THE AI TOOLCHAIN

NO. 060

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED OCTOBER 13, 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 183 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 40 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.

BUILD

Run RL and preference tuning against a hosted control plane instead of standing up your own trainer

vim ? help

Training sessions, GPU resources and checkpoints are now addressable primitives, with GRPO, DPO and SFT pipelines and FP4 quantisation on top. The usual week of cluster plumbing and checkpoint babysitting before the first reward curve goes away, and multi-LoRA deployment means variants land behind one served model rather than one endpoint each.

Together AI

DEPLOY

Scale query serving without touching the write path, and resume a dropped stream where it stopped

Pinecone's dedicated read nodes take configurable scaling plus a recall/latency knob, so a read-heavy RAG surface stops being sized by its ingest load. Both Google SDKs add continuation-token resumption for streaming and chat — a network blip on a long generation no longer means paying for the whole response twice.

Pinecone · Google Gen AI Python SDK · Google Gen AI JavaScript SDK

OPERATE

Cap token spend inside the pipeline rather than finding it on the invoice

OntoCast enforces a run-level token meter with hard caps and a dedicated error, plus per-document limits on the processing endpoints and separate accounting for cache-replayed tokens, so a runaway document cannot quietly consume a month's budget. AgentCore Memory's TTL expiry purges stale long-term records automatically instead of needing a cleanup job.

OntoCast · Amazon Bedrock AgentCore

BUILD

◆ AI Coding Agents

Cline

Feedback ⓘ sdk/sdk/v0.0.93 NOTES ↗ CODE ↗

Cline is an open-source AI coding agent for VS Code that can read and edit files, run terminal commands, and use browsers with user approval.

In-process plugin loading with state tracking, Composio meta-tool auth, host-authenticated connector requests, and per-image token estimation.

└──▷ GET THIS VERSION

```
$ git clone --branch sdk/sdk/v0.0.93
https://github.com/cline/cline.git
# already have the repo? check out this version:
$ git checkout sdk/sdk/v0.0.93
```

Sets `CLINE_PLUGIN_MODE=sandbox` to keep the legacy subprocess sandbox while plugins otherwise load in-process through one registry per process.

The `session.create` call now accepts a `plugins` policy field and returns a `pluginIssues` field when plugins have problems.

The hub now answers `plugins.list`, `plugins.status`, and `plugins.reload` and broadcasts `plugin.status_changed` events.

Adds `connectorsRequest` option (`ConnectorsRequest` exported from `@cline/core`) so hosts can supply their own Composio authentication, forwarded to scheduled sessions too.

New `createImageAwareReplacer` export in `@cline/shared` pairs with updated `estimateRequestInputTokens` and `createTokenEstimator` that bill each image at a flat `ESTIMATED_TOKENS_PER_IMAGE` (1600 tokens) instead of the text rate.

+3 more

NEW INTEGRATION AI/ML API provider

└──▷ BREAKING ON UPGRADE

! `resolveComposioToolsStatePath` is renamed to `resolveComposioConnectionStatePath`; any code calling the old name will break.

! Connector tools are now discovered through Composio meta search; saved desktop connection state no longer supplies model tools, so setups relying on that path will stop receiving connector tools.

WAS THIS USEFUL?  

yetone magpie

Feedback ⓘ v0.1.1224 CODE ↗ RANK

Magpie is 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.

> Across six releases magpie added a glob-pattern model blocklist and a self-reported usage API for apps that call AI vendors directly, alongside weighted provider-group routing, broad session/usage ingestion for Cursor, OpenCode, Kimi Code, Goose, Claude Desktop and WSL omp sessions, Library skills management improvements, and Homebrew/Microsoft Store based agent version tracking.

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

01 **Glob-pattern model blocklist** NEW 100

Adds `DisabledModels` to `settings.json`, accepting exact IDs or globs (e.g. `*fable*`, `anthropic/claude-fable-5*`) matched case-insensitively against `<provider>/<model>`, bare ID, and bare ID's last segment — a disabled model is removed from all catalogs and refused at the gateway before any upstream call, across every route: direct requests, routing groups, fallbacks, sub-agent models, `count_tokens`, and decision models.

Block all Claude models from a specific provider so they never appear in listings or receive any request traffic.

```json
{ "DisabledModels": ["anthropic/*", "*claude-fable-5*"] }
```

— Exact config key, matching rules, scope, and runnable example. v0.1.1219

02 **Self-reported usage API endpoint** NEW 95

Adds `POST /v1/magpie/usage/requests` so apps that call their AI vendor directly (not through magpie) can report those calls to magpie's Usage page, with backfill support, deduplication by `rid`, auto-pricing, and a `?via=reported` filter; batches are all-or-nothing and a bad record returns a `400` naming the field.

Filter the Usage view to show only self-reported (non-gateway) calls when reviewing externally-made AI spend.

```
$ curl 'http://127.0.0.1:3425/v1/magpie/usage?via=reported'
```

— Full endpoint spec, params, error behavior, runnable example. v0.1.1220

03 **Codex multi_agent_v2 override warning** NEW 80

Magpie now warns when Codex's `features.multi_agent_v2` is enabled in its config and overrides magpie's `settings.CodexAgentsV1` multi-agent V1 routing, so users know to turn it off in Codex's config rather than silently losing subagent delegation.

— Names both exact config keys and the remediation. v0.1.1223

04 **Grok App independent mode with agent-home** NEW 80

Selecting a custom provider switches Grok App from the shared `~/.grok` config to its own agent-home, where the Library writes `AGENTS.md`, `[mcp_servers]`, and `skills/`.

— Names exact paths and files written by the Library. v0.1.1218

05 **Agent version detection and updates via Homebrew and Microsoft Store**

IMPROVED 75

Magpie detects Grok Build from Homebrew's `grok-build` cask, shows its version, and updates it with `brew`; Homebrew Node formula agents (`gemini-cli`, `pi`, `cline`) now update via `brew upgrade` instead of npm; and the Windows Codex app's version is shown on the Agents page, read from the Microsoft Store package registered for the user.

— Names exact package identifiers and update commands. v0.1.1224 v0.1.1219

06 **Codex session read limit raised to 512MB** IMPROVED 75

Raises the read limit for long Codex sessions full of images to 512 MB, and returns a magpie `413` error that names magpie as the limit source and directs the user to `/compact`.

— Names exact limit, error code, and remediation path. v0.1.1220

07 **Library skills management: multi-select, grouping, collapse controls**

IMPROVED 70

The Library's skills area gains bulk actions — selecting and removing multiple agent skills together — plus `Collapse all` / `Expand all` controls for skill groups, and a site's built-in skill set is now grouped into a single skill group keyed by `sourceBaseUrl`.

— Names UI controls and grouping key, lacks any limits. v0.1.1218 v0.1.1222

08 **KDE Plasma Wayland clipboard fix** IMPROVED 70

Clipboard now works correctly on KDE Plasma Wayland, including when `EnablePrimarySelection=false` is set in `kwinrc` under `[Wayland]`.

— Names the exact config file and key for the fix. v0.1.1222

09 **Qwen Code library instructions support** NEW 65

Adds `~/.qwen/QWEN.md` support: the Library Instructions tab now lists Qwen Code and writes its instructions file.

— Names the exact file path and UI tab. v0.1.1224

10 **ZCode quota window count alignment** IMPROVED 65

ZCode quota windows now report their count as `used` or `remaining` consistent with the adjacent percentage, aligning `QuotaWindow.Count` with `amount` / `limit` fields for Start Plan buckets and GLM Coding Plan limits.

— Names exact fields; purely a display consistency fix. v0.1.1223

11 **Routing key management improvements** IMPROVED 65

On Routing and in the editor, magpie now reports a key set to a protocol its provider has no URL for rather than silently skipping it, lets every resting key or account be retried by hand from Routing and the provider's key list, and allows changing a key's value in place while keeping its name, position, and settings.

— Describes three behaviors but no exact flag or field names. v0.1.1220

12 **Cursor and OpenCode session reading** NEW 60

Reads `Cursor` editor history and `Cursor` / `OpenCode` conversations into the Sessions view, including search and system-prompt suppression for Cursor chats.

— Names the agents and view but no exact navigation. v0.1.1220

THINNER COVERAGE BELOW

13 **Weighted routing for provider groups** NEW 50

Provider groups gain a `By weight` routing mode, with each member's share set beside it.

— Names the mode but mechanism beyond share value is thin. v0.1.1224

14 **omp session listing across WSL distros** NEW 50

Magpie now lists omp sessions running inside a WSL distro on Windows, surfaced on the Sessions page and in the Usage Sessions tab.

— Names the exact pages but not deeper mechanism. v0.1.1224

15 **Per-agent daily check-in toggle** NEW 50

Each WorkBuddy, Trae CN, MiniMax, Qoder, or plugin account's daily check-in can now be turned off individually.

— Names account types but not the toggle location. v0.1.1224

16 **Claude Code project folder relocation tracking** IMPROVED 50

A renamed or moved Claude Code project folder can now be relocated along with its sessions, from the Sessions page or the CLI.

— Gives UI/CLI path but no exact command. v0.1.1224

17 **musl Linux plugin support** NEW 50

Plugins now run on Bun's musl build on musl Linux (Alpine), kept separate from the glibc build.

— Names the runtime and platform but no config surface. v0.1.1220

18 **Claude Desktop conversation titles in stats** NEW 50

Reads Claude Desktop's saved conversation titles into stats and Context, refreshed once per second from every handle Desktop keeps.

