PRs. Braintrust's `cancel_event` kills a runaway eval mid-flight instead of waiting out the bill.

BUILD

Move an agent's code execution off your laptop without rewriting the agent

smolagents makes the CodeAgent executor pluggable across local, blaxel, e2b, modal and docker, so the same agent runs in a throwaway container in CI and on a dev box with one config change; Cline's Cloud Agents go further and run the whole coding session in a browser-launched sandbox that emits a GitHub branch and pull request. Both remove the step where you hand-build an isolated runner before letting a model run generated code.

smolagents · Cline

BUILD

Carry an agent's accumulated preferences and lessons into the next session

Devin Desktop's Memory is Git-backed and consolidated in the background, so the corrections you give an agent survive the session rather than being re-typed each morning, and admins can govern what gets retained. Agent Memory replaces its GLiNER v1/GLiREL extraction stack with a GLiNER2.5 joint extractor and ships an installable metrics module, so you can measure whether the entities and relations going into the memory graph are actually right.

Devin Desktop · Agent Memory

EVALUATE

Stop a bad eval run before it finishes paying for itself

A `cancel_event` mechanism halts a running evaluation, which matters because a misconfigured scorer or a wrong model on a large dataset otherwise burns the full token spend before anyone can look at it. SQL-based dashboard charts and expiring API keys land in the same release.

Braintrust

DEPLOY

Ship one quantized export to llama.cpp-class runtimes from the same pipeline

> The unified HF export adds three new GGML quantization formats, so a model quantized for GPU serving and a build headed for CPU or local execution come out of one toolchain instead of a separate conversion script. Megatron and fused-MoE export paths now carry documented constraints, so you find the unsupported case in the docs rather than at export time.
>
> NVIDIA Model Optimizer

BUILD

◆ AI Coding Agents

Augment Code

Feedback ⓘ snapshot-20261008 NOTES ↗

Your engineers have agents. Your organization doesn't.

Outcome Analytics exits beta with CSV export, drill-down sessions, and custom metric columns; adds reasoning effort selection for models and Experts.

Outcome Analytics exits beta and is open to everyone: results on the Outcomes tab are grouped by category, support custom metric columns, trend views for individual outcomes, and export to CSV.

Adds search, filter, and export to CSV for Outcome Analytics results, with drill-down into the sessions behind each outcome.

Adds a reasoning effort level selector when choosing a model for sessions and Experts.

Adds thumbs-up / thumbs-down rating on any completed agent response, with reason codes such as `Didn't follow my instructions` or `Too slow or stuck`.

Automations now show who created each trigger, when it was last updated, and run history shows who owns each session.

+5 more

NEW INTEGRATION 28 new vendor-hosted MCP servers

NEW INTEGRATION Self-hosted GitLab and Apollo OAuth providers

WAS THIS USEFUL? 👍 👎

yetone magpie

Feedback ⓘ v0.1.1117 CODE ↗ RANK

A macOS menu-bar app, TUI, and CLI that lists every AI coding agent on the machine and switches each one's model, such as Codex on DeepSeek or Claude Code on Kimi, by rewriting only that key in the agent's own config.

> This window magpie added Markdown export of full session conversations, Codex config.toml import with per-agent config discovery, `${NAME}` env-var placeholders for MCP server configs, and MSIX-packaged Claude Desktop support on Windows, alongside a string of routing, structured-output, and provider-specific fixes.

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

01 ${NAME} env-var placeholders in MCP headers and command env

NEW

85

Supports `${NAME}` placeholders in a remote MCP server's `headers` and a command's `env` fields; the placeholder is kept as written and rewritten into each agent's own syntax at delivery time, so tokens stay in the environment and never appear in any agent's config file.

— Names exact placeholder syntax and security mechanism v0.1.1112

02 MSIX-packaged Claude Desktop support on Windows NEW

85

Supports Claude Desktop installed as an MSIX package on Windows (e.g. Windows 10 LTSC), discovering it in `%LOCALAPPDATA%\Packages\Claude_<publisher>\LocalCache` and wiring MCP servers, skills, and Cowork session reads from the package's own folders.

— Names exact discovery path and wired surfaces v0.1.1111

03 Codex provider model listing and tier handling

UPDATED INTEGRATION IMPROVED

80

Passes Codex's Ultrafast tier (`service_tier` `ultrafast`) through to the ChatGPT backend when magpie's Codex account pool serves the turn, matching Codex's own sign-in behavior; the Codex provider now lists every ChatGPT account's models across all plan tiers rather than only those available to the currently signed-in account; and the Codex panel now warns when the configured model list exceeds the Codex app menu's limit of 100 models, advising users to reorder or trim the list.

— Names tier field, limit number, and scope of fix v0.1.1110 v0.1.1113 v0.1.1114

04 Markdown export for session conversations NEW

75

Exports a session's full conversation as a Markdown file via `WriteMarkdown` in `internal/sessions/markdown.go`, triggered from the Sessions page's details view and saved to Downloads; includes all turns, tool calls/results, thinking blocks, and agent metadata.

— Names function, file path, and exact save destination v0.1.1116

05 **Structured output translation across upstream targets** IMPROVED

75

Structured output (`text.format` / `response_format`) now translates correctly across all upstream targets — Chat, Responses, Messages, and Gemini — so a Responses `text.format` or Chat `response_format` reaches any upstream in its own format.

— Names exact fields and all four upstream targets v0.1.1110

06 **Qoder campaign window persistence fix** IMPROVED 70

Persists the Qoder campaign `until` field so successful check-in results are reused until the campaign's `[startAt, endAt)` window expires, allowing campaigns that run past Beijing midnight to be claimed and the next one to open correctly.

— Names exact field and window semantics fixed v0.1.1111

07 **Routing page filtering and provider-mode labeling** IMPROVED

65

Adds a multi-checkbox purpose filter to the Routing page, letting users show requests matching any combination of purposes (e.g. Memory and Title) simultaneously, and fixes the Routing page to label weight-routed providers as By weight instead of Smart, including the mode line under the stage heading.

— Thin UI additions to same page, both named v0.1.1113 v0.1.1110

08 **Copilot CLI sign-in fallback on token expiry** NEW INTEGRATION IMPROVED 65

Copilot provider now falls back to the Copilot CLI's sign-in (via `COPILOT_HOME`) when the editor token GitHub returns 401, keeping the same-account session alive.

— Names env var and exact failure condition v0.1.1110

09 **Routing-group reasoning flag fix** IMPROVED 65

Fixes routing-group reasoning so a group reports `reasoning: true` when ANY member supports reasoning, not only when ALL members do, preventing a non-reasoning member from silently disabling thinking for the whole group.

— Names exact flag and before/after behavior v0.1.1112

10 **MiniMax thinking routed as folded reasoning in Codex** IMPROVED

60

Routes MiniMax's thinking content, delivered between `<think>` tags in MiniMax-M2 streams, to Codex as folded reasoning rather than as answer text.

— Mechanism named, but no user-facing action to take v0.1.1114

THINNER COVERAGE BELOW

11 **Synthetic user turn fix for Gemini/Code Assist** IMPROVED

55

Inserts a synthetic user turn before model-first histories so Gemini and Code Assist no longer reject them with HTTP 400.

— Names exact error fixed and affected backends v0.1.1114

12 **Parallel tool-call result preservation across backends** IMPROVED

50

Preserves parallel tool-call results split across two consecutive assistant messages on Chat, Cursor, and Devin backends.

— Names affected backends but no mechanism depth v0.1.1114

13 **Per-key proxy assignment for provider API keys** NEW

50

Enables each provider API key to be assigned its own proxy, independent of other keys on the same provider.

— Clear capability but no config key or UI path given v0.1.1113

14 **Daily warm-up scheduling** NEW

45

Adds a Daily warm-up feature that can run several times a day, each run starting its own 5-hour window.

— Describes behavior but no setup detail v0.1.1113

NEW INTEGRATION Ante (antigma.ai) provider integration

UPDATED INTEGRATION Grok root-union tool-schema folding for relay and xAI API

UPDATED INTEGRATION Library MCP servers extended to WorkBuddy and CodeBuddy Code

NEW SUPPORTED PLATFORM Traditional Chinese (zh-TW) localisation

v0.1.1117 CODE ↗ LATEST STABLE

Adds per-model `+model` / `-model` CLI flags, group-name routing, unexposed-model diagnostics, and JetBrains Copilot BYOK wiring.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v0.1.1117 https://github.com/yetone/magpie.git
# already have the repo? check out this version:
$ git checkout v0.1.1117
```

Adds `magpie provider models <id> +<model>` and `magpie provider models <id> -<model>` subcommands to add or remove a single model from a provider's exposed list without replacing the whole list.

Adds display of unexposed models in `magpie models` and `magpie provider <id>` output, naming which models a provider's list has that are not exposed and why (not picked, or only first entries with none picked).

Serves a routing group by its name in addition to its model list, so a client configured with a renamed group's name reaches it rather than receiving a 404 error.

v0.1.1115 CODE ↗ STABLE

Adds WSL agent detection toggle under Settings → General; when off, magpie skips all wsl.exe queries and hides WSL agents and sessions.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v0.1.1115 https://github.com/yetone/magpie.git
# already have the repo? check out this version:
$ git checkout v0.1.1115
```

v0.1.1107 CODE ↗ STABLE

Adds context-window inspection per request/session/agent, a `magpie-quota` official skill, intent-based routing rules via Jev classifier, and a GLM reset-counter card.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v0.1.1107 https://github.com/yetone/magpie.git
# already have the repo? check out this version:
$ git checkout v0.1.1107
```

Adds a `magpie-quota` skill to the skills market — marked official and listed first — so an agent can read remaining quota across every subscription, plan, and key via `magpie quota --json`.

Adds context-window inspection for every request, session, and agent: shows prompt composition by part (`system` , `tools` , `memory` , `files` , `results` , `chat`), fill rate,

and vendor cache reads.

Adds a Context tab to the Usage page, scoring each agent's window utilisation, tagging it, and listing sessions' fill turn by turn with per-request drill-down into Routing.

Adds a `Route by what it asks for` intent-based rule option to routing groups, using Jev as the classifier with a TypeSafe provider.

Adds GLM Coding Plan key card display of reset counters (five-hour and weekly) sourced from `customer-package-reset/list` (`targetType=PERSONAL`).

WAS THIS USEFUL?  

Cline

Feedback ⓘ cli-v3.0.70 NOTES ↗ CODE ↗ RANK

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

> Cline added non-interactive CLI management for MCP servers and launched Cloud Agents, a new browser-based way to run isolated, sandboxed coding sessions that land their work as GitHub branches and pull requests.

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

01 **Non-interactive CLI for MCP server management** NEW 90

Adds `CLINE_MCP_SETTINGS_PATH` env var to override the default MCP settings file at `~/.cline/data/settings/cline_mcp_settings.json`, plus new subcommands: `cline mcp add <name> <source> --transport <transport> --header <header> --yes` to register a server from a script without the interactive wizard, `cline mcp remove <name>` to remove a named server, and a list subcommand to view configured servers, all usable non-interactively for scripting or CI.

Add a local filesystem MCP server and a remote HTTP server in CI without interactive prompts.