— Names refresh cadence but no UI path. v0.1.1220

19 **Gateway image-model key rotation fallback** IMPROVED 50

Gateway image-model key rotation now reads a relay's full model list and falls back to the next key on a 404 response for the model.

— Explains the mechanism but scope detail is limited. v0.1.1218

20 **Request log service tier visibility** NEW 45

Each request's log row now shows the service tier asked for, the tier each attempt was actually sent at, and the tier the vendor reported back.

— Describes three tier fields but no exact UI path. v0.1.1224

21 **Kimi Code and Goose usage integrations** NEW 45

Reads Kimi Code 's own calls into Usage from its wire files, and shows Goose 's requests as Goose on Routing and in Usage.

— Thin one-line integrations for two agents. v0.1.1220

22 **Connect snippet platform selection** NEW 45

Adds platform selection to Connect's Shell and `curl` snippets, letting users choose Windows or macOS/Linux for generated connection code.

— Names the snippet types but not output content. v0.1.1218

23 **Plugins page filtering for unofficial GitHub plugins** NEW 45

Adds a kind filter and a dedicated narrowing field to the Plugins Unofficial, on GitHub section.

— Names the UI section but not filter values. v0.1.1218

24 **Multi-protocol provider model catalog merge** IMPROVED 40

A multi-protocol provider's model catalog now merges every endpoint's models, so models from all protocols remain visible.

— Explains the fix but no named config surface. v0.1.1224

25 **Enterprise WorkBuddy credit display** NEW 40

An enterprise WorkBuddy account's credits from the enterprise meter are now shown on its provider card.

— States what's shown but not how credits are computed. v0.1.1224

26 **Forced HTTP/1.1 for ChatGPT backend** IMPROVED 40

Forces all requests to the ChatGPT backend over HTTP/1.1, including magpie's own requests, not only gateway turns.

— States the change but not the underlying reason. v0.1.1220

NEW INTEGRATION DevEco Code (Huawei's OpenCode) agent

v0.1.1221 CODE ↗ STABLE

Adds bulk agent visibility management, Ollama Cloud quota display, proxy credential redaction in logs, and per-model Codex thread descriptions.

└──▷ GET THIS VERSION

```
$ git clone --branch v0.1.1221 https://github.com/yetone/magpie.git
# already have the repo? check out this version:
$ git checkout v0.1.1221
```

Adds a `Manage` button in the Agents list footer (`#agentsManage`) to bulk-hide or unhide many agents at once without touching agent configs.

Displays Ollama Cloud plan's included allowance and purchased credits in Usage and magpie quota views.

Keeps proxy usernames and passwords out of all log lines, error messages, and status output that reference a proxy.

Sends Codex thread descriptions to a separately chosen model in Settings, distinct from the model used for thread titles.

Lets `models` and `visible` CLI subcommands find a WSL agent by its ID in any letter case.

WAS THIS USEFUL? 👍 👎

Diagram Design

Feedback ⓘ changes since 2026-10-05 CODE ↗

RANK

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

> Diagram Design expanded its Mermaid input handling and added richer export options, including a self-contained HTML default and an SVG format with embedded, portable CSS.

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

01 **HTML and SVG diagram export** NEW 65

Diagrams export as self-contained HTML by default, or as SVG including CDATA-wrapped diagram CSS with selectors bound to the original SVG root, intended for use in Figma, slides, or social cards.

— Names export formats and CSS binding mechanism, no exact command product docs

THINNER COVERAGE BELOW

02 **Mermaid file and fenced block input support** NEW 55

Diagram Design now accepts `.mmd` and `.mermaid` files as well as fenced `mermaid` code blocks in Markdown as diagram inputs, and supports UTF-8 files with a byte-order mark (BOM) for Mermaid input.

— Names exact file types and BOM handling but no limits product docs

changes since 2026-10-05 CODE ↗

Adds exploded axonometric, axonometric plan, and service blueprint diagram types to Diagram Design.

└──▷ **GET THIS VERSION**

```
$ git clone --branch commits-2026-10-05
https://github.com/cathrynlavery/diagram-design.git
# already have the repo? check out this version:
$ git checkout commits-2026-10-05
```

Adds the service blueprint visual type, with `scripts/verify-service-blueprint.py` enforcing layer order, visibility-line placement, stage-grid alignment, and contrast invariants.

> Ships new axonometric examples: AI agent stack, keyboard, coffee shop, and warehouse.

WAS THIS USEFUL? 👍 👎

claude-mem Feedback ⓘ v13.37.0 NOTES ↗ RANK

Gives coding agents persistent memory by capturing session activity, compressing it with a model, and injecting relevant context into later sessions.

> claude-mem's only change this window hardens memory capture against provider outages by giving the quota fallback a sane default and broader trigger conditions.

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

01 Default fallback provider for memory capture BREAKING

88

`CLAUDE_MEM_QUOTA_FALLBACK_PROVIDER` now defaults to `claude` (previously `""`), so memory capture keeps running when the primary provider is rate-limited after retries, over capacity, fully down, or returns the same batch twice with no usable output — not just on quota exhaustion. Set it to `off` to disable the fallback and wait for the primary provider to recover instead; existing settings files that had saved `""` will silently pick up the new `claude` default on upgrade.

Prevent automatic fallback to Claude and instead wait for your primary provider to recover before resuming memory capture.

```
$ CLAUDE_MEM_QUOTA_FALLBACK_PROVIDER=off npx claude-mem install
```

— Names exact env var, default values, triggers, and a runnable command. v13.37.0

└──▷ **BREAKING ON UPGRADE**

! `CLAUDE_MEM_QUOTA_FALLBACK_PROVIDER` now defaults to `claude` instead of `""`. Settings files that previously saved `""` will receive the new default on upgrade, silently enabling Claude as the fallback provider.

WAS THIS USEFUL? 👍 👎

Letta Code

Feedback ⓘ v0.34.13 NOTES ↗ RANK

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

> Letta Code v0.34.13 adds several agent-autonomy features: agents can now write binary files, manage their own webhooks, sign in via agent-issued tokens, run OpenCode subagent workers, and benefit from a backup wake trigger for silent monitors.

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

01 **Base64 binary file writes via write_file** NEW 50

The listener's `write_file` calls now support `base64` encoding, enabling agents to write binary files rather than being limited to text content.

— Names the exact flag but no further mechanism detail. v0.34.13

02 **Agent-issued connect token sign-in** NEW 40

The server now supports agent-issued connect token sign-in, allowing authentication via tokens generated by an agent.

— Describes the mechanism but no endpoint or flag given. v0.34.13

03 **OpenCode subagent workers** NEW 40

Letta Code now supports OpenCode coding workers as subagents, expanding the pool of multi-agent coding workers available.

— Names the integration but no setup steps given. v0.34.13

04 **Agent-managed webhooks as skills** NEW 35

Agents can now manage their own webhooks as skills, giving them self-service control over webhook registration.

— Clear capability but no concrete API or config surface named. v0.34.13

05 **Backup Wake trigger for silent monitors** NEW 35

Adds an optional backup Wake trigger for the monitor that fires when a watch goes silent, preventing silent watch failures.

— Explains purpose but lacks config key or exact trigger name. v0.34.13

WAS THIS USEFUL? 👍 👎

Token Optimizer

Feedback ⓘ v5.13.39 NOTES ↗ RANK

Token Optimizer is a coding-agent plugin that trims wasted context, checkpoints work across sessions and compactions, and reports token spend on a live dashboard.

> Token Optimizer's v5.13.39 release adds a beta Devin CLI compression skill, improves Codex host inspection checks, and surfaces compaction/session status in the Doctor and Hooks UI.

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

01 **Devin CLI beta compression skill** NEW 55

An optional, manually triggered Devin CLI (beta) skill shrinks large command output while keeping the original recoverable on the local machine. It introduces no automatic hooks or permission changes.

— Names the skill and behaviour but no exact command shown v5.13.39

02 **Compaction and tracking status in Doctor and Hooks tab**

IMPROVED 50

Doctor and the Hooks tab now show Smart Compaction and Session Tracking as enabled for plugin installs.

— Names exact UI locations, lacks deeper mechanism v5.13.39

03 **Codex host session reporting and rewrite checks** IMPROVED 41

The Codex host now reports the session that requested quality and state checks, and uses the same inspection-only check for optional command rewrites as all other hosts.

— Describes behaviour change without a runnable surface v5.13.39

WAS THIS USEFUL? 👍 👎

Earendil Works Pi

Feedback ⓘ v1.2.0 NOTES ↗ RANK

> Pi v1.2.0 adds OSC 52 remote-clipboard support, published config schemas, per-key OpenRouter model filtering, and several extension/codemode API additions, alongside breaking changes to agent-start event semantics and MCP resource reporting.

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

01 **Published JSON schemas for config files** NEW 95

Adds `$schema` support to `settings.json`, `models.json`, keybindings, and themes config files for editor validation and completion, with schemas published at `https://pi.dev/schemas/*.schema.json`. Breaking: the theme schema URL moved to `https://pi.dev/schemas/theme.schema.json`, so theme `$schema` references must be updated, reusable colors defined under `vars`, and unsupported metadata removed.