```
$ cline mcp add files --yes -- npx -y
@modelcontextprotocol/server-filesystem
cline mcp add docs https://example.com/mcp --transport http --
header "Authorization: Bearer TOKEN" --yes
```

Point Cline at a non-default MCP settings file, e.g. for a team-shared config.

```
$ export CLINE_MCP_SETTINGS_PATH=/etc/cline/mcp_settings.json
```

Remove a stale MCP server registration from a teardown script.

```
$ cline mcp remove docs
```

— Runnable commands and env var given, fully actionable product docs

02 **Cline Cloud Agents at app.cline.bot** NEW 80

Adds `app.cline.bot` as a browser-based entry point requiring no local install, running coding tasks in isolated cloud sandboxes that persist after the tab closes. Supports up to 10 concurrent sessions and 50 new sessions per day, each with its own repo copy and a

dedicated `cline/` branch that never touches the default branch, pushing progress to GitHub continuously. Sessions support mid-session model switching, three billing modes (`ClineFree` , `ClinePass` , and usage-billing Cline credits), Plan and Act modes, per-command approvals or auto-approve, image context, and mid-task follow-ups; sandboxes pause after 15 minutes of inactivity and resume with workspace intact, and sessions are viewable/steerable from any mobile browser via a new Agents tab showing status (working, done, waiting).

— Rich mechanism and limits but only UI navigation, no commands `product docs`

NEW INTEGRATION Claude Haiku 5.5 added to model catalog

UPDATED SUPPORTED PLATFORM Default model updated for multiple providers

NEW SUPPORTED PLATFORM Claude Haiku 5.5 added to model catalog

`cli-v3.0.70` NOTES ↗ CODE ↗ LATEST STABLE

Cline CLI v3.0.70 moves `cline dashboard` under `cline hub` , adds `cline doctor log` with `CLINE_LOG_PATH` , and enforces delivery validation on `cline schedule` .

└──▷ GET THIS VERSION ▾

```
$ git clone --branch cli-v3.0.70 https://github.com/cline/cline.git
# already have the repo? check out this version:
$ git checkout cli-v3.0.70
```

Moves `cline dashboard` to `cline hub dashboard` ; the old `cline dashboard` continues to work as a hidden alias.

Adds `cline doctor log` command that opens the log file set by `CLINE_LOG_PATH` when that variable is present.

`cline hub` with no subcommand now prints its subcommand list instead of erroring.

`cline schedule create` , `update` , and `import` now reject a delivery configuration that can never be posted — e.g. `--delivery-adapter` without `--delivery-thread` , or an unknown connector — naming the flag or `--metadata-json` path to fix.

`cline mcp add` now preserves `--header` values (such as Authorization) when opening the setup wizard, asking for one header per prompt and validating each the same way as `--yes` .

└──▷ BREAKING ON UPGRADE

! `cline history --page` is now rejected as unknown option — remove it from any scripts.

! `cline schedule create` / `update` `--autonomous` , `--no-autonomous` , `--idle-timeout` , `--poll-interval` , and `--delivery-channel` are now rejected

as `unknown option` — remove them from any scripts.

`sdk/sdk/v0.0.92` NOTES ↗ CODE ↗ STABLE

`@cline/core` exports `parseMcpHeaders` for preserving `--header` values in the CLI `mcp add` wizard.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch sdk/sdk/v0.0.92
https://github.com/cline/cline.git
# already have the repo? check out this version:
$ git checkout sdk/sdk/v0.0.92
```

WAS THIS USEFUL? 👍 👎

Superset

Feedback ⓘ cli-v1.37.0 NOTES ↗ RANK

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

> Superset's CLI v1.37.0 adds a new automation trigger and overhauls its MCP server architecture, while also dropping task columns from the Linear mirror.

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

01 **One MCP server per plugin** BREAKING 45

Superset now runs one MCP server per plugin instead of a shared server, with the account passed as a tool argument.

— Explains architecture change but no migration steps given cli-v1.37.0

02 **Linear mirror drops task columns** BREAKING 45

The Linear mirror's task columns are dropped from the tasks table; fields that were previously populated are now null.

— States exact schema impact but no remediation guidance cli-v1.37.0

03 **PR-closed automation trigger** NEW 30

Automations can now be triggered when a pull request is closed, adding a new trigger type alongside existing ones.

— Names the trigger but no config or mechanism detail cli-v1.37.0

NEW INTEGRATION Neon plugin

NEW INTEGRATION Google Calendar plugin

└──▷ BREAKING ON UPGRADE

! The Linear mirror's task columns are dropped from the tasks table; fields that were previously populated are now null.

desktop-v1.37.0 NOTES ↗ LATEST STABLE

Adds voice mode (OpenAI Realtime), ACP chat on mobile, a PR closed automation trigger, `superset pages storage` CLI, and `Cmd+;` workspace switcher.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch desktop-v1.37.0 https://github.com/superset-sh/superset.git
# already have the repo? check out this version:
$ git checkout desktop-v1.37.0
```

Adds `superset pages storage` CLI command to read a page's shared storage.

Renames `workspaces delete` to `workspaces archive` in the CLI.

Adds a PR closed trigger for automations.

Adds voice mode on mobile, implemented as an orchestrator over OpenAI Realtime.

Adds ACP chat on mobile with a Cursor-style UI, behind the `acp-chat` flag.

+10 more

└──▷ BREAKING ON UPGRADE

! The `workspaces delete` CLI subcommand is renamed to `workspaces archive`; any script invoking `workspaces delete` will break.

WAS THIS USEFUL? 👍 👎

Letta Code

Feedback ⓘ v0.34.5 NOTES ↗ RANK

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

> Letta Code v0.34.5 adds always-on redaction of vault secrets, reports hardware metadata on registration, and lets workflow scripts call agent MCP tools directly.

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

01 **Always-on vault secret redaction** NEW 43

Letta Code now automatically strips secrets from tool returns and user messages, redacting vault secrets on an always-on basis rather than requiring manual configuration.

— Describes behavior but no config surface or opt-out given v0.34.5

02 **Workflow scripts can call agent MCP tools** NEW 30

Workflow scripts can now call agent MCP tools directly, per the release summary.

— Mentioned only in summary, no concrete syntax shown v0.34.5

03 **Hardware metadata reporting on registration** NEW 23

The server now reports computer hardware metadata during server registration.

— Bare description, no fields or mechanism named v0.34.5

WAS THIS USEFUL? 👍 👎

Diagram Design

Feedback

changes since 2026-09-15

CODE ↗

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

Adds marimekko, Streamgraph, heatmap, architecture delta, exploded axonometric, and axonometric plan diagram types plus a print-A3-landscape size preset.

Adds `print-a3-landscape` size preset to the output spec.

Adds marimekko as a Treemap variant.

Adds Streamgraph as a Line variant.

Adds heatmap as visual type.

Adds architecture delta diagrams.

+1 more

WAS THIS USEFUL?

Cotool

Feedback ⓘ v0.73.0 NOTES ↗

Cotool is an AI security operations platform for investigating alerts, enriching threat intelligence, and automating response workflows.

Adds suggested alert dispositions with context reuse, sub-agent delegation for every agent and chat, and Claude Haiku 5.5 with extra-high reasoning effort.

Adds suggested dispositions to escalated alerts: confirm a statement about your environment from the alert page or Slack to close the alert, with later triage runs reusing it as context.

Every agent and chat can now delegate focused subtasks to sub-agents.

Adds Claude Haiku 5.5 to the model catalog, and adds an extra-high reasoning effort level to Claude models.

NEW INTEGRATION Proofpoint integration

NEW INTEGRATION proxycheck.io integration

UPDATED INTEGRATION Databricks integration rebuild

UPDATED INTEGRATION HackerOne report actions for agents

WAS THIS USEFUL? 👍 👎

Earendil Works Pi

Feedback ⓘ v1.1.0 NOTES ↗ RANK

AI agent toolkit: unified LLM API, agent loop, TUI, coding agent CLI

> Pi's biggest addition this window is `codemode`, a QuickJS-sandboxed JavaScript tool that lets models call tools in parallel, run classifiers, and generate images directly from script; alongside it, Pi shipped configurable keybindings, a Program Status Protocol for terminal/agent dashboards, dozens of new slash commands, and a large batch of settings covering tool selection, caching, built-in extensions, and terminal UI.

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

01 **Codemode: sandboxed JavaScript tool execution** NEW 90

Adds a `codemode` tool that runs raw JavaScript (not JSON or markdown) as the body of an async function inside a QuickJS sandbox, with top-level `await` and `return` support. A `// @options: {"max_output_tokens": 2000, "timeout_ms": 60000}` line controls output size (default 10000 tokens, hard cap 16,777,216 chars) and sets a per-script deadline. Scripts get store(key, value)/load(key) globals to persist JSON values (up to 262,144 chars per value, 1,048,576 chars total) across calls scoped per session branch; searchTools(query, { limit?, namespace? }) to BM25-rank callable tools (default limit 8); describeTool(name)/describeNamespace(name) for full TypeScript declarations; `ALL_TOOLS` exposing every callable tool as `{ name, description }`; and text(value)/image(value) to add text or images (PNG, JPEG, GIF, WebP) to output. A `models` global adds getModelsOfType(), getAvailableOfType(), getModelOfType(), classify() and generateImages() using session credentials, with classify() supporting `choice`, `score`, and `bool` question types and returning `probabilities` / `confidence`; up to four concurrent classify()/generateImages() calls run per script, with extras queued. `tools.bash` now returns `{ output, truncated, full_output_path?, exit_code, wall_time_seconds }` with up to 1 MiB of output, and tools.read() returns image files as `{ type: 'image', data, mimeType, note }` blocks. The `codemode.mode` setting (`'on'` | `'only'`) controls whether other tools are also directly annotated or hidden behind `codemode`, and `codemode.inlineBudget` caps the estimated tokens (characters / 4) spent declaring tools in the description at 3000 by default, with over-budget tools still discoverable via searchTools().

Force the model to reach all active tools through codemode — useful when you want scripts to drive tool dispatch exclusively.

```json
"codemode": {
  "mode": "only",
  "inlineBudget": 4000
}
```

— Exhaustive named globals and mechanism, example shows config not a script product docs

02 **Configurable keybindings and `/hotkeys` command** NEW

80

Adds `<agent-dir>/keybindings.json` (default path `~/.pi/agent/keybindings.json`) to assign, remap, or disable keybindings by mapping action identifiers to keys or key lists, and a `/hotkeys` command to display all active shortcuts for the main editor and application.

Remap session and tree actions to custom shortcuts, and disable a default page-up binding you never use.