Add editor schema validation and autocomplete to your Pi settings file so your editor flags unknown keys and suggests values.

json

```
{
  "$schema": "https://pi.dev/schemas/settings.schema.json",
  "theme": "dark"
}
```

— Names files, URL pattern and migration step for themes v1.2.0

02 **Per-key OpenRouter model filtering** NEW 90

With an OpenRouter key, Pi lists only the chat models that key may use, showing the key's context size, max output, and prices, including regional endpoints configured as the base URL (`baseUrl`) in `models.json`; the list is saved and used offline.

Pin a regional OpenRouter endpoint and let Pi surface only the models your API key can access, with accurate pricing and context limits.

json

```
{
  "$schema": "https://pi.dev/schemas/models.schema.json",
  "openrouter": {
    "baseUrl": "https://openrouter.ai/api/v1"
  }
}
```

— Config key and behaviour fully specified v1.2.0

03 **Session busy during run start** BREAKING 85

The session now counts as busy while a run starts, including during `before_agent_start` handlers: ctx.isIdle() returns false, pi.sendMessage() queues into the starting run, and pi.sendUserMessage() now requires `deliverAs`.

— Clear migration note naming affected methods v1.2.0

04 **OSC 52 remote clipboard forwarding** NEW 80

Setting `PI_CLIPBOARD_OSC52=1` sends copied text via OSC 52 so it reaches a remote client's clipboard when Pi cannot detect the remote session, e.g. a multiplexer attached from another machine or a remote IDE daemon.

Enable OSC 52 clipboard forwarding when running Pi inside a tmux or screen session attached from a remote machine.

```
$ PI_CLIPBOARD_OSC52=1 pi
```

— Exact env var and runnable command given v1.2.0

05 **MCP resource reporting simplified in list output** BREAKING 80

`pi mcp list --json` now reports `resources: true` for servers that offer resources instead of separate `resources` and `resourceTemplates` counts; Pi no longer lists resources when connecting.

— Exact command and field change named v1.2.0

06 **before_agent_start triggered by custom turns** IMPROVED 75

`before_agent_start` now also fires for runs started by pi.sendMessage(..., { triggerTurn: true }). `BeforeAgentStartEvent` is now a union on `messageType`: `'user'` events carry `prompt` and `images`, `'custom'` events carry the starting `message` and no `prompt`; handlers that should act only on user prompts must check `event.messageType === 'user'`.

— API union documented but no code sample v1.2.0

07 **New tool definition and namespace fields** NEW 60

Adds `reachesTools` to tool definitions so tools that reach other tools, such as `codemode` and `tool_search`, declare which tools a call may reach, and adds

`aliases` to tool namespaces, matched by codemode's describeNamespace() and the `namespace` option of searchTools(), with MCP namespaces using the server name.

— Names the field but no usage example Names APIs affected but no example v1.2.0

THINNER COVERAGE BELOW

08 Codemode tool declaration and display improvements IMPROVED

55

Codemode tool declarations now show input constraints (integer, minimum/maximum, length, pattern, default) as `@` tags on property comments, and codemode tool rows now display elapsed time while a script runs and total time once finished.

— Two thin, related codemode UI additions grouped v1.2.0

└──▷ BREAKING ON UPGRADE

! The theme schema URL moved to `https://pi.dev/schemas/theme.schema.json`; update any theme `$schema` references, define reusable colors under `vars`, and remove unsupported metadata.

! `BeforeAgentStartEvent` is now a union on `messageType`: `'user'` events carry `prompt` and `images`, `'custom'` events carry the starting `message` and no `prompt`. Handlers that should act only on user prompts must check `event.messageType === 'user'`.

! The session now counts as busy while a run starts (including during `before_agent_start` handlers): ctx.isIdle() returns false, pi.sendMessage() queues into the starting run, and pi.sendUserMessage() now requires `deliverAs`.

! `pi mcp list --json` now reports `resources: true` for servers that offer resources instead of separate `resources` and `resourceTemplates` counts; Pi no longer lists resources when connecting.

WAS THIS USEFUL? 👍 👎

Graphify Labs Graphify

Feedback ⓘ v0.9.88 NOTES ↗

Graphify converts code, docs, SQL schemas, configs, and PDFs into a queryable knowledge graph with explainable relationships.

Adds `--seed` to `graphify query` for external-ranker-controlled traversal start sets, and hover tooltips in the `graph.html` sidebar.

└──▷ **GET THIS VERSION**

```
$ git clone --branch v0.9.88 https://github.com/Graphify-Labs/graphify.git
# already have the repo? check out this version:
$ git checkout v0.9.88
```

Adds `--seed` to `graphify query` to start graph traversal at specific nodes (ids, labels, callable names, or paths), enabling external rankers to choose the start set; unresolved seeds are reported and skipped, and an all-unresolved set fails closed.

Adds hover tooltips in `graph.html` sidebar showing the full truncated legend/search/neighbor name and the legend node count.

Java call resolution now traces through chained, `new`, and cast receivers (e.g. factory.builder().size(3).build(), new Builder().size(1), ((Builder) o).build()) by following declared return types on the method chain.

NEW SUPPORTED PLATFORM EJS template extraction

└──▷ **BREAKING ON UPGRADE**

! The AST cache schema is bumped; the first `graphify` build after upgrading to v0.9.88 will re-extract the entire project from scratch.

WAS THIS USEFUL? 👍 👎

Sourcegraph Amp

Feedback ⓘ

First Things First, Faster (2026-10-12) BLOG ↗ RANK

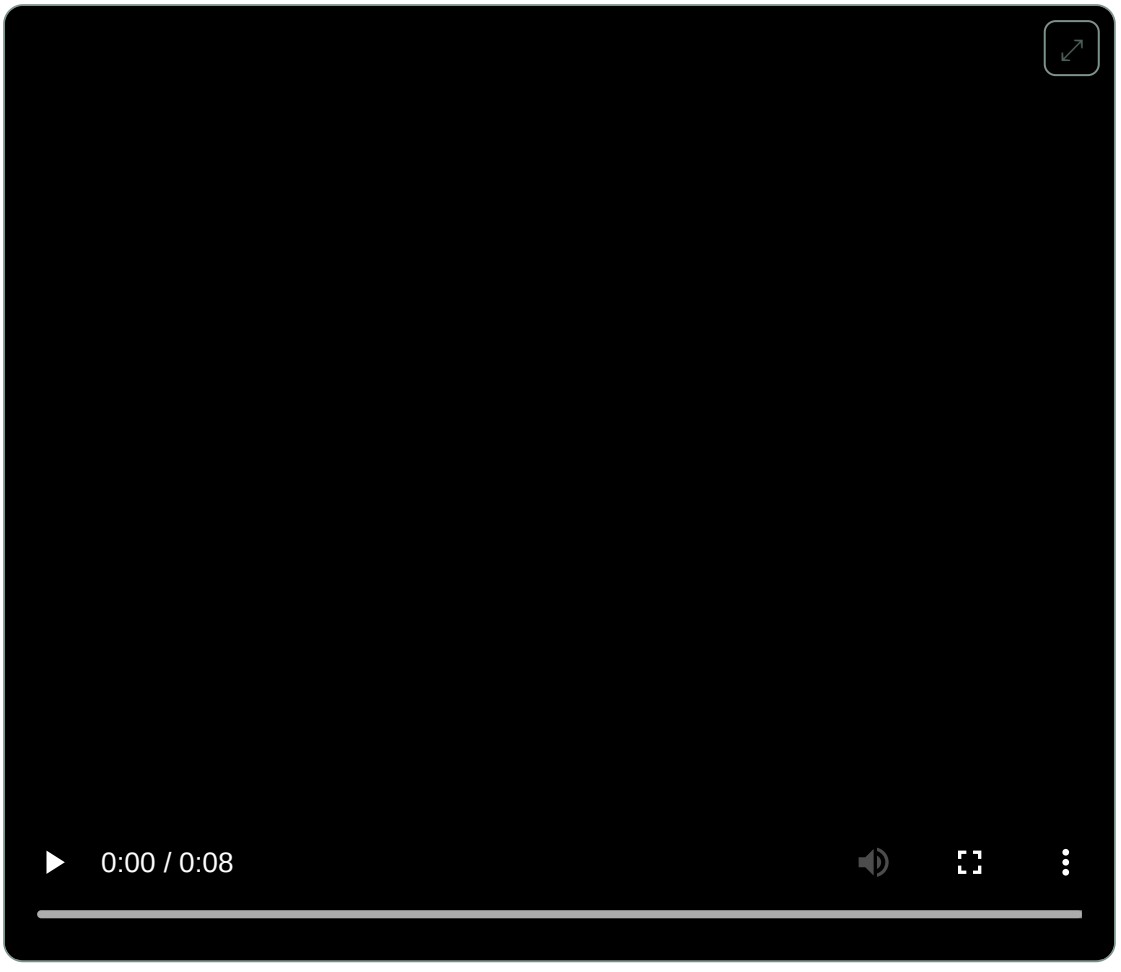
Amp is Sourcegraph's agentic coding tool for the terminal and editor, running multi-step edits with subagents and shared team threads.

> Amp's Changes pane now intelligently reorders files by relevance to the diff, with live re-ranking and an alphabetical fallback mode.

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

01 **Intelligent file ordering in Changes pane** NEW 58

The Changes pane automatically reorders files by relevance to the diff, putting essential files first without disrupting an in-progress review. A blue dot appears on the ordering button when the diff changes, letting you apply the updated order on demand rather than having your review interrupted. An optional alphabetical file ordering mode is also available, accessible by clicking the ordering button.