```json
{
  "app.session.new": "ctrl+shift+n",
  "app.session.tree": ["ctrl+shift+t", "alt+shift+t"],
  "tui.altScreen.pageUp": []
}
```

Check which keybindings are currently active in your Pi session before or after editing the config.

```
$ /hotkeys
```

— Named config file and command with a runnable example product docs

03 **Session management slash commands** NEW

75

Adds `/new` to start a new session and `/resume` to switch to a saved one, `/fork` to create a new session from an earlier user message and `/clone` to duplicate the current session at its current position, `/tree` to navigate the session tree, `/session` to show current session info and statistics, `/name [name]` to set or show the session display name, `/import <path>` to import and resume a JSONL session, `/export [path]` to export a session as HTML or JSONL, `/share` to upload a session and return a viewer link, `/trust` to persist a project trust decision, and `/reload` to reload keybindings, extensions, skills, templates, themes, and context files.

Export the current session to a JSONL file for offline review or archiving after a sensitive engagement.

```
$ /export /tmp/session-2024-review.jsonl
```

— Many named commands with one runnable export example `product docs`

04 **Named built-in extensions control** NEW 75

Adds `builtin:mcp`, `builtin:llama.cpp`, `builtin:codemode`, and `builtin:tool-search` as named built-in extensions, controllable with `+builtin:<name>` / `-builtin:<name>` in project or user settings and with `--no-extensions` / `-e builtin:<name>` on the CLI.

— Named extensions and exact CLI/config flags `product docs`

05 **Program Status Protocol (OSC 7501) reporting** NEW 70

Implements Program Status Protocol (OSC 7501) support so terminals and agent dashboards can track Pi's live state — `working`, `blocked`, `done`, `error`, or `idle` — with per-state messages such as session name, dialog title, or first error line. The `PI_PROGRAM_STATUS` environment variable overrides automatic detection, forcing reports (`PI_PROGRAM_STATUS=1` / `always`) or disabling them entirely (`PI_PROGRAM_STATUS=0` / `never`); otherwise Pi reports only after the terminal confirms support.

— Detailed protocol mechanism, no usage example given `product docs`

06 **Terminal and fullscreen UI display settings** NEW 65

Adds `terminal.showTerminalProgress` to show OSC 9;4 progress in the terminal tab, `fullscreenWheelScrollLines` (1–100 lines per mouse-wheel event) for scroll speed, `terminal.mermaid` (`'off'` | `'final'` | `'streaming'`) for Mermaid diagram rendering, `terminal.imageProtocol` (`'kitty'` | `'iterm2'` | `'auto'` | `false`) for inline-image protocol detection, `terminal.scrollbar` (`'auto'` | `'always'` | `'hidden'`), `terminal.uiMode` (`'regular'` | `'fullscreen'`), `fullscreenExitOutput` (`'transcript'` | `'resume-hint'`), a double-Escape action setting (`"tree"` | `"fork"` | `"none"`), a resize-before-send option capping inline images at 2000×2000 pixels, and a separate setting to block images from being sent to models entirely.

— Many named settings and values, no usage example `product docs`

07 **Session and model introspection environment variables** NEW 60

Adds `PI_REASONING_LEVEL` to expose the current reasoning level of the active model, `PI_SESSION_ID` to expose the current session identifier, and `PI_SESSION_FILE` to

expose the path to the session file so shell scripts can tail session content. Also adds an environment variable to disable automatic network activity, including model catalog refreshes.

Inspect the active provider, model, reasoning level, and session at a glance — useful for debugging CI pipelines or wrapper scripts.

```
$ printf '%s/%s\n' "$PI_PROVIDER" "$PI_MODEL"
printf 'reasoning=%s session=%s\n' "$PI_REASONING_LEVEL"
"$PI_SESSION_ID"
```

— Named vars with example, but last var unnamed `product docs`

---

08 **Model and provider control slash commands** NEW INTEGRATION NEW

60

Adds `/model [provider/model]` to select a model inline, `/thinking [level]` to set the thinking level for the session, `/scoped-models` to configure models used by interactive cycling, `/login [provider]` and `/logout` to manage provider authentication, and `/llama` to manage models on the configured llama.cpp router.

Switch to a specific model mid-session to use a more capable or cost-effective provider for a particular task.

```
$ /model openai/gpt-4o
```

— Named commands with one usage example `product docs`

---

09 **Context compaction command `/compact`** NEW

60

Adds `/compact [instructions]` to compact the current context, optionally with custom compaction instructions.

Compact a long session context with custom instructions before continuing work, to stay within context limits.

```
$ /compact focus on the authentication module decisions and
discard unrelated tool output
```

— Single command with mechanism and runnable example `product docs`

---

10 **Incremental tool selection in `defaultTools`** NEW

60

Adds `+name` / `-name` syntax to `defaultTools` so project settings can extend or subtract from the user's tool selection without fully replacing it, e.g. `["+codemode"]`

enables `codemode` alongside the existing defaults.

Enable codemode alongside the default tools in a project without replacing the inherited tool list.

```json
"defaultTools": ["+codemode"]
```

— Named config syntax with runnable example product docs

THINNER COVERAGE BELOW

11 **Provider transport and retry settings** NEW 55

Adds a configurable transport preference (`'auto'` | `'sse'` | `'websocket'` | `'websocket-cached'`) for AI providers that support multiple transports, plus agent-level retry settings: an enable/disable toggle, maximum attempts, initial exponential-backoff delay, maximum delay, provider request timeout, provider-level retry attempts, and maximum server-requested delay.

— Values named, retry setting keys not given product docs

12 **Prompt-cache warming setting** NEW 50

Adds a cache-warming setting with values `'off'` , `'streaming'` , or `'idle'` to keep eligible provider prompt caches warm during active runs or, with `idle` , between runs; it only fires when Pi estimates at least $0.05 in avoided cache-miss cost.

— Mechanism and threshold given but setting key name missing product docs

13 **Shell and npm command customization settings** NEW 50

Adds an `npm` command setting (command and arguments) for npm package lookup and installation behavior, and a `shell.prefix` setting to prepend a fixed prefix to every shell command.

— Two named settings, no example of use product docs

14 **Agent-directory proxy and trust settings** NEW 45

Adds a proxy URL setting applied as the proxy for Pi-managed HTTP clients, and a `trust` fallback setting (`'ask'` | `'always'` | `'never'`) for project-trust behavior; both are agent-directory only.

— Trust values named, proxy setting left unnamed product docs

15 **Per-model compaction and thinking-budget overrides** NEW 40

Supports per-model compaction token overrides keyed by exact model name, merged from project and user settings objects before model lookup, and per-model startup

thinking levels keyed by exact model name with built-in budgets for `minimal`, `low`, `medium`, and `high` levels.

— Mechanism described but exact setting keys not named product docs

16 **Glob-based resource filtering in settings** NEW 40

Supports glob exclusions (`!`), exact inclusion, and exact exclusion in resource arrays for extensions, skills, prompt templates, and themes.

— Mechanism named, no concrete config key or example product docs

17 **Utility slash commands** NEW 35

Adds `/bug [description]` to prepare a private bug report for Pi developers, `/skill:name` to invoke skill commands when enabled, `/copy` to copy the last assistant message, and `/changelog` to show changelog entries.

— Named commands but no mechanism or example product docs

18 **Transcript notices setting** NEW 35

Adds a `show-notices` setting to surface cache-miss, cache-warming, compaction, and provider-recovery events in the transcript.

— Named setting, minimal mechanism detail product docs

19 **Queued message delivery modes** NEW 25

Adds steering and follow-up message delivery modes (`'all'` | `'one-at-a-time'`) for queued messages.

— Values named but no setting key or example product docs

NEW INTEGRATION Claude Haiku 5.5 model support

NEW INTEGRATION GPT-6 Luna classifier model via Decisions API

v1.1.0 NOTES ↗ LATEST STABLE

Pi v1.1.0 adds OSC 7501 status reporting, additive `+name` / `-name` tool modifiers, `aborted` event field, and `durationMs` in tool render context.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v1.1.0 https://github.com/earendil-
works/pi.git
# already have the repo? check out this version:
$ git checkout v1.1.0
```

Adds `+name` / `-name` modifiers to `--tools` (e.g. `pi -t +codemode,-write`) so entries change the default tool selection instead of replacing it entirely.

Adds `PI_PROGRAM_STATUS=1|0` environment variable to override OSC 7501 program-status detection for terminals and agent dashboards that report whether Pi is working, blocked, done, or failed.

Adds `aborted` field to `agent_settled` session, extension, and JSON events so integrations can distinguish a cancelled run from a finished one.

Adds `durationMs` to the tool render context and to `tool_execution_end` extension events, recording the execution time of a final tool result.

Adds `outputPad` to the tool render context.

+2 more

WAS THIS USEFUL? 👍 👎

Zed

Feedback ⓘ v1.23.2 NOTES ↗ RANK

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

> Zed v1.23.2 adds JSONL/NDJSON table previews, new settings for agent thread retention and soft-wrap indentation, and several editor and Git panel refinements, while changing title-bar menu behavior on Linux and Windows to require a click.

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

01 **Soft-wrap indentation setting** NEW 77

Adds `soft_wrap_indent` setting with values `"none"`, `"same"`, `"extra_one"`, `"extra_two"` to configure indentation for soft-wrapped continuation lines, e.g. setting it to `"same"` keeps continuation lines aligned with the wrapped line's indentation.

Keep continuation lines aligned with the indentation of the wrapped line to improve readability of long wrapped text.

```json
// In your Zed settings.json
{
  "soft_wrap_indent": "same"
}
```

— Config key, all values, and example all given v1.23.2

02 **JSONL/NDJSON tabular data preview** NEW 73

Adds JSONL and NDJSON support to the tabular data preview, openable alongside the source file with `cmd-k v` (macOS) or `ctrl-k v` (Linux/Windows).

Preview a JSONL log file as a searchable table alongside the raw source in a split pane.

📍 cmd-k v

— Exact keybindings and formats named, with usage example v1.23.2

03 **Agent idle thread retention setting** NEW 70

Adds `agent.max_idle_retained_threads` setting to control how many idle agent threads remain loaded, letting users cap memory use by limiting resident AI agent threads between tasks.

Cap memory use by limiting how many idle AI agent threads stay resident between tasks.

```json
// In your Zed settings.json
{
  "agent": {
    "max_idle_retained_threads": 2
  }
}
```

— Names exact config key with example; mechanism briefly explained v1.23.2

04 **Title-bar menu hover behavior changed** BREAKING 69

Title-bar application menus on Linux and Windows now require a click before opening; hover-to-open is disabled by default. The `title_bar.open_menus_on_hover` setting can be set to `true` to restore the previous hover-to-open behavior.

— Clear before/after and exact setting to revert, no example v1.23.2

THINNER COVERAGE BELOW

05 **Multi-keystroke shortcut completion hint** NEW 45

Adds `alt-/` keybinding to show available completions for a pending multi-keystroke shortcut.

— Exact keybinding named but no further mechanism or example v1.23.2

06 **Whole-word aware select next/previous** IMPROVED 45

Improves `editor: select next` and `editor: select previous` to respect the buffer search whole-word option.

— Names exact commands and option but no example v1.23.2

07 **Typst rewrap support** NEW 30

Adds support for rewrapping text in Typst files.

— Names the language but no command or flag given v1.23.2

08 **Git Panel folder context menu** NEW 28

Adds a folder-specific context menu in the Git Panel's tree view.

— Bare description with no UI path or scope detail v1.23.2

09 **Broader extension install suggestions** IMPROVED 20

Adds extension installation suggestions for many more languages when opening unsupported files.

— Vague 'many more languages' with no list or numbers v1.23.2

└──▷ BREAKING ON UPGRADE

! Title-bar application menus on Linux and Windows now require a click before opening; hover-to-open is disabled by default. Set `title_bar.open_menus_on_hover` to `true` to restore the previous behavior.

WAS THIS USEFUL? 👍 👎

Anthropic Claude Code

Feedback ⓘ v2.1.293 NOTES ↗

CODE ↗ RANK

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

> Claude Code v2.1.293 switches to a new default Haiku model with expanded context and lower pricing, and ships a trio of developer-facing additions for mod/plugin authoring and testing.

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

01 **Mod and plugin development API additions** NEW 80

Adds `agentType` field to the `subagentStatusLine` payload so scripts can distinguish custom subagent types from one another; adds `isDeferred` option to `$.tool.register` for mods, which when set to `false` lists the tool's schema in the prompt from the start rather than behind tool search; and adds `mock.session` to `claude plugin test`, letting tests for mods that call `$.session.append` read back the appended rows.

Register a mod tool so its schema is available in the prompt immediately rather than requiring tool search, useful for tools Claude should always know about.

```
$ $.tool.register({ name: "myTool", isDeferred: false, ... })
```

— Names three distinct mod APIs with mechanism and a runnable example v2.1.293

02 **Claude Haiku 5.5 as default Haiku model** NEW 66

Claude Code now uses `claude-haiku-5-5` (Claude Haiku 5.5) as the new default Haiku model on the Anthropic API, with 1M token context and pricing of $0.10/$0.50 per Mtok (input/output), rising to $0.50/$2.50 per Mtok for prompts over 100K tokens.

— Names model, context size and tiered pricing, but no usage command v2.1.293

WAS THIS USEFUL? 👍 👎

Amazon Q Developer

Feedback ⓘ changelog · Oct 8, 2026

CHANGELOG ↗

Amazon Q Developer provides AI-assisted code generation, security scanning, and Java application modernization for AWS.

NEW INTEGRATION AWS CodeConnections repository support

WAS THIS USEFUL? 👍 👎

AGENT

◆ AI Agent Frameworks

AutoGPT

Feedback ⓘ autogpt-platform-beta-v0.8.3 NOTES ↗

AutoGPT automates multi-step workflows with continuous AI agents that users can build and deploy.

Adds `ENABLE_USER_NOTIFICATIONS` env switch, email-verified sign-up gating, stream cursors for turn resumption, and native chat-channel approval buttons.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch autogpt-platform-beta-v0.8.3
https://github.com/Significant-Gravitas/AutoGPT.git
# already have the repo? check out this version:
$ git checkout autogpt-platform-beta-v0.8.3
```

Adds `ENABLE_USER_NOTIFICATIONS` environment-variable switch so production deployments can restrict outbound emails to auth messages only.

Requires a verified email before the first session and provisions the platform User at sign-up.

Adds signup consent line, marketing opt-out, and MailerLite gate.

IDs every stream entry, checkpoints every persist, and enables resuming a turn from a cursor.

Lets users edit and combine suggested answers in question cards.

+2 more

NEW INTEGRATION PayPal MCP Server added to MCP catalog

NEW INTEGRATION Canva MCP Server added to MCP catalog

NEW INTEGRATION Conductor integration blocks

NEW INTEGRATION GitHub repository traffic block

NEW INTEGRATION Hacker News search, stories, item and user blocks

NEW SUPPORTED PLATFORM Native approval-card buttons for Discord, Slack, Teams, and Telegram channels

NEW SUPPORTED PLATFORM GPT-6.1 Sol added to LLM catalog

NEW SUPPORTED PLATFORM Mistral Large 4 added to LLM catalog

NEW SUPPORTED PLATFORM Mistral Medium 3.5, Small 4, Ministral 3 added to LLM catalog

RETIRED SUPPORTED PLATFORM Mistral Medium 3.1 and Small 3.2 retired from LLM catalog

└──▷ **BREAKING ON UPGRADE**

! Mistral Medium 3.1 and Small 3.2 are retired from the LLM catalog; agents configured to use either model must be updated.

WAS THIS USEFUL? 👍 👎

Vercel AI SDK

Feedback ⓘ ai@7.0.133 NOTES ↗ RANK

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

> The Vercel AI SDK expanded its experimental decision state with ordered text/file/JSON parts (with a breaking change for raw JSON arrays), added image input support to OpenAI Decisions, and introduced an `onUIMessageStepEnd` callback for persisting UI messages at step boundaries.

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

01 Ordered parts and telemetry for decision state

BREAKING

83

Adds ordered `text`, `file`, and `json` parts to the experimental decision state; all public state forms are normalized into an array of parts before calling decision providers, with providers receiving text and JSON objects as `text` and `json` parts. Decision file bytes are serialized as base64 in OpenTelemetry state attributes, with model-call spans recording normalized state parts while outer spans retain public state inputs; native JSON state in TypeSafe AI is preserved when state contains one JSON part, and shared state is now labeled in language-model decision prompts. Gateway and TypeSafe AI retain their existing string and JSON request format and reject files. Breaking: JSON arrays passed directly as state are now interpreted as decision state parts arrays — wrap them in an object or a `json` part to avoid misinterpretation.

— Names parts, telemetry fields, and migration note but no code sample ai@7.0.133

02 onUIMessageStepEnd callback for message persistence

NEW

60

Adds `onUIMessageStepEnd` callback to the AI SDK, enabling developers to persist messages at the end of each UI message step.

— Names exact callback but no code example shown ai@7.0.132

THINNER COVERAGE BELOW

03 Image input support in OpenAI Decisions

NEW

30

OpenAI Decisions and language model adapters now accept image input.

— Bare statement with no mechanism or usage detail ai@7.0.133

└──▷ **BREAKING ON UPGRADE**

! Arrays passed directly as state now contain decision state parts — JSON arrays must be wrapped in an object or a `json` part to avoid being interpreted as a parts array.

WAS THIS USEFUL? 👍 👎

LangChain

Feedback ⓘ `langchain-huggingface==1.2.3` NOTES ↗ RANK

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

> langchain-huggingface 1.2.3 improves device handling for HuggingFace pipelines and enriches tracing metadata with package version info.

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

01 **Device-agnostic accelerator detection** IMPROVED 50

Switches to `torch.accelerator` for device-agnostic device count detection, removing the need to hard-code CUDA/CPU logic in HuggingFace pipelines.

— Names the API but no usage detail `langchain-huggingface==1.2.3`

02 **Package version in tracing metadata** NEW 50

Adds package version tracking to tracing metadata, surfacing the active `langchain-huggingface` version in LangSmith traces.

— Clear what changed, no example shown `langchain-huggingface==1.2.3`

WAS THIS USEFUL? 👍 👎

Stacklok Mecatl

Feedback ⓘ v0.0.47 NOTES ↗ RANK

Mecatl is an open-source agent harness for running the same provider-agnostic agent loop locally, remotely, or on Kubernetes, with durable state, permissions, and gRPC/HTTP-SSE APIs.

> Mecatl's two releases this window added visibility into agent activity in mecatui and a new skills management workflow in Studio.

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

01 **Edit and Subagent visibility in mecatui** NEW 37

`mecatui` now shows inspection of Edit requests and bounded Subagent activity, giving operators visibility into what agents are editing and doing in sub-agent runs.

— Names the surface but not mechanism or limits of 'bounded' v0.0.47

02 **Skills inventory and review in Studio** NEW 37

Studio gains a skills inventory, a skill detail page, and a learned-skill review workflow for managing skills the agent has acquired.

— Describes the workflow but lacks UI path or config detail v0.0.46

WAS THIS USEFUL? 👍 👎

Microsoft Agent Framework

Feedback ⓘ python-1.21.0

NOTES ↗

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

Adds maximum reasoning-effort control for OpenAI, full-buffer Purview policy evaluation for streamed responses, variable-expansion argument exposure, tool-result standing guidance, FIDES enforcement warnings for CodeAct tools, and session-state isolation when agents are invoked as tools.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch python-1.21.0
https://github.com/microsoft/agent-framework.git
# already have the repo? check out this version:
$ git checkout python-1.21.0
```

Adds the maximum reasoning-effort option to `agent-framework-openai`.

Supports full-buffer Purview policy evaluation for streamed responses in `agent-framework-purview`.

Exposes rewritten variable-expansion arguments in `agent-framework-core`.

Adds tool-result standing guidance in `agent-framework`.

Warns when CodeAct tools cannot support FIDES security enforcement in `agent-framework-hyperlight` and `agent-framework-monty`.

+1 more

NEW INTEGRATION Oracle native vector-store connector (alpha)

WAS THIS USEFUL? 👍 👎

OpenClaw

Feedback ⓘ v2026.9.9 NOTES ↗ CODE ↗ RANK

OpenClaw is a personal AI assistant platform that connects messaging channels, tools, and model providers through a gateway.

> OpenClaw added GPT-6.1 Sol support to its managed Codex runtime in this release.

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

01 **GPT-6.1 Sol model in managed Codex runtime** NEW 68

The managed Codex runtime (0.160.0) now includes GPT-6.1 Sol in its model catalog, allowing native Codex sessions to discover and select the model with its `low`, `medium`, `high`, `xhigh`, `max`, and `ultra` effort choices plus a `priority` / Fast tier.

— Names exact effort/tier options but no usage example or command shown. v2026.9.9

WAS THIS USEFUL? 👍 👎

Feynman

Feedback ⓘ v0.5.30 NOTES ↗ RANK

Feynman is a personal AI research assistant that searches papers and the web, builds literature reviews, and produces cited research reports.

> Feynman's Pi 1.1.0 runtime update adds Claude Haiku 5.5 model support and fixes a Windows/WSL browser login issue with automatic port fallback.

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

01 **Automatic port fallback for Claude browser login** IMPROVED

65

Claude browser login now falls back to a free port when port `53692` is reserved or in use (common with Hyper-V/WSL on Windows), eliminating 'localhost refused to connect' errors.