— Describes behavior and UI control but no config/flag names

First Things First, Faster (2026-10-12)

WAS THIS USEFUL? 👍 👎

Amazon Kiro

Feedback ⓘ

CLI: Switch to V3 at Chat Start and Model Fallback Notices (2026-10-12) BLOG ↗

RANK

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

> Kiro CLI began rolling out V3 with an opt-in flag, a gradual switchover prompt, and transcript fallback notices, while Kiro Crew 0.8 added Crewmate Teams, an Agent Templates tab, new monitoring and memory config keys, new kirocrew subcommands, a DeepSeek backend, and several breaking changes to default logging, MCP auto-approval, and cron validation.

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

01 Monitoring loop controls: longer watches and quiet check-ins

IMPROVED 91

The `monitoring.max_runtime_secs` config key raises the pull-request watch ceiling from the default seven days up to thirty days. A new Quiet check-ins feature, enabled under Settings › Developer › Feature Previews › Decisions (Jev), makes routine monitoring checks cost no model turn and only wakes the session when review or comment activity requires it.

Extend a pull-request watch to run for up to 30 days instead of the default 7, so long-running reviews are not silently dropped.

yaml

```
monitoring.max_runtime_secs: 2592000
```

Enable quiet monitoring check-ins so routine loop checks cost no model turn and only wake the session on reviewer requests.

📍 In Kiro Crew, go to Settings › Developer › Feature Previews › Decisions (Jev) and enable Quiet check-ins: wake or skip .

— Gives config key, exact day limits, and exact UI path to enable.

Crew: Crewmate Teams, Agent Templates, …

02 Default-on crew event logging and CLI binary override

80

BREAKING

Kiro Crew now records a redacted event log for every chat by default before starting the Gateway; set `KIROCREW_CREW_LOG=0` to disable it. The separate `KIROCREW_KIRO_BIN` environment variable lets operators override the path to the bundled Kiro CLI binary.

— Names both env vars and the exact value to disable the new default.

Crew: Crewmate Teams, Agent Templates, …

03 **MCP autoApprove enforcement change** BREAKING 80

Owner-written MCP `autoApprove` lists in `mcp.json` or agent files now take effect by default, bypassing the approval card and Kiro Crew's tool gate. Set the new `mcp.honour_auto_approve` config key to `false` and restart the Gateway to keep those tools behind the gate as before.

— Names exact config key, files, and restart step to revert behavior.

Crew: Crewmate Teams, Agent Templates, …

04 **Stricter command cron validation** BREAKING 80

Command cron validation is now stricter: stored commands using brace expansion fail until edited, imports skip those jobs, and quoted regex intervals such as `grep -E '[0-9]{1,3}'` are refused — use a script cron for shell features outside the command-cron subset.

— Gives exact failing pattern and the workaround to use a script cron.

Crew: Crewmate Teams, Agent Templates, …

05 **V3 opt-in flag and switchover prompt** NEW 75

Kiro CLI adds a `kiro-cli --v3` flag to try V3 sessions on demand ahead of the default switch in CLI 3.0, plus a gradually-rolling-out startup prompt that offers to switch to V3, upgrade configs, enable automatic agent upgrades, and begin migrating custom agent configurations.

Opt into V3 immediately without waiting for the gradual rollout prompt.

```
$ kiro-cli --v3
```

— Names exact flag and prompt actions, lacks full migration mechanics.

CLI: Switch to V3 at Chat Start and Mod…

06 **New kirocrew subcommands: cron add and doctor** NEW 70

`kirocrew cron add` creates agent, script, command, and one-shot scheduled jobs, while `kirocrew doctor` reports stale skills.

— Names runnable subcommands and job types but no flags shown.

Crew: Crewmate Teams, Agent Templates, …

07 **Memory control config keys** NEW 68

Four new config keys — `memory.persistence_enabled`, `memory.inject_memory`, `memory.inject_lessons`, and `memory.inject_activity` — let operators control memory writes and context-injection blocks independently.

— Names all four keys but not their file location or defaults.

Crew: Crewmate Teams, Agent Templates, …

08 **Hooks editor Kiro Agent event support** NEW 60

The Hooks editor now stores Kiro Agent `task`, `file`, and `manual` event names, runnable on demand via the Test action.

— Names the three event types and the Test action to run them.

Crew: Crewmate Teams, Agent Templates, …

THINNER COVERAGE BELOW

09 **Crewmate Teams** NEW 58

Crewmate Teams, available via the Crew Members preview, adds a `+ › New team` workflow to group the roster and surface crewmate status, pending questions, and recent activity.

— Names the UI entry point but no further configuration detail.

Crew: Crewmate Teams, Agent Templates, …

10 **Agent permission preservation during V3 upgrade** IMPROVED 55

Upgraded agent permissions now refresh to match V3 permissions whenever a V2 trust setting changes on an agent whose generated V3 fields are unchanged, preventing stale permission drift during the migration.

— Describes exact refresh condition but gives no command to invoke it.

CLI: Switch to V3 at Chat Start and Mod…

11 **Agent Templates tab** NEW 55

A new Agent Templates tab groups installed templates by origin and supports create, duplicate, edit, delete, one-off chat, and crewmate enrollment actions.

— Lists supported actions but no exact navigation path given.

Crew: Crewmate Teams, Agent Templates, …

12 **Model fallback notices in V3 transcripts** NEW 50

When a turn moves to a different model because of refusal or unavailability, Kiro CLI's V3 sessions now surface a fallback notice directly in the transcript so the switch is visible rather than silent.

— Explains trigger and effect but gives no control to act on.

CLI: Switch to V3 at Chat Start and Mod…

13 **DeepSeek Harness agent backend** NEW 50

Kiro Crew adds DeepSeek Harness as a selectable agent backend, available when `dsh` is installed on the Gateway host.

— Names the backend and its install prerequisite, no further detail.

Crew: Crewmate Teams, Agent Templates, …

14 **Subnet ID for EC2 crew launches** NEW 48

An optional Subnet ID field pins built-in EC2 crew launches to a specific subnet and VPC.

— Names the field but no defaults or further scope.

Crew: Crewmate Teams, Agent Templates, …

└──▷ **BREAKING ON UPGRADE**

! Crew logs are now on by default — each chat records a redacted event log; set `KIROCREW_CREW_LOG=0` before starting the Gateway to disable.

! Owner-written MCP `autoApprove` lists in `mcp.json` or agent files now take effect, bypassing the approval card and Kiro Crew's tool gate; set `mcp.honour_auto_approve` to `false` and restart the Gateway to keep those tools gated.

! Command cron validation is stricter: stored commands using brace expansion now fail until edited, imports skip those jobs, and quoted regex intervals such as `grep -E '[0-9]{1,3}'` are refused — use a script cron for shell features outside the command-cron subset.

WAS THIS USEFUL? 👍 👎

AGENT

◆ AI Agent Frameworks

Databricks Omnigent

Feedback ⓘ v0.18.0 NOTES ↗ RANK

Open-source meta-harness that orchestrates Claude Code, Codex, Cursor, Pi and custom agents behind one interface, with policy controls, OS-level sandboxing and live session sharing.

> Omnigent v0.18.0 extends PR review to GitLab and Azure DevOps, adds reusable custom agents installable via CLI, a Harnesses settings view, and an Inbox for tracking agent replies.

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

01 Reusable custom agents via CLI NEW

80

Adds `omnigent agent add` to install reusable custom agents, with `omnigent agent list` and `omnigent agent remove` to manage them; installed agents persist in the new-session picker across hosts.

List all installed custom agents and remove one that is no longer needed.

```
$ omnigent agent list
omnigent agent remove <agent-name>
```

— Names exact subcommands with runnable example v0.18.0

THINNER COVERAGE BELOW

02 PR review extended to GitLab and Azure DevOps IMPROVED

55

Extends PR/MR review in the session panel to GitLab merge requests and Azure DevOps pull requests alongside GitHub, including checks, comments, diffs, and automatic tracking of requests created or updated by agents.

— Describes scope and behaviour but no exact UI/command path v0.18.0

03 Harnesses view in Settings NEW

50

Adds a Settings → Harnesses view to browse each harness's MCP servers, skills, plugins, and startup settings.

Review MCP servers and plugins configured for a harness before starting a session.

📍 In the desktop app, go to Settings → Harnesses, select a harness, and inspect its MCP servers, skills, plugins, and startup settings.

— Exact UI navigation path given, mechanism thin v0.18.0

04 **Inbox and pinned session reordering** NEW 45

Adds an Inbox with All / Unread / Awaiting response tabs and expandable cards grouped by recency for catching up on unread agent replies; pinned sessions can now be drag-reordered in the sidebar.

— Names tabs and behaviour but no commands or keys v0.18.0

WAS THIS USEFUL? 👍 👎

Feynman

Feedback ⓘ v0.5.42 NOTES ↗ RANK

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

> Feynman expanded model provider support to 41 providers with per-account model refreshing, added flexible `--model` naming and exit-code signalling for scripted use, and unified telemetry opt-out with its underlying Pi agent.

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

01 Flexible model name formats for --model

IMPROVED — 90

`--model` now accepts bare model IDs (`claude-opus-5-5`), short names (`sonnet`), and a thinking-suffix form (`anthropic/claude-opus-5-5:high`), preferring a provider you are already signed in to.

Run a research workflow against a specific model using a short name or thinking-suffix variant without knowing the full provider path.

```
$ feynman --model anthropic/claude-opus-5-5:high deepresearch
"mechanistic interpretability"
```

— Exact flag syntax and a runnable example command v0.5.41

02 Provider login expanded to 41 providers with per-account model refresh

IMPROVED — 85

`feynman model login` now supports 41 model providers (up from 16), including DeepSeek, Fireworks, Together, NVIDIA, Google Vertex, Cloudflare, Qwen, and Bedrock with its profile and bearer-token options. Running `feynman model login` refreshes a provider's model list immediately after sign-in, so keys like OpenRouter show only the models the account can actually use, with real context sizes and prices.

After adding an OpenRouter API key, refresh the available model list so Feynman picks a default you can actually use.

```
$ feynman model login
```

— Names provider list, counts, and a runnable login command v0.5.42 v0.5.41

03 Unified telemetry opt-out with Pi

IMPROVED — 75

Setting `FEYNMAN_TELEMETRY=off` or `DO_NOT_TRACK=1` now also sets `PI_TELEMETRY=0`, disabling Pi's own install reporting alongside Feynman's own telemetry.

Opt out of all telemetry — both Feynman's PostHog reporting and Pi's install reporting — in one environment variable.

```
$ export FEYNMAN_TELEMETRY=off
feynman deepresearch "scaling laws"
```

— Named env vars with a runnable opt-out example v0.5.42

04 **Model set and default-model checks use Pi's settings API**

IMPROVED 65

`feynman model set` and the default-model check now use Pi's settings API, matching what Pi actually launches with, including trusted project settings.

— Mechanism explained but no example command shown v0.5.42

THINNER COVERAGE BELOW

05 **Wider browser detection for PDF export** IMPROVED 55

Preview and PDF export now detect Chromium, Edge, and Brave on Linux; Edge under Program Files (x86) and Brave in the user folder on Windows; and apps in `~/Applications` on macOS.

— Lists platforms and paths but is automatic, no command v0.5.41

06 **Session directory resolution from Pi settings** IMPROVED 50

Feynman now honors Pi's `sessionDir` setting and the `PI_CODING_AGENT_SESSION_DIR` environment variable when `--session-dir` is not passed.

— Names the setting, env var and flag, no example shown v0.5.42

07 **Exit-code signalling for truncated output in -p mode** NEW 45

`feynman -p` now warns and exits with code 1 when the model's output limit cuts off an answer.

— Names the flag and behaviour but no example v0.5.42

WAS THIS USEFUL?  

Stanford NLP DSPy

Feedback ⓘ 3.4.1 NOTES ↗ RANK

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

> DSPy 3.4.1 makes a structured, JSON-backed InstructionProposer the default for GEPA optimization and lets dspy.RLM's REPL spin up DSPy sub-agents under a shared call budget, alongside several breaking internal reorganizations and a refreshed LiteLLM model-metadata snapshot.

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

01 Structured InstructionProposer default for GEPA

BREAKING

95

`InstructionProposer` (backed by `dspy.Predict` + JSONAdapter) is now the default `instruction_proposer` in `dspy.GEPA`, replacing the old raw-prompt backtick-extraction approach. It gains `skills`, `additional_instructions`, `base_instructions`, `max_chars`, `truncate_history_outputs`, and `adapter` parameters for fine-grained control of reflection proposals, with a new `pyyaml>=6.0.1` dependency added to parse YAML frontmatter in `skills`. Because the reflection request is now a system message with structured input fields and JSON output, GEPA results can differ from 3.4.0 even with the same seed, `reflection_lm` must now be a `dspy.LM` or `dspy.BaseLM` (plain functions leave the program unoptimized), and `InstructionProposalSignature` must be imported from `gepa.strategies.instruction_proposal` instead of `dspy.teleprompt.gepa.gepa_utils`.

Configure GEPA with a customized InstructionProposer to inject domain-specific prompt-engineering skills and cap instruction length during optimization.

```python
import dspy
from dspy.teleprompt.gepa import InstructionProposer

optimizer = dspy.GEPA(
    metric=metric,
    reflection_lm=dspy.LM("openai/gpt-5"),
    instruction_proposer=InstructionProposer(
        skills=["./skills/prompt-engineering"],
        additional_instructions="Write instructions in imperative
voice.",
        max_chars=1500,
        truncate_history_outputs=True,
    ),
    auto="medium",
)
```

— Names every new param, dependency and migration path `3.4.1`

02 **Sub-agents inside dspy.RLM REPL** BREAKING 90

REPL code run by `dspy.RLM` can now `import dspy` and instantiate `dspy.Predict`, `dspy.ChainOfThought`, `dspy.ReActV2`, or a nested `dspy.RLM` as sub-agents, with all sub-agent LM calls counted against the single `max_llm_calls` budget. Every RLM prompt now includes a sub-agent section with no opt-out, so LM caches from 3.4.0 won't match; sub-agent calls reaching `max_llm_calls` sooner; an RLM input or tool named `dspy` / `_dspy` / `__dspy` now raises `ValueError`; and `dspy.ReActV2` signatures can no longer declare inputs named `history` or `tools`.

Use dspy.RLM with a sub-agent budget so REPL-generated code can delegate structured subtasks to `dspy.ChainOfThought` or `dspy.ReActV2`.

```python
import dspy

rlm = dspy.RLM(
    "logs, question -> answer",
    sub_lm=dspy.LM("openai/gpt-4o-mini"),
    max_llm_calls=30,
)
```

— Runnable example plus exact budget flag and new restrictions `3.4.1`

03 **Shared shim constants relocated to primitives.facade** BREAKING

70

Moves shared shim constants `SHIM_SETUP`, `CONSTRUCT_TOOL`, `CALL_TOOL`, `BRIDGEABLE_KINDS`, `SIGNATURE_MARKER`, `TOOL_MARKER`, and `prediction_to_fields` from `dspy.predict.flex.bridge` to `dspy.primitives.facade`, now shared by both `dspy.Flex` and `dspy.RLM`. Importing them from the old path now raises `ImportError`, and the private shim file moved from `dspy/predict/flex/_sandbox_shim.py` to `dspy/primitives/_facade_shim.py`, so `dspy.predict.flex._sandbox_shim` no longer imports.

— Exact old and new import paths given for migration `3.4.1`

---

04 **Score level descriptions can no longer be null** BREAKING

70

Programs saved on 3.4.0 with a null `Score` level description no longer load; program.load(path) now raises `ValueError`. Declaring `Score[None, ...]` or calling predictor.set_criteria(name, [None, ...]) raises `ValueError: Score level descriptions cannot be null.` Fix by replacing each `null` in the `criteria` list with a string or object, or deleting the `criteria` key.

— Gives exact error and concrete migration fix `3.4.1`

---

05 **Bundled LiteLLM model metadata refreshed** BREAKING

60

Updates the bundled LiteLLM model-metadata snapshot from 1.102.1 to 1.105.0, adding 803 entries and removing 264. Offline setups using the bundled snapshot lose tool-calling, structured-output, and reasoning capability flags for models removed in this update, e.g. `anthropic/claude-3-7-sonnet-20250219`, `gemini/gemini-2.0-flash`, `xai/grok-4`, and `groq/llama-3.3-70b-versatile`.

— Gives exact version numbers and named affected models `3.4.1`

THINNER COVERAGE BELOW

---

06 **dspy.Flex works with caller-chosen interpreters** IMPROVED

35

Extends `dspy.Flex` to work with in-process interpreters the caller chooses, without replacing the host's `dspy` module.

— States the change but no mechanism or example given `3.4.1`

---

07 **Python 3.15 support** NEW

10

Adds Python 3.15 support.

— Bare one-line mention with no further detail `3.4.1`

└──▷ BREAKING ON UPGRADE

- ! GEPA optimization results can differ from 3.4.0 even with the same seed: the default reflection request is now a system message with structured input fields and JSON output.
- ! The `reflection_lm` passed to `dspy.GEPA` must now be a `dspy.LM` or `dspy.BaseLM`; plain functions no longer work and cause the program to stay unoptimized.
- ! `dspy.teleprompt.gepa.gepa_utils` no longer re-exports `InstructionProposalSignature`; import it from `gepa.strategies.instruction_proposal` instead.
- ! Every RLM prompt now includes a sub-agent section with no opt-out flag; LM caches and recorded responses from 3.4.0 won't match.
- ! Sub-agent LM calls now count toward `max_llm_calls`; RLMs using sub-agents will reach the limit sooner.
- ! An RLM input named `dspy`, or a tool named `dspy` or starting with `_dspy` or `__dspy`, now raises `ValueError`.
- ! Importing `SHIM_SETUP`, `CONSTRUCT_TOOL`, `CALL_TOOL`, `BRIDGEABLE_KINDS`, `SIGNATURE_MARKER`, `TOOL_MARKER`, or `prediction_to_fields` from `dspy.predict.flex.bridge` now raises `ImportError`; import from `dspy.primitives.facade` instead.
- ! The private shim file moved from `dspy/predict/flex/_sandbox_shim.py` to `dspy/primitives/_facade_shim.py`; `dspy.predict.flex._sandbox_shim` no longer imports.
- ! `dspy.ReActV2` signatures can no longer declare inputs named `history` or `tools`.
- ! Programs saved on 3.4.0 with a null `Score` level description no longer load; program.load(path) raises `ValueError`. Replace each `null` in the `criteria` list with a string or object, or delete the `criteria` key.
- ! Declaring `Score[None, ...]` or calling predictor.set_criteria(name, [None, ...]) now raises `ValueError: Score level descriptions cannot be null.`
- ! Offline setups using the bundled model-metadata snapshot lose tool-calling, structured-output, and reasoning capability flags for models removed in LiteLLM 1.105.0 (e.g. `anthropic/claude-3-7-sonnet-20250219`, `gemini/gemini-2.0-flash`, `xai/grok-4`, `groq/llama-3.3-70b-versatile`).