— Names exact port and failure mode fixed v0.5.30

THINNER COVERAGE BELOW

02 **Claude Haiku 5.5 model support** NEW

40

Pi 1.1.0 adds Claude Haiku 5.5 as an available model; Feynman's default for Anthropic remains Claude Opus.

— Names model and runtime but no usage detail v0.5.30

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ Agentic Infrastructure

HeyGen HyperFrames

Feedback ⓘ v0.8.141 NOTES ↗ RANK

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

> HyperFrames added a configurable API gateway environment variable and improved diagnostics for font resolution failures.

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

01 Configurable HeyGen API base via environment variable NEW

60

HyperFrames adds the `HEYGEN_API_BASE` environment variable so that `media-use` can reach HeyGen through a host application's gateway instead of a fixed endpoint.

— Names exact env var but lacks usage example v0.8.141

THINNER COVERAGE BELOW

02 Structured diagnostics for unresolved font errors IMPROVED

45

Unresolved-font errors now carry structured diagnostics about which resolver step failed, without exposing font names, URLs, or page text.

— Explains behavior change but no API or flag given v0.8.141

WAS THIS USEFUL? 👍 👎

Ollama

Feedback ⓘ v0.40.1 NOTES ↗ CODE ↗ RANK

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

> Ollama v0.40.1 adds local proxying for cloud usage and balance checks and simplifies CLI onboarding by removing an extra account screen.

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

01 **Local proxy for cloud usage and balance APIs** NEW 83

Ollama proxies `GET /api/usage` and `GET /api/balance` through the local Ollama server using the signed-in account, with added signing and disabled-cloud-access handling. Readers can check these directly via `curl http://localhost:11434/api/usage` and `curl http://localhost:11434/api/balance` instead of hitting the cloud endpoint manually.

Check cloud usage or account balance directly from the local Ollama server without hitting the cloud endpoint manually.

```
$ curl http://localhost:11434/api/usage
curl http://localhost:11434/api/balance
```

— Names exact endpoints and runnable curl commands v0.40.1

THINNER COVERAGE BELOW

02 **Streamlined CLI onboarding flow** IMPROVED 35

Removes the intermediate account screen from CLI onboarding — pressing Enter on the Welcome screen now opens the launcher directly.

— Describes the change but no further mechanism v0.40.1

WAS THIS USEFUL? 👍 👎

Google Gen AI Python SDK

Feedback ⓘ v2.29.0 NOTES ↗

RANK

The Google Gen AI Python SDK provides Python interfaces to Gemini model generation, streaming, tools, and embeddings.

> Google Gen AI Python SDK introduced a unified Interactions API for stateful Gemini sessions, adding multimodal output, built-in tool access, text-to-speech and image generation models, plus a new network allowlist-disable option.

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

01 Text-to-speech generation via gemini-3.8-flash-tts NEW

95

Adds TTS generation support using model `'gemini-3.8-flash-tts'` with configurable voice via `speech_config`, `SpeechConfig`, `voice_config`, `VoiceConfig`, `prebuilt_voice_config`, `PrebuiltVoiceConfig`, and `voice_name` (e.g. `'Aoede'`), producing base64 audio output accessible for saving to a WAV file.

Generate spoken audio from text and save it as a WAV file, using a named prebuilt voice.

python

```python
from base64 import b64decode
from google import genai
from google.genai.types import SpeechConfig, VoiceConfig,
PrebuiltVoiceConfig

client = genai.Client()

interaction = client.interactions.create(
    model='gemini-3.8-flash-tts',
    input='Give me a cheerful greeting.',
    output_modality='AUDIO',
    speech_config=SpeechConfig(
        voice_config=VoiceConfig(

prebuilt_voice_config=PrebuiltVoiceConfig(voice_name='Aoede')
        )
    )
)

with open('greeting.wav', 'wb') as f:
    f.write(b64decode(interaction.output_audio.base64))
```

Generate speech audio from a prompt using the TTS model and a prebuilt voice, then save the result to a WAV file.

```python
from base64 import b64decode
interaction = interactions(
    'gemini-3.8-flash-tts',
    input='Give me a cheerful greeting.',
    output_modality='AUDIO',
    speech_config=SpeechConfig(
        voice_config=VoiceConfig(

prebuilt_voice_config=PrebuiltVoiceConfig(voice_name='Aoede')
        )
    )
)
with open('greeting.wav', 'wb') as f:
    f.write(b64decode(interaction.output_audio.base64))
```

— Full config chain and runnable audio-generation example. product docs

02 **Interactions API for stateful multi-turn sessions** NEW

85

Adds the `interactions` API, a unified interface that simplifies state management, tool orchestration, and long-running tasks when working with Gemini models and agents. Multi-turn conversations can be continued server-side via `previous_interaction_id` passed from the prior turn's result, removing the need for client-side history tracking.

Maintain a multi-turn conversation server-side — no local history list needed — by passing `previous_interaction_id` from the prior turn.

```python
from google import genai

client = genai.Client()

# First turn
interaction1 = client.interactions.create(
    model='gemini-flash-latest',
    input='Hi, my name is Amir.'
)
print(interaction1.output_text)

# Second turn — SDK continues the session on the server
interaction2 = client.interactions.create(
    model='gemini-flash-latest',
    input='What is my name?',
    previous_interaction_id=interaction1.id
)
print(interaction2.output_text)
```

Continue a multi-turn conversation without managing history client-side, by passing `previous_interaction_id` from the first turn into the next.

```python
interaction1 = interactions('gemini-flash-latest', input='Hi, my
name is Amir.')
interaction2 = interactions('gemini-flash-latest', input='What
is my name?', previous_interaction_id=interaction1.id)
print(interaction2.output_text)
```

Run a stateful two-turn conversation without managing history client-side — useful in chatbots or agentic workflows.

```python
interaction1 = interactions.create(model='gemini-flash-latest',
input='Hi, my name is Amir.')
interaction2 = interactions.create(model='gemini-flash-latest',
input='What is my name?',
previous_interaction_id=interaction1.id)
print(interaction2.output_text)
```

— Names API, param, and gives runnable multi-turn code. product docs

03 **Built-in tool access via Interactions API** NEW 80

Built-in tools such as `'google_search'` and code execution can be invoked directly through the `interactions` API by passing a `tools` argument, letting a single call route a query through live tools like web search.

Answer questions that require live web data by routing a query through Google Search inside the Interactions API.

python

```python
from google import genai

client = genai.Client()

interaction = client.interactions.create(
    model='gemini-flash-latest',
    input='Who won the last Super Bowl?',
    tools=['google_search']
)
print(interaction.output_text)
```

Answer a real-time question by routing it through Google Search as a built-in tool inside a single interactions call.

python

```python
interaction = interactions(
    'gemini-flash-latest',
    input='Who won the last Super Bowl?',
    tools=['google_search']
)
print(interaction.output_text)
```

Answer a real-time factual question by invoking Google Search as a built-in tool from within an Interactions API call.

python

```python
interaction = interactions.create(model='gemini-flash-latest',
input='Who won the last Super Bowl?', tools=['google_search'])
print(interaction.output_text)
```

— Names tool identifiers with a runnable example. product docs

---

04 **Multimodal output in Interactions API** NEW 65

The `interactions` API supports multimodal output (image and audio) by specifying an output modality of `'IMAGE'` or `'AUDIO'`, with results accessible via the `output_image` and `output_audio` fields.

— Names fields and modality values but no standalone example. product docs

THINNER COVERAGE BELOW

05 **Disable network allowlist in environment config** NEW

55

Supports `"allowlist": "disabled"` as a value in environment network config, enabling explicit disablement of the network allowlist.

— Names config value but no surrounding usage context. v2.29.0

06 **Image generation via gemini-3.1-flash-lite-image** NEW

35

Adds image generation support through the `interactions` API using the model `'gemini-3.1-flash-lite-image'`.

— Names the model but gives no mechanism or example. product docs

WAS THIS USEFUL? 👍 👎

Google Gen AI JavaScript SDK

Feedback ⓘ v2.28.0

NOTES ↗ RANK

The Google Gen AI JavaScript SDK provides TypeScript and JavaScript interfaces to Gemini APIs and live model sessions.

> The Google Gen AI JavaScript SDK added a new value to its environment network configuration for explicitly disabling allowlists.

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

01 **Explicit allowlist disabling in network config** NEW

50

The environment network config now supports `"allowlist": "disabled"` as a value, enabling explicit disabling of network allowlists.

— Names exact config value but no further mechanism given v2.28.0

WAS THIS USEFUL? 👍 👎

emisar

Feedback ⓘ v0.51.0 NOTES ↗ RANK

Emisar lets AI agents securely run infrastructure actions through MCP, with approval controls that protect production systems.

> Emisar v0.51.0 replaces personal logins with workspace-level sign-in, adds service accounts with scoped API keys, and ships a new SuperTokens auth-core pack alongside rate limiting and pack-visibility improvements.

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

01 **Service accounts with scoped API keys** NEW 73

A dedicated Settings page in the Console lets admins create service accounts and generate API keys with scoped access, for cases like a non-human app needing limited workspace access.

Create a service account for an app such as Ryker that needs scoped, non-human API access to a workspace.

📍 In the Console, go to Settings › Service Accounts and select Create service account, then generate an API key with the required scoped access.

— Has exact UI path and example steps v0.51.0

02 **Workspace-level sign-in replaces personal logins** BREAKING 60

Sign-in moves to the workspace level, replacing personal logins, with per-workspace email, MFA, and SSO enforcement; refused SSO sign-ins are now recorded in the workspace audit trail. Any setup relying on the old personal login flows will break.

— Explains scope and before/after but no migration steps v0.51.0

THINNER COVERAGE BELOW

03 **Pack availability visibility in Console** IMPROVED 55

The Console now shows pack availability: which actions are available and which runners reported an exact pack version and hash, with partial evidence explicitly labeled as partial. The Runner also reports local admission separately from the complete trusted pack descriptor.

— Explains what's shown but not where to view it v0.51.0

04 **SuperTokens auth-core pack** NEW 50

A new SuperTokens pack covers auth-core health, users, sessions, and roles, with gated mutations.

— Names covered areas but no usage detail v0.51.0

05 **Per-client-address request caps for MCP and audit-export endpoints**

NEW

45

Adds rate limiting in the form of per-client-address request caps for MCP and audit-export endpoints.

— Names endpoints but no limit values given v0.51.0

06 **Member email updates via Profile and directory sync** IMPROVED

45

Members can update their email address from Profile after mailbox verification; directory sync can also push email updates, and refused changes are surfaced.

— Describes flow but no exact navigation or config v0.51.0

└──▷ **BREAKING ON UPGRADE**

! Workspace sign-in replaces personal logins — members now authenticate through workspace-specific email, MFA, and SSO settings; any setup relying on personal login flows will break.