WAS THIS USEFUL? 👍 👎

◆ MCP Servers & Tooling

gridctl

Feedback ⓘ v1.0.0-rc.2 NOTES ↗ RANK

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

> gridctl's v1.0.0-rc.2 release ships direct tool discovery and invocation from the CLI, offline stack policy validation, a unified import pipeline for servers/skills/agents/context, a hardened container execution mode, and experimental A2A tool support, alongside a breaking change making gateway security settings restart-only.

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

01 Direct tool search and invocation via gridctl call NEW

90

`gridctl tools search <query>` searches tools advertised by running MCP servers by substring match without starting a daemon, and `gridctl call <tool> '<json-args>'` invokes a tool directly through the gateway, e.g. `gridctl call fetch__fetch '{"url":"https://example.com"}'`.

Discover which tools a running gateway exposes, then call one directly — no client wiring needed.

```
$ gridctl tools search fetch
gridctl call fetch__fetch '{"url":"https://example.com"}'
```

Search for and invoke a tool on the running gateway directly from the CLI — no agent client required.

```
$ gridctl tools search fetch
gridctl call fetch__fetch '{"url":"https://example.com"}'
```

— Exact commands and worked example given v1.0.0-rc.2

02 Offline stack declaration policy validation NEW

90

`gridctl validate stack.yaml --policy policy.yaml` performs offline declaration-policy checks against a stack file; policies can be opt-in enforced via `validate --policy` with CI workflow support, catching disallowed server configurations before apply, e.g. `gridctl validate --policy stack-declaration-policy/policy.yaml stack.yaml`.

Enforce offline stack declaration policy in CI so disallowed server configurations are caught before apply.

```
$ gridctl validate --policy stack-declaration-policy/policy.yaml
stack.yaml
```

— Named command, flag, and CI usage example `v1.0.0-rc.2`

---

03 **Unified client import for servers, skills, agents, and context**

NEW 90

`gridctl import --kind all` imports a client's servers, skills, agents, and global context in a single run with one summary; `--kind agents` imports Claude Code subagent definitions (`agents/*.md`) from a git repo, local directory, or client home via `gridctl import` and `gridctl skill import` ; `gridctl skill import <client-home>` imports skills already present in a client home directory; and `gridctl skill project sync --kind agent` places imported agents onto separate targets — an identity copy for Claude Code, and rendered dialects for OpenCode, Copilot, and Gemini CLI. Local skill and agent import from directories and client skill locations is also supported.

Import all existing client config — servers, skills, agents, and global context rules — into gridctl in one pass.

```
$ gridctl import opencode --kind all
```

— All flags and commands named with an example `v1.0.0-rc.2`

---

04 **Hardened execution mode for managed containers** NEW

90

The `execution.mode: hardened` config key in the stack file enforces explicit numeric UID/GID, finite per-replica limits, read-only root filesystem, bounded scratch space, and no network access by default for managed MCP containers.

Enforce hardened container execution for MCP servers — read-only root, no network, bounded resources — by opting in via the stack file.

```yaml
execution:
  mode: hardened
```

— Exact config key and enumerated enforced constraints `product docs`

05 **Passive security evidence via gridctl doctor --security** NEW

70

`gridctl doctor --security` surfaces passive security evidence reports, including partial, stale, suppressed, unknown, and N/A states.

— Named command and enumerated states, no worked example v1.0.0-rc.2

06 **Experimental outbound A2A tool support** NEW

60

Adds experimental outbound A2A tool support with card trust and secret task/context handles (see `examples/a2a/`), along with A2A wire codecs, capability-guarded A2A dispatch, and capability/observation primitives for an outbound Agent Card adapter.

— Mechanism named but no runnable command given v1.0.0-rc.2

07 **Resource and prompt aggregation in gridctl status** NEW

60

Downstream resource and prompt aggregation surfaces new `PROMPTS` and `RESOURCES` columns in `gridctl status` output.

— Names exact columns but no example shown v1.0.0-rc.2

08 **Restart-only gateway security settings** BREAKING

60

Gateway authentication and listener security settings are now restart-only: `reload` rejects security changes, including mixed edits, and requires a full restart; see the restart recovery docs.

— Clear behavior change but no migration command shown product docs

THINNER COVERAGE BELOW

09 **Pack manifests with pinned repos and stack bootstrap** NEW

45

Pack manifest entries can reference pinned external repositories, and pack-carried stack bootstrap can initialise a stack from a pack.

— Described in prose without a concrete command v1.0.0-rc.2

10 **Self-contained config files for container servers** NEW

40

Inline or stack-relative config files can now be copied into MCP container servers as self-contained config files.

— Thin description with no example or key name v1.0.0-rc.2

11 **tools/list_changed notification support** NEW

30

Adds support for `tools/list_changed` notifications from MCP servers.

— Bare name with minimal elaboration `v1.0.0-rc.2`

└──▷ **BREAKING ON UPGRADE**

! Gateway authentication and listener security settings are restart-only: `reload` rejects security changes (including mixed edits) and requires a full restart; see restart recovery docs.

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ Inference Providers & GPU Clouds

Together AI

Feedback ⓘ snapshot-20261013 NOTES ↗ RANK

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

> Together AI launched a full post-training API, adding low-level training-session, GPU-resource, and checkpoint control planes alongside GRPO reinforcement learning, DPO, SFT, multi-LoRA deployment, and FP4 quantisation pipelines.

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

01 **Training session lifecycle API** NEW 95

Adds `POST /training/sessions` and `GET /training/sessions/{id}` to create and retrieve training sessions, `PATCH /training/sessions/{id}` to update a session's display name or metadata (omitted fields unchanged; empty strings clear existing values), `POST /training/sessions/{id}/forward-backward` and `POST /training/sessions/{id}/custom-forward-backward` for forward-backward passes with externally-computed gradients against per-token log-probabilities, `POST /training/sessions/{id}/optim-step` to asynchronously apply accumulated gradients, `POST /training/sessions/{id}/weights-sync` to make updated parameters available for sampling after an optimizer step, and `POST /training/sessions/{id}/sample` to asynchronously generate text completions with log-probabilities from the current policy.

— Full endpoint set with exact methods, paths and behaviour named. snapshot-20261013

02 **FP4 quantisation and shaping job pipeline** NEW 95

Adds `POST /shaping/prepare-for-fp4-inference` to merge a fine-tuned LoRA adapter into its base model and quantise the result to FP4 for dedicated inference, with `POST /shaping/prepare-for-fp4-inference/estimate` to price the job before submission, plus `GET /shaping/jobs`, `GET /shaping/jobs/{id}`, `POST /shaping/jobs/{id}/cancel`, and `GET /shaping/jobs/{id}/events` to manage and monitor FP4 preparation jobs.

After training, merge a LoRA adapter into its base model and quantise to FP4 for low-latency dedicated inference — price it first to avoid surprises.

```
$ # Price the FP4 prep job
curl -X POST https://api.together.ai/v1/shaping/prepare-for-fp4-
inference/estimate \
  -H 'Authorization: Bearer $TOGETHER_API_KEY' \
  -H 'Content-Type: application/json' \
  -d '{"adapter_model": "my-project/my-lora-adapter"}'

# Submit the merge + quantise job
curl -X POST https://api.together.ai/v1/shaping/prepare-for-fp4-
inference \
  -H 'Authorization: Bearer $TOGETHER_API_KEY' \
  -H 'Content-Type: application/json' \
  -d '{"adapter_model": "my-project/my-lora-adapter"}'
```

— Named endpoints plus a runnable curl example for estimation and submission.

snapshot-20261013

03 **GPU training resource provisioning** NEW 90

Adds `POST /training/resources` to provision a standalone GPU model resource, `GET /training/resources/{id}` and `POST /training/resources/{id}/stop` (with `force=true` to override active sessions) to manage its lifecycle, and `POST /training/resources/estimate-cost` to estimate on-demand hourly GPU resource price without provisioning.

— Exact endpoints, flag and lifecycle behaviour given. snapshot-20261013

04 **Training and inference checkpoint management** NEW 90

Adds `POST /training/sessions/{id}/checkpoints/inference` to save the current LoRA adapter as a downloadable inference checkpoint, `POST /training/sessions/{id}/checkpoints/training` to save the full training state (adapter + optimizer + step), `GET /training/checkpoints` to list checkpoints filterable by type, session, or base model, and `GET /training/checkpoints/{id}/download` to retrieve presigned download URLs for inference checkpoints.

— All four checkpoint endpoints named with their distinct purposes. snapshot-20261013

05 **GRPO reinforcement learning training** NEW 78

Adds `GET /rl/models` to list models supported by the RL service with their training and sampling limits, and supports reinforcement learning with GRPO — training a model against a custom reward function using group relative policy optimization, including multi-turn tool-use agent training over whole episodes.

— One named endpoint plus described mechanism, but no full workflow shown.

snapshot-20261013

THINNER COVERAGE BELOW

06 **DPO and SFT via post-training API** NEW 50

Supports direct preference optimization (DPO) post-training on preference pairs, including custom client-side loss functions computed from the model's log-probabilities, and supervised fine-tuning (SFT) via the post-training API, distinct from the existing fine-tuning API, allowing adapter and full-weight training modes.

— Describes capability and modes but names no specific endpoint. snapshot-20261013

07 **Multi-LoRA deployment with per-request adapter selection** NEW 50

Supports Multi-LoRA deployment: attach adapters from post-training inference checkpoints to one dedicated base-model deployment and select an adapter per request.

— Describes mechanism but gives no deployment API or flag name. snapshot-20261013

08 **Multi-adapter post-training on shared resources** NEW 40

Supports multi-adapter post-training: share one provisioned model resource across several adapter sessions to run hyperparameter sweeps at the cost of a single base model.

— Explains mechanism and benefit but no concrete surface named. snapshot-20261013

09 **Weights & Biases metric streaming** NEW 30

Enables Weights & Biases metric streaming for post-training sessions to monitor run health alongside console-based tracking.

— Bare description with no configuration detail given. snapshot-20261013

10 **Bring Your Own Model fine-tuning** NEW 30

Supports Bring Your Own Model fine-tuning for Hugging Face models not in the Together catalog.

— One-line description with no setup steps or limits given. snapshot-20261013

WAS THIS USEFUL? 👍 👎

Eigen Labs Darkbloom

Feedback ⓘ v0.9.20 NOTES ↗

CODE ↗ RANK

Darkbloom is a for the innovators pushing the frontier.

> Darkbloom v0.9.20 adds a dedicated CLI for configuring encrypted cache storage and write limits, raises the default SSD write budget to 4 TB, and introduces socket-bound endpoint resumption, isolated App Attest, and SSD-backed quantized KV caching.

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

01 **Cache storage and write-budget configuration** NEW 95

New `darkbloom cache set --daily-write-gb` and `--directory` flags configure encrypted cache storage location and per-day write limits, saved under `[cache]` in `provider.toml`; `--reset-directory` reverts to built-in storage while preserving the write limit. A new `darkbloom cache status` subcommand inspects the current cache configuration and write-limit source. The default SSD cache write budget was also raised from 750 GB to 4 TB (4,000 decimal GB) per rolling day, with saved limits and environment overrides still taking precedence.

Point the provider cache at a dedicated SSD and cap daily writes — useful when the default volume is small or shared.

```
$ darkbloom cache set --daily-write-gb 100 --directory
"/Volumes/Cache SSD/provider-cache"
darkbloom cache status
darkbloom restart
```

— Names exact flags, config file, subcommand, and the 750GB→4TB change v0.9.20

THINNER COVERAGE BELOW

02 **Native KV with quantized SSD checkpoints** NEW 40

Serves native KV with quantized SSD checkpoints, enabling larger model key-value caches backed by local SSD storage.

— States capability but lacks config or command detail v0.9.20

03 **Socket-bound endpoint resumption** NEW 35

Adds socket-bound endpoint resumption to the provider, allowing in-flight connections to resume across transient network interruptions.

— Describes mechanism but no flags or commands given v0.9.20

04 **App Attest isolated in supervised worker** IMPROVED 35

Isolates App Attest in a supervised worker, keeping attestation responsive and fault-contained during model waits.

— Explains behavior change but no actionable surface named v0.9.20

05 **Build-time default coordinator and CDN selection** NEW 25

Allows the default coordinator and CDN to be chosen at build time.

— Single-sentence mention with no build flags or values named v0.9.20

WAS THIS USEFUL? 👍 👎

Baseten

Feedback ⓘ Egress restrictions NOTES ↗ RANK

Baseten deploys and serves machine learning models as autoscaling inference endpoints, with a model API catalog and dedicated GPU deployments.

> Baseten added network egress restrictions, letting deployments restrict outbound traffic to an allow list of IP ranges and domains via Truss config.

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

01 **Egress restrictions in Truss config** NEW 81

Adds `egress_restrictions` under `runtime` in the Truss config to restrict a deployment's outbound network traffic to an allow list of IP ranges and domains, e.g. specifying CIDR ranges like `203.0.113.0/24` and domains like `example.com`.

Lock down a sensitive model deployment so it can only reach specific external IPs and domains at runtime.

yaml

```yaml
runtime:
  egress_restrictions:
    - 203.0.113.0/24
    - example.com
```

— Names exact config key and shows runnable config example. Egress restrictions

WAS THIS USEFUL? 👍 👎

◆ Agentic Infrastructure

HeyGen HyperFrames

Feedback ⓘ v0.8.152 NOTES ↗ RANK

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

> HyperFrames shipped flags to guard shared project history turns, moved artifact and render output storage to per-revision paths with richer history tracking, and introduced a live Inspector subsystem alongside a CLI tween-order warning.

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

01 **Turn guard flags for shared project history** NEW 90

`--exclusive`, `--who`, and `--if-open` flags guard history turns so agents sharing a project can refuse to start while another turn is open, end only their own turn, or end safely when no turn is open.

— Names all three flags and their exact guard behavior v0.8.152

02 **Per-revision artifact storage and history tracking** BREAKING 90

Check reports now save to `.hyperframes/artifacts/checks/<id>.json`, snapshots to `.hyperframes/artifacts/snapshots/<id>/`, and render outputs to `.hyperframes/render-outputs.json`, scoping all artifacts per revision; video state, edit targets, and artifact cleanup are recorded in `.hyperframes/video-state.json` after every edit. The default snapshot path moved from `snapshots/` to the new per-revision location, so anything reading the old path or old check/render locations will find nothing after upgrading.

— Lists every new path and the breaking move from old locations v0.8.152

03 **Live Inspector subsystem** NEW 65

Adds an Inspector element map, a live inspection engine, and a Node runtime for real-time composition analysis, with the ability to query timelines and inspect the live preview via the Inspector, plus a new component inspector package added to the Inspector subsystem.

— Names subsystem components but gives no entry command v0.8.151

THINNER COVERAGE BELOW

04 **CLI tween-order warning** NEW 45

The CLI now warns when a tween ends just after a later tween on the same property.

— Describes trigger condition but no flag or output format v0.8.151

└──▷ **BREAKING ON UPGRADE**

! Default snapshot path moved from `snapshots/` to `.hyperframes/artifacts/snapshots/<id>/` ; anything reading `snapshots/` will find nothing after upgrading.

! Check reports are now written to `.hyperframes/artifacts/checks/<id>.json` instead of their previous location.

! Renders now write `.hyperframes/render-outputs.json` ; any tooling expecting output at the old path will break.

WAS THIS USEFUL?  

Google Gen AI JavaScript SDK

Feedback ⓘ v2.29.0

NOTES ↗ RANK

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

> Google Gen AI JavaScript SDK v2.29.0 adds resumable Interactions via continuation tokens, precise transcript timing for live audio sessions, and expanded token-usage accounting.

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

01 Interim transcript timestamps in live sessions NEW

80

Adds `interimTranscriptTimestampEnabled` field to `RealtimeInputConfig` and `startOffset` / `endOffset` fields to `Transcription`, giving precise timing for each transcription segment during real-time audio sessions.

Enable interim transcript timestamps to get `startOffset` / `endOffset` on each transcription segment during real-time audio sessions.

typescript

```
const session = await ai.live.connect({
  model: 'gemini-flash-latest',
  config: {
    realtimeInputConfig: {
      interimTranscriptTimestampEnabled: true,
    },
  },
});
// Each Transcription object now includes startOffset and
endOffset
```

— Named fields plus a runnable code example enabling the feature v2.29.0

02 Resumable interactions via continuation tokens NEW

75

Adds a `continuation_token` field to the `interaction.status_update` Interactions event, letting clients resume long-running interactions. A new `CONTINUATION_REQUIRED` status in the Interactions status enum signals when this token must be consumed, and the SDK automatically consumes it on mid-stream errors or cutoffs, resuming generation from the latest checkpoint.

— Named field and enum, mechanism explained, but no runnable example v2.29.0

THINNER COVERAGE BELOW

03 **Expanded usage metadata and token accounting** NEW 50

Adds billable token count fields to usage metadata responses, an `initialPromptTokenCount` field to streaming continuation usage metadata that accumulates across continuation segments, and usage reporting within `InteractionStatusUpdate` events.

— Names fields but no mechanism or example of use v2.29.0

WAS THIS USEFUL? 👍 👎

Google Gen AI Python SDK

Feedback ⓘ v2.30.0 NOTES ↗

RANK

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

> v2.30.0 adds timestamped interim transcription for live sessions and introduces automatic continuation-token resumption for streaming and chat, alongside new billing and usage metadata fields.

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

01 Timestamped interim transcripts in live sessions NEW

80

Adds `interimTranscriptTimestampEnabled` to `RealtimeInputConfig` and `startOffset` / `endOffset` fields to `Transcription`, enabling timestamped interim transcripts to correlate speech segments with audio offsets during real-time sessions.