WAS THIS USEFUL? 👍 👎

ArcBox

Feedback ⓘ v0.9.0 NOTES ↗ RANK

ArcBox runs containers, microVM agent sandboxes, and full Linux or macOS virtual machines on a Mac from one Rust daemon and CLI.

> ArcBox v0.9.0 adds a v7 storage protocol with health monitoring and recovery endpoints, lets callers inject agents into rootfs at boot (a breaking change), and translates Mac AppleDouble sidecars into real extended attributes.

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

01 Storage health monitoring and v7 recovery protocol NEW

73

ArcBox defines a `v7` storage protocol with observations and recovery messages for structured health and recovery signalling between daemon and agents. The API gains explicit storage-check and recovery endpoints, the agent gains persistent storage mount-health observation and unmounted-storage checking in a read-only mode (without triggering repair), and the daemon publishes storage health and recovery progress with cancellable storage-verification RPCs streamed from the engine. Writes to Docker are rejected while storage is in a protected (recovering) state, volume provisioning is enforced as single-use, paired volume identity manifests verify recovery copy identities, the system reserves storage during recovery and rejects stale storage agents before boot, and storage recovery is isolated from runtime startup with writable-volume verification owned in core.

— Names many protocol/API surfaces but no exact endpoint paths given v0.9.0

02 AppleDouble xattr translation and sidecar lifecycle NEW

64

ArcBox translates Mac AppleDouble sidecars into real extended attributes (`xattrs`) on target files, preserving macOS metadata when crossing filesystem boundaries. The agent parses and builds AppleDouble sidecar images, supporting storage of a sidecar's content as the target file's `xattrs` , and sidecar images gain an apply mode, a generation counter, and a quiet period to control lifecycle of in-flight operations.

— Mechanism well described but no CLI command or usage example shown v0.9.0

THINNER COVERAGE BELOW

03 Agent injection at boot in computer subsystem BREAKING

58

ArcBox exposes agent injection in the `computer` subsystem and selects it at boot, allowing callers to inject agents into rootfs at machine creation time. This is a breaking

change: existing setups that do not account for agent injection at boot may behave differently after upgrade.

— Clear mechanism and explicit breaking-change note but no config example v0.9.0

04 **Rootfs and registry image handling in computer subsystem** NEW

40

The `computer` subsystem now builds rootfs at caller-owned capacity and path, and exposes registry image sources while verifying sparse capacity.

— Thin bullets grouped, no concrete values or commands given v0.9.0

└──▷ **BREAKING ON UPGRADE**

! Agent injection is now exposed and selected at boot in the `computer` subsystem — existing setups that do not account for agent injection at boot may behave differently after upgrade.

WAS THIS USEFUL? 👍 👎

Fireworks AI

Feedback ⓘ

Doubled serverless rate limits for Small models · NOTES ↗ · RANK

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

> Fireworks AI doubled serverless rate-limit ceilings for its Small model tier and clarified tier assignments in its rate-limit documentation.

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

01 **Doubled rate limits for Small serverless models** IMPROVED

68

The Small serverless model tier (models under 600B total parameters) now has adaptive rate-limit ceilings of `216M Total Prompt TPM`, `54M Uncached Prompt TPM`, and `2.16M Generated TPM` — double the previous limits. Medium and Large tier ceilings are unchanged.

— Gives exact new numeric limits but no usage mechanism

Doubled serverless rate limits for Smal…

THINNER COVERAGE BELOW

02 **Model tier listing in rate limits docs** IMPROVED

30

The Serverless rate limits reference page now lists which tier (Small, Medium, or Large) each Serverless model belongs to.

— Documents a doc page addition with no further detail

Doubled serverless rate limits for Smal…

WAS THIS USEFUL? 👍 👎

BerriAI LiteLLM

Feedback ⓘ v1.104.2 NOTES ↗ RANK

LiteLLM is an AI gateway that calls over 100 model APIs in OpenAI format with cost tracking, guardrails, load balancing, and logging.

> LiteLLM shipped text-to-speech support across 8 providers with a proxy endpoint and a chat-completions bridge, an official Moyai coding-agent integration with spend tracking and budget enforcement, and new OpenAI Decisions endpoints on the AI Gateway.

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

01 **Moyai coding agent integration** NEW 95

Adds support for routing Moyai coding-agent harnesses (`/v1/messages` , `/v1/responses` , `/v1/chat/completions`) through the LiteLLM AI Gateway, keeping provider API keys on the gateway rather than in the agent sandbox. Configuration uses `LITELLM_API_BASE` , `LITELLM_API_KEY` , and `AGENT_MODEL` in Moyai's `.env` file; `x-litellm-response-cost` and `x-litellm-call-id` response headers let Moyai record per-teammate spend and correlate requests, and gateway spend logs gain a `spend_logs_metadata` field containing `moyai_request_id` . Budget enforcement is available via `max_budget` on a dedicated virtual key created at `POST /key/generate` , stopping requests when the limit is reached, and each Moyai harness supports cross-provider model routing — e.g. the Claude Agent SDK harness can use an OpenAI model — as long as the model supports tool calls.

Create a budget-capped virtual key scoped to Moyai's model aliases so the gateway hard-stops the agent before runaway spend.

```
$ curl -X POST "$PROXY_BASE_URL/key/generate" \
  -H "Authorization: Bearer $LITELLM_MASTER_KEY" \
  -H "Content-Type: application/json" \
  -d '{
    "key_alias": "moyai",
    "models": ["openai/gpt-6-astra", "claude-sonnet-5-5"],
    "max_budget": 100
  }'
```

Point a Moyai installation at your LiteLLM gateway and choose which model alias it sends to each harness.

```text
LITELLM_API_BASE=https://your-gateway.example.com/v1
LITELLM_API_KEY=sk-...
AGENT_MODEL=openai/gpt-6-astra
```

Register the model aliases Moyai will use so the gateway can route them to the correct provider.

```yaml
model_list:
  - model_name: openai/gpt-6-astra
    litellm_params:
      model: openai/gpt-6-astra
      api_key: os.environ/OPENAI_API_KEY
  - model_name: claude-sonnet-5-5
    litellm_params:
      model: anthropic/claude-sonnet-5-5
      api_key: os.environ/ANTHROPIC_API_KEY
```

— Names env vars, headers, log field, endpoint and runnable budget example product docs

02 **Text-to-speech SDK and proxy endpoint** NEW 91

Adds speech() and aspeech() Python SDK functions for text-to-speech across 8 providers: OpenAI, Azure OpenAI, Azure AI Speech Service (AVA), AWS Polly, Vertex AI, Gemini, ElevenLabs, and MiniMax. Exposes an OpenAI-compatible `/audio/speech` proxy endpoint so existing OpenAI TTS clients route through LiteLLM Proxy without code changes, plus a `/audio/speech` to `/chat/completions` bridge enabling TTS via chat-completion models like `gemini/gemini-2.5-flash-preview-tts` that aren't natively accessible through a speech endpoint. A new `max_file_size_mb` config key for `audio/transcriptions` models (Enterprise) enforces a per-model request file size limit, and TTS calls support cost tracking, logging, end-user tracking, fallbacks, load balancing, and guardrails (guardrails apply to input text in non-streaming mode only).

Generate speech from text using the async SDK and save the result to an MP3 file.

```python
import asyncio
from litellm import aspeech
from pathlib import Path
import os

os.environ["OPENAI_API_KEY"] = "sk-.."

async def main():
    speech_file_path = Path(__file__).parent / "speech.mp3"
    response = await aspeech(
        model="openai/tts-1",
        voice="alloy",
        input="the quick brown fox jumped over the lazy dogs"
    )
    response.stream_to_file(speech_file_path)

asyncio.run(main())
```

Route TTS requests through LiteLLM Proxy using a curl call compatible with any OpenAI TTS client.

```
$ curl http://0.0.0.0:4000/v1/audio/speech \
  -H "Authorization: Bearer $LITELLM_API_KEY" \
  -H "Content-Type: application/json" \
  -d '{"model": "tts-1", "input": "The quick brown fox jumped
over the lazy dog.", "voice": "alloy"}' \
  --output speech.mp3
```

Cap the file size accepted by a Whisper transcription model in the proxy to prevent oversized uploads.

```yaml
model_list:
  - model_name: whisper
    litellm_params:
      model: whisper-1
      api_key: sk-*******
      max_file_size_mb: 25
    model_info:
      mode: audio_transcription
```

— Names SDK functions, endpoint, config key, providers and runnable examples product docs

THINNER COVERAGE BELOW

03 **OpenAI Decisions provider and endpoints** NEW 45

Adds `/v1/decisions` (OpenAI-format) and `/v1/systemone` endpoints to the AI Gateway, enabling decisions-style completions through LiteLLM's unified proxy, along with an OpenAI Decisions provider that allows LiteLLM to route requests to the Decisions API using the same OpenAI-compatible client interface.

— Names endpoints and provider but no usage example or mechanism `v1.104.2`

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

vMLX

Feedback ⓘ v1.6.76 NOTES ↗ RANK

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

> vMLX 1.6.76 brings image and video prefill to DFlash2 sessions along with stop-string handling, cache isolation fixes, and snapshot/rollback correctness improvements across the native and vision caching layers.

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

01 **Image/video prefill in DFlash2 sessions** NEW 57

Qwen DFlash2 sessions now support image and video prefill, with reusable media context and preserved ordering of mixed image/video history across turns.

– Describes mechanism and scope but no API/flag named v1.6.76

02 **API stop-string handling in DFlash2 streaming** NEW 50

Adds explicit API stop-string support during DFlash2 generation, correctly detecting stop markers even when they are split across streamed tokens.

– Clear behaviour described, no concrete flag/endpoint named v1.6.76

03 **SSD vision cache isolation and clearing** IMPROVED 50

SSD vision cache entries are now isolated by media, model weights, and runtime identity, ensuring they are cleared only by the appropriate cache-clear action.

– Explains isolation keys but no exact command or config v1.6.76

04 **Rotating native cache rollback protection** IMPROVED 43

Rotating native caches now preserve their absolute position and reject an unrecoverable rollback instead of returning corrupted continuations.

– Describes failure mode fixed but no named config or API v1.6.76

05 **Native multimodal prefill usage accounting** IMPROVED 24

Native multimodal usage and prefill reporting now account for reused tokens correctly.

– Bare statement of a fix with no detail on mechanism v1.6.76

└──▷ BREAKING ON UPGRADE

! Existing DFlash2 snapshots are invalidated — incompatible older snapshots will not be restored on upgrade.

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

Neo4j Agent Memory

Feedback ⓘ python-v0.7.0 CODE ↗ RANK

Agent Memory is a graph-native memory system for AI agents that stores conversations, builds knowledge graphs, and tracks reasoning traces with Neo4j.