Enable timestamped interim transcripts in a real-time session to correlate speech segments with audio offsets.

python

```python
from google import genai
from google.genai import types

client = genai.Client(api_key='GEMINI_API_KEY')

config = types.LiveConnectConfig(
    realtime_input_config=types.RealtimeInputConfig(
        interimTranscriptTimestampEnabled=True
    )
)

# Each Transcription object in the stream will carry startOffset
and endOffset
async with client.aio.live.connect(model='gemini-flash-latest',
config=config) as session:
    async for message in session.receive():
        if message.server_content and
message.server_content.input_transcription:
            t = message.server_content.input_transcription
            print(t.startOffset, t.endOffset, t.text)
```

— Named config field plus a runnable code example. v2.30.0

02 **Automatic continuation-token resumption for streaming and chat**

NEW

75

Adds `continuation_token` to the `interaction.status_update` event and a new `CONTINUATION_REQUIRED` status to the Interactions API, plus `initial_prompt_token_count` in streaming continuation usage metadata that accumulates across continuation chunks. Enables automatic continuation-token resumption for `generate_content` and `generate_content_stream` (disabled by default) and for the `chats` module.

— Names API fields and methods but lacks a usage example. v2.30.0

THINNER COVERAGE BELOW

03 **Usage metadata additions for billing** NEW

30

Adds billable token count fields to usage metadata and a `usage` field to `InteractionStatusUpdate` events.

— Field names unspecified beyond generic description. v2.30.0

WAS THIS USEFUL?  

AWS Amazon Bedrock AgentCore

Feedback ⓘ

Memory: Time-to-live for long-term memory records (2026-10-12) BLOG ↗ RANK

Amazon Bedrock AgentCore provides managed agent runtimes, memory, identity, gateways, observability, and evaluation services.

> Amazon Bedrock AgentCore Memory now supports time-to-live expiry for long-term memory records, letting stale data be purged automatically.

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

01 **TTL for long-term memory records** NEW 90

AgentCore Memory adds `recordExpiryDurationDays` (minimum 31 days) to `CreateMemory` and `UpdateMemory` for strategy-level TTL, and to `BatchCreateMemoryRecords` and `BatchUpdateMemoryRecords` for per-request TTL overrides on long-term memory records. A new `bedrock-agentcore:UpdateMemoryRecordExpirySpecification` IAM permission is required to configure or remove a record's TTL.

Set a 90-day expiry on a memory strategy at creation time so stale long-term records are automatically purged.

```
$ aws bedrock-agentcore create-memory \
  --name 'customer-context' \
  --strategies '[{"strategyId": "s1", "recordExpiryDurationDays":
90}]'
```

Override TTL per-request when bulk-ingesting records that should expire sooner than the strategy default.

```
$ aws bedrock-agentcore batch-create-memory-records \
  --memory-id <memory-id> \
  --records '[{"content": "...", "recordExpiryDurationDays":
31}]'
```

— Names exact fields, minimum value, IAM permission, and runnable CLI commands.

Memory: Time-to-live for long-term memo…

WAS THIS USEFUL? 👍 👎

◆ Inference Serving & Optimization

OpenNMT CTranslate2

Feedback ⓘ v4.8.3 NOTES ↗ RANK

CTranslate2 runs Transformer models with a C++ inference engine, quantization, and CPU and GPU optimizations.

> CTranslate2 v4.8.3 adds support for converting UMT5 models from Hugging Face Transformers for optimized inference.

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

01 **UMT5 model conversion from Hugging Face** NEW 35

CTranslate2 now supports converting UMT5 models from Hugging Face Transformers for optimized inference.

— Names UMT5 and source but no commands or mechanism given v4.8.3

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

MLX Serve

Feedback ⓘ v26.10.2 NOTES ↗ RANK

The mlx-serve runtime provides Apple Silicon inference with OpenAI- and Anthropic-compatible APIs, chat, agent mode, and tool calling.

> mlx-serve v26.10.2 extends Apple Silicon inference to Linux/NVIDIA hardware and ships a wave of media-generation capabilities — video storyboards, full-song composition, sound effects and fast image generation — alongside new frontier language models like DeepSeek-V4.1-Flash, a new Code Launcher menu, and a precision-control environment variable.

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

01 bf16 precision env var for Qwen3.8 Flash Next IMPROVED

88

Adds `MLX_SERVE_GDN_STATE_F32=0` environment variable to restore bf16 precision for the Qwen3.8 Flash Next linear-attention state; the default is f32, and switching to bf16 halves that state's memory footprint.

Switch Qwen3.8 Flash Next linear-attention state to bf16 to halve its memory footprint on memory-constrained Macs.

```
$ MLX_SERVE_GDN_STATE_F32=0 mlx-serve
```

— Exact env var and command given, with before/after memory behaviour. v26.10.2

02 Homebrew cask renamed to mlx-serve BREAKING

75

The Homebrew cask is renamed to `mlx-serve`; run `brew trust ddalcu/mlx-serve` once to adopt the new cask.

— Exact migration command given for a breaking change. v26.10.2

03 New app launchers in mlx-serve launch NEW

70

Adds `mlx-serve launch` with new launchers: ZCode, fx, and Grok, joining the existing Code menu. The Code Launcher panel in MLX Serve Web lets you select a target such as OpenCode to copy a generated setup command to paste into a terminal on the remote machine.

Launch OpenCode pointed at your local mlx-serve instance from the built-in Code Launcher.

📍 In MLX Serve Web, open the Code Launcher panel and select `OpenCode` to copy the generated setup command, then paste it into a terminal on the remote machine.

— Names launchers and gives exact UI navigation steps. v26.10.2

04 **Linux support for NVIDIA DGX Spark and RTX cards** NEW 63

Extends Linux support to NVIDIA DGX Spark and RTX 20–50 series cards, with ready-made x64 and arm64 downloads per release.

— Names hardware and architectures but no setup command. v26.10.2

05 **Qwen-Image-2.1-Turbo image generation** NEW 63

Adds Qwen-Image-2.1-Turbo image generation, producing 1024×1024 images in approximately 8 seconds on M5 Ultra.

— Names model, resolution, speed and hardware benchmark. v26.10.2

THINNER COVERAGE BELOW

06 **New frontier model support added** NEW 58

Adds support for DeepSeek-V4.1-Flash (with SSD expert streaming via `mlx-stream`), GLM-5.3-Flash, MiMo-V2.6-Flash, Kolibri-1, and EmbeddingGemma 2 (text, image, and video embeddings).

— Names five models and one mechanism, no usage steps. v26.10.2

07 **Video storyboard generation with Enhance** NEW 55

Adds video storyboard generation: describe a story (or attach a first frame) and Enhance writes shots joined into a single clip up to two minutes long.

— Describes mechanism and length limit but no UI path given. v26.10.2

08 **YuE2 full-song composition with vocals** NEW 55

Adds YuE2 for full-song composition with vocals generated from lyrics and a style, including score display for melody editing.

— Names model and capability but no usage steps. v26.10.2

09 **Media model downloads via mlx-serve pull** IMPROVED 45

Adds `mlx-serve pull` support for downloading media models in addition to language models.

— Names the command but no flags or model list. v26.10.2

10 **AI Radio continuous music composition** NEW 45

Adds AI Radio for continuous theme-based music composition via MLX-Amp, a spectrum-visualising player.

— Names feature and player, no usage path given. v26.10.2

11 **Stable Audio 3 sound-effect generation** NEW 40

Adds Stable Audio 3 for sound-effect generation in approximately one second.

— Names model and speed, no usage detail. v26.10.2

└──▷ **ALSO FROM THESE RELEASES**

.
.
.

└──▷ **BREAKING ON UPGRADE**

! The Homebrew cask is renamed to `mlx-serve`; run `brew trust ddalcu/mlx-serve` once to adopt the new cask.

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

Pinecone

Feedback ⓘ snapshot-20261013 NOTES ↗ RANK

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

> Pinecone's biggest release in this window adds dedicated read nodes with configurable scaling and recall/latency tuning, a full admin REST API for managing orgs, projects, users, service accounts and role bindings, and three new cloud storage integrations for bulk import.

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

01 Dedicated read nodes indexes NEW

95

Indexes can now be created with provisioned read hardware — from scratch, from a backup, or by migrating an existing on-demand or pod-based index. Scale controls add replicas for query throughput and shards for storage capacity, and new `scan_factor` and `max_candidates` query parameters let you trade recall for lower latency and higher throughput.

Tune a dedicated read nodes index to lower latency at the cost of some recall by adjusting scan_factor and max_candidates at query time.

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

— Names exact query params and shows a runnable tuning example snapshot-20261013

02 Admin REST API for orgs, projects, users and service accounts

NEW

95

A new admin REST API covers full project lifecycle (list all projects, create a project, get project details, update a project's name/Pod limit/CMEK, delete a project), organization management (list organizations, get details, update name, delete an organization), API key management (list, create, get details, update name and roles, delete), user management (list users filterable by email, get user details, remove a user, and list/create/get/delete/resend invites), service accounts (list, create with optional role bindings, get details, update name, delete, and rotate OAuth client secret via `POST /organizations/<org-id>/service-accounts/<sa-id>/rotate-secret`), and

role bindings (list filterable by principal, resource, role; create via `POST /organizations/<org-id>/role-bindings`; get details; delete).

Rotate a service account OAuth client secret via the admin API — the previous secret and its tokens are revoked within seconds.

```
$ curl -X POST 'https://api.pinecone.io/organizations/<org-id>/service-accounts/<sa-id>/rotate-secret' \
  -H 'Api-Key: <PINECONE_API_KEY>'
```

Grant a role to a