> Agent Memory's python-v0.7.0 release replaces its entity/relation extraction stack with a new GLiNER2.5-based joint extractor, adds an installable metrics module for evaluation, and removes the old GLiNER v1/GLiREL dependencies.

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

01 **GLiNER2.5 joint entity-relation extractor** NEW 97

Adds a `GLiNER2Extractor` class (`extraction/gliner2_extractor.py`) for joint entity and relation extraction using GLiNER2.5 checkpoints (`fastino/gliner2.5-small-v1` , `fastino/gliner2.5-base-v1` , `fastino/gliner2.5-multi-v1`) via the `gliner2` package's JointIE engine, with constructors .for_schema(name), .for_poleo(), and .for_ontology(doc), identity surface `name == 'gliner2'` , and fields `version` , `model_id` , and `ontology` . Ontology-declared endpoint types are enforced during beam search rather than post-filtering, so extracted relations carry `source_id` / `target_id` mention ids and every endpoint is guaranteed to be an entity in the same result; long-document extraction is supported via `extract_long` (windowed for inputs exceeding `max_words` of 384) and batch decoding via extract_batch().

Extract entities and typed relations from a support ticket using the ontology-aware GLiNER2.5 extractor in a single forward pass.

python

```
from neo4j_agent_memory.extraction.gliner2_extractor import
GLiNER2Extractor

extractor = GLiNER2Extractor.for_poleo()
result = await extractor.extract("Alice called about her account
#12345 being locked after a failed login.")
for entity in result.entities:
    print(entity)
for relation in result.relations:
    print(relation.source_id, '->', relation.target_id)
```

Benchmark GLiNER2.5 checkpoint variants at different relation thresholds to choose the right model for your extraction pipeline.

```
$ uv run python benchmarks/compare_extractors.py \
    --models fastino/gliner2.5-small-v1,fastino/gliner2.5-base-
v1 \
    --relation-thresholds 0.3,0.5 \
    --output results.json
```

Use a custom ontology document to drive extraction when your domain has specialised entity and relation types.

```python
from neo4j_agent_memory.extraction.gliner2_extractor import
GLiNER2Extractor

extractor = GLiNER2Extractor.for_ontology(my_ontology_doc)
result = await extractor.extract_batch([text1, text2, text3])
```

— Names class, checkpoints, constructors, and mechanism with runnable example

python-v0.7.0

02 **Installable extraction metrics module** NEW 75

Ships `neo4j_agent_memory.core.metrics` as an installed module (previously dev-only under `benchmarks/`), exporting `calculate_entity_metrics`, `calculate_relation_metrics`, `calculate_extraction_metrics`, `bcubed`, `BCubedScore`, `ResolutionGold`, `ResolutionMention`, `DEFAULT_MATCH_POLICY`, `STRICT_MATCH_POLICY`, and related types so `neo4j_agent_memory.memory.eval` can use scoring without a repository checkout.

— Names exact exports and module path but no usage example python-v0.7.0

03 **Removal of GLiNER v1 and GLiREL stack** BREAKING 70

The `gliner` v1 package and the separate GLiREL relation model are removed; the local extraction stack now requires the `gliner2` package and GLiNER2.5 checkpoints (`fastino/gliner2.5-{small,base,multi}-v1`) — any code or configuration referencing the old `gliner` package or GLiREL will break on upgrade.

— States exact removed/required packages and upgrade impact python-v0.7.0

THINNER COVERAGE BELOW

04 **Ontology labels on entity nodes in Bolt backend** IMPROVED

25

Writes ontology labels on entity nodes in the Bolt backend.

— One-line addition with no further mechanism given `python-v0.7.0`

└──▷ **BREAKING ON UPGRADE**

! The `gliner` v1 package and the separate GLiREL relation model are removed; the local extraction stack now requires the `gliner2` package and GLiNER2.5 checkpoints (`fastino/gliner2.5-{small,base,multi}-v1`) — any code or configuration referencing the old `gliner` package or GLiREL will break on upgrade.

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

LangChain LangSmith

Feedback ⓘ snapshot-20261008 NOTES ↗

RANK

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

> LangSmith's biggest changes this window are breaking: extended trace retention on SaaS is now capped at 180 days, workspace role permissions were renamed, and Organization Operators lost some service-key role controls, alongside an LLM Gateway improvement for fallback model selection and small UI additions.

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

01 Provider-aware fallback model selection in LLM Gateway IMPROVED

70

LLM Gateway now selects fallback models using each configuration's provider and API preference, so Bedrock Claude models use the Messages API. The models actually served to a credential can be audited without making a real inference call via `GET /v1/models`.

Audit which models the LLM Gateway will actually serve to a given credential — without making a real inference call.

```
$ curl -s -H 'Authorization: Bearer <api_key>'
https://<langsmith-host>/v1/models
```

— Names mechanism and gives a runnable audit endpoint snapshot-20261008

02 Deep-linking to specific runs after SSO login NEW

70

SAML-authenticated users can now be redirected directly to a specific run after SSO sign-in instead of landing on the workspace root, using a `redirect_to` query parameter on the SSO login URL, e.g. `https://<langsmith-host>/sso/login/<slug>?redirect_to=/o/<org>/projects/p/<project>/r/<run>`.

Deep-link a SAML-authenticated user directly to a specific run after SSO sign-in instead of the workspace root.

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

— Exact URL pattern reader can use directly `snapshot-20261008`

THINNER COVERAGE BELOW

03 Service key Org Admin role restrictions for Operators BREAKING

47

Organization Operators can no longer grant the Organization Admin role to a service key, change the roles of a service key that already holds it, or allow a service key to change its own roles.

— Describes restriction but no API or config surface named `snapshot-20261008`

04 New gpt-6-astra model suggestion NEW

33

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

— Names model but only a dropdown suggestion `snapshot-20261008`

05 PII option in data policy dialogs NEW

28

Data policy dialogs now show a Personally Identifiable Information (PII) option under Data protection for every organization.

— Bare UI addition with no further mechanism `snapshot-20261008`

└──▷ BREAKING ON UPGRADE

! Extended-retention traces on LangSmith SaaS are now capped at 180 days; existing settings above 180 days are applied as 180 days for new traces, and new settings above 180 days are rejected.

! Built-in workspace roles no longer return the retired `repos:*` and `run-rules:*` permissions (renamed to `prompts:*` and `rules:*`); using the old names when creating a custom role is now rejected.

! Organization Operators can no longer grant the Organization Admin role to a service key, change the roles of a service key that already holds it, or allow a service key to change its own roles.

WAS THIS USEFUL?  

Braintrust

Feedback ⓘ snapshot-20261008 NOTES ↗ RANK

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

> Braintrust shipped SQL-based dashboard charts, optional expiration for API keys and service tokens, and a `cancel_event` mechanism to halt running evaluations, alongside breaking changes to `pi` model-call span fields, removal of `metadata.bt_daemon_version`, and a restructured .NET SDK tracing API.

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

01 pi tracing and model-call span field changes BREAKING

85

`bt` CLI v0.23.0 adds `pi` tracing that nests pi 1.0 codemode tool calls under the calling tool and records virtual model selections, request settings, and tool definitions on model call spans. On `pi` model call spans, `reasoning_tokens` is renamed to `completion_reasoning_tokens`, `cost` is renamed to `estimated_cost`, and `metadata.provider_request` is replaced by top-level fields such as `temperature` and `tools` (skill loads now use `tool_name: 'read'` with `tool_kind: 'skill'`); update any queries or dashboards that reference the earlier field names.

— Exact field renames given, no migration script provided snapshot-20261008

02 `cancel_event` to stop running evaluations NEW

85

Eval() and EvalAsync() in the Python SDK now accept a `cancel_event` (`threading.Event`) parameter, readable from `hooks.cancel_event`, that stops a running evaluation; a timeout also sets it.

Stop a long-running evaluation early from inside a task by checking `hooks.cancel_event`, so you can clean up resources before the evaluation rejects.

python

```python
async def my_task(input, hooks):
    for chunk in expensive_work():
        if hooks.cancel_event.is_set():
            break
        process(chunk)
```

Stop a long-running evaluation early from an external signal, such as a CI timeout handler.

```python
import threading
from braintrust import Eval

stop = threading.Event()

async def my_task(input, hooks):
    if hooks.cancel_event.is_set():
        return None
    return call_model(input)

await Eval('my-project', data=cases, task=my_task,
cancel_event=stop)
# elsewhere: stop.set() to halt the run
```

— Runnable code examples show exact parameter usage `snapshot-20261008`

03 **SQL-based dashboard charts** NEW 80

Dashboard charts can now be built from a SQL query, with a toggle between the SQL query text and a visual query builder that edits the same underlying query; Braintrust automatically injects the dashboard's timeframe, filters, and grouping at render time. SQL charts also work in the SQL sandbox and can be saved to a dashboard; the feature requires data plane v2.1.1+ and is available on Pro and Enterprise plans only.

— Version and plan requirements given but no exact navigation path `snapshot-20261008`

04 **Tracing plugin auto-update commands** IMPROVED 80

`bt trace update` now updates installed tracing plugins even when `bt` itself is already up to date, and `bt update` also updates installed tracing plugins, requiring a coding-agent restart to pick up the new plugin versions.

Update all tracing plugins in your coding agents to pick up the latest versions without waiting for a `bt` CLI update.

```
$ bt trace update
```

— Exact runnable commands given for both subcommands `snapshot-20261008`

05 **.NET SDK tracing builder API changes** BREAKING 80

Adds UseBraintrustLLMTracing() on `ChatClientBuilder` and moves UseBraintrustFunctionTracing() from `ChatClientBuilder` to `AIAgentBuilder` in the .NET SDK. ChatClientBuilder.UseBraintrustTracing() has been removed; existing code

must switch to UseBraintrustLLMTracing() on `ChatClientBuilder` and UseBraintrustFunctionTracing() on `AIAgentBuilder`.

— Exact method names given but no migration code sample `product docs`

06 **Copy project log rows into datasets** NEW 75

Project log rows can now be copied into datasets using the Braintrust MCP server or Loop, preserving input, metadata, tags, and source-log links. Requires data plane v2.18.0+.

— Version requirement given, no exact command shown `snapshot-20261008`

07 **Span customizers to redact PII** NEW 65

Spans can be customized at init time by registering a `span_customizers` callback, letting you redact sensitive PII (such as fields like `user_email`) from spans before they leave your environment.

Redact sensitive PII from spans before they leave your environment by registering a span customizer at init time.

ruby

```ruby
Braintrust.init(
  project: 'my-project',
  span_customizers: [
    ->(span) { span.metadata.delete('user_email'); span }
  ]
)
```

— Code example given but no accompanying doc bullet `product docs`

08 **Anthropic span cache-miss diagnostics** NEW 65

Anthropic spans now record the `diagnostics` request parameter along with `cache_miss_reason` and `cache_missed_input_tokens` in metadata when Anthropic reports a prompt cache miss.

— Field names given, no query example shown `snapshot-20261008`

09 **Coding agent insights dashboard** NEW 65

A Dashboards > Coding agent insights view shows cost and usage breakdowns for coding agent sessions across users, models, and branches, filterable by user or API key, alongside daily Loop automation findings for recurring inefficiencies.

View coding-agent cost and usage breakdowns for your team across users, models, and branches.

📍 1. In the Braintrust console, go to Dashboards › Coding agent insights. 2. Select the time range and filter by user or API key to see cost breakdowns by model and

branch. 3. Review the daily Loop automation findings for recurring inefficiencies across team sessions.

— Exact navigation path given, no API or flag named `snapshot-20261008`

10 **Expiring API keys and service tokens** NEW 60

API keys and service tokens now support optional expiration dates set at creation time and unchangeable afterward; expired tokens stop authenticating and cannot be renewed. A service token with an expiration can only create child service tokens that expire no later than it does, and attempts to create longer-lived child tokens now return a 403 error stating the maximum allowed lifetime.

— Enforcement described but no API endpoint or flag named `product docs`

11 **OpenAI Responses API span tracing additions** NEW 60

OpenAI Responses API spans now record `access_programs` in metadata, and trace computer use calls (capturing screenshots as attachments), browser authentication requests, and computer use approval items for Agents API sessions.

— Fields and scope named, no usage example shown `snapshot-20261008`

THINNER COVERAGE BELOW

12 **Open trace from dashboard table charts** NEW 50

Table charts that return raw spans from project logs now offer an Open trace action, which opens the exact trace and span behind the clicked row.

— Names the action but gives no navigation detail `product docs`

13 **`metadata.bt_daemon_version` removed from spans** BREAKING 50

Spans no longer include `metadata.bt_daemon_version`; update any queries or dashboards that read that field.

— Field named but no replacement field given `snapshot-20261008`

14 **`process_submission_result()` works without `run_id`** IMPROVED 50

process_submission_result() no longer requires `run_id` when an `external_id` is passed instead.

— Named function and parameter, no code sample given `snapshot-20261008`

15 **Loop-generated human review scores** NEW 45

Loop can now create human review scores from a plain-language description, including categorical choices, sliders, and free-form inputs.

— No command or UI path given for invoking it product docs

16 **Pattern alerts in Slack notifications** IMPROVED 45

Patterns Slack notifications now include individual alerts for new or reactivated patterns, alongside a digest of active patterns with frequency estimates.

— Describes behavior but no setup or config shown product docs

17 **Markdown notes on dashboards** NEW 35

Dashboards now support Markdown notes to explain what a dashboard tracks, link to runbooks, or annotate groups of charts.

— Thin description, no config or API surface named product docs

18 **Loop troubleshooting for scoring failures** NEW 35

Loop can now troubleshoot scoring failures using scorer and error context pulled from project logs.

— No steps given to trigger the troubleshooting flow product docs

19 **Workflow evaluation types in top-level package** IMPROVED 30

Workflow evaluation types can now be imported from the top-level package.

— No import path or example given product docs

BREAKING ON UPGRADE

- ChatClientBuilder.UseBraintrustTracing() has been removed; replace it with UseBraintrustLLMTracing() on `ChatClientBuilder` and UseBraintrustFunctionTracing() on `AIAgentBuilder`.
- On pi model call spans, `cost` metadata field `metadata.provider_request` is replaced by top-level fields (e.g. `tool_name: 'read'`, `tool_kind: 'skill'`); update any queries that reference the earlier fields.
- Spans no longer include `metadata.bt_daemon_version`; update any queries or dashboards that read that field.
- A `timeout` of `0` now times out immediately in JS/TS workflow evaluations.
- On `pi` model call spans, `reasoning_tokens` is renamed to `completion_reasoning_tokens`, `cost` is renamed to `estimated_cost`, `metadata.provider_request` is replaced by top-level fields such as `temperature` and `tools`, and skill loads now use `tool_name: 'read'` with `tool_kind: 'skill'` — update any queries that reference the earlier field names.
- Spans no longer include `metadata.bt_daemon_version`.

WAS THIS USEFUL? 👍 👎

GOVERN

◆ AI/LLM Security

AI-Infra-Guard

Feedback ⓘ v4.6.5 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 v4.6.5 adds a new Agent-Scan provider and SSE transform support, MCP protocol auto-negotiation, the ability to resume interrupted DeepTeam runs, and 271 new CVE detection rules.

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

01 **MCP-Scan protocol auto-negotiation** NEW 50

MCP-Scan now auto-negotiates between modern and legacy MCP protocol versions and records the negotiated protocol version for the scan.

– Explains mechanism but no config/flag named v4.6.5

02 **Agent-Scan provider and SSE transform support** NEW 35

Agent-Scan adds `OrcaRouter` as a first-class provider and now supports custom transforms for SSE responses.

– Names surfaces but no mechanism or usage detail v4.6.5

03 **Resume interrupted DeepTeam runs** NEW 30

AI-Infra-Guard can now resume an interrupted DeepTeam red team run instead of restarting it from scratch.

– States capability but no command or flag given v4.6.5

04 **271 new CVE detection rules** NEW 30

This release adds 271 new CVE rules to the vulnerability scanning engine.

– Concrete number but no rule list or scope given v4.6.5

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Frontier Models

Anthropic

Feedback ⓘ snapshot-20261008 NOTES ↗ RANK

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

> Anthropic shipped beta SDK tool-use base classes for browser and computer automation, tightened URL fetching rules in Managed Agents, and expanded Dreams research preview model support.

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

01 **Stricter web_fetch URL policy in Managed Agents** BREAKING

73

In Claude Managed Agents, `web_fetch` is now restricted to URLs already seen in the session — from user messages, `web_search` results, or prior `web_fetch` responses. Calls for URLs appearing only in Claude's own output, the system prompt, attached documents, or tool outputs such as `bash`, `read`, or MCP tools now return a new `url_not_in_prior_context` error to the agent instead of fetching, which breaks agents that previously relied on fetching such URLs.

– Names exact error code and all excluded source types snapshot-20261008

02 **BrowserUseTool and ComputerUseTool SDK base classes** NEW

70

Adds beta `BrowserUseTool` and `ComputerUseTool` base classes to the Python and TypeScript SDKs. Developers subclass one and implement a single method per tool, while the SDK manages the tool loop, URL/file policies, and the approval callback.

– Names exact classes and SDK behavior but no code sample snapshot-20261008

THINNER COVERAGE BELOW

03 **Dreams model support expansion** IMPROVED

45

Dreams (research preview) now supports Claude Opus 5.5, Claude Fable 5.1, and Claude Sonnet 5.5.

– Names models added but no further mechanism detail snapshot-20261008

└──▷ BREAKING ON UPGRADE

! In Claude Managed Agents, `web_fetch` now returns a `url_not_in_prior_context` error for any URL that appears only in Claude's own output, the system prompt, an attached document, or the output of tools such as `bash`, `read`, or MCP tools — agents that previously fetched such URLs will now receive an error instead.

WAS THIS USEFUL? 👍 👎

Anthropic Claude Haiku 5.5 Feedback ⓘ

released · Oct 8, 2026 MODEL ↗ DOCS ↗ BLOG ↗

Anthropic: "Claude Haiku 5.5 is our fastest, most capable small model. Built for high-volume work like summarization, subagents, and browser use."

Anthropic says: "On average, it now costs around 75% less to run.[2]"

Anthropic says: "Haiku 5.5 is especially good value when used for tasks with prompts up to 100,000 tokens, which make up around 90% of requests to our previous Haiku model."

Claude Haiku 5.5 from Anthropic was released on 7 Oct 2026. 1M-token context; $0.10 in / $0.50 out per million tokens (OpenRouter).

Succeeds Claude Haiku 4.5.

Claude Haiku 5.5 launches with adaptive thinking, 1M token context, 128K output, and tiered pricing starting at $0.10/MTok.

+40 more

└──▷ BREAKING ON UPGRADE

- ! `claude-sonnet-4-5` and `claude-sonnet-4-5-20250929` reach end-of-life on November 30, 2026; callers must migrate to `claude-sonnet-5-5`.
- ! Code written for Claude Haiku 4.5 can break on Claude Haiku 5.5: manual extended thinking returns a 400 error, adaptive thinking is on by default so responses can begin with thinking blocks, and the same text counts as more tokens.
- ! In Claude Managed Agents `limited` networking, creating a session fails with a 400 error when an enabled web tool's configuration has an entry not within `allowed_hosts`; a session update that adds such an entry also fails. `docs.example.com` is not within `['example.com']` unless `*.` prefix is used.
- ! Claude Sonnet 4.5 is deprecated with retirement on the Claude API scheduled for November 30, 2026.

WAS THIS USEFUL? 👍 👎

◆ Open-Weight Models

DeepSeek V4 Flash 0731 Feedback ⓘ

price increase · Oct 7, 2026 MODEL ↗

DeepSeek V4 Flash 0731 is a chat model from DeepSeek.

Price increase on OpenRouter (deepseek/deepseek-v4-flash-0731): $0.0152 → $0.018 in, $1.28 → $1.28 out per million tokens (+18%), first seen 7 Oct 2026.

WAS THIS USEFUL? 👍 👎

Z.ai GLM 5.1 Feedback ⓘ price cut · Oct 7, 2026 MODEL ↗

GLM 5.1 is a chat model from Z.ai.

Price cut on OpenRouter (z-ai/glm-5.1): $1.40 → $0.966 in, $4.40 → $3.036 out per million tokens (-31%), first seen 7 Oct 2026.

WAS THIS USEFUL? 👍 👎

Moonshot AI Kimi K2.6 Feedback ⓘ price cut · Oct 7, 2026

MODEL ↗

Kimi K2.6 is a multimodal model from Moonshot AI.

Price cut on OpenRouter (moonshotai/kimi-k2.6): $0.95 → $0.465 in, $4.00 → $2.45 out per million tokens (-51%), first seen 7 Oct 2026.

WAS THIS USEFUL? 👍 👎

NVIDIA Nemotron 3.5 Lightning

Feedback ⓘ

price cut · Oct 7, 2026 MODEL ↗

Nemotron 3.5 Lightning is a chat model from NVIDIA.

Price cut on OpenRouter (nvidia/nemotron-3.5-lightning): $0.06 → $0.049 in, $0.16 → $0.14 out per million tokens (-18%), first seen 7 Oct 2026.

WAS THIS USEFUL? 👍 👎

Alibaba Qwen3.6 27B

Feedback ⓘ price cut · Oct 7, 2026

MODEL ↗

Qwen3.6 27B is a multimodal model from Alibaba.

Price cut on OpenRouter (qwen/qwen3.6-27b): $0.32 → $0.30 in, $3.20 → $2.00 out per million tokens (-38%), first seen 7 Oct 2026.

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · October 8, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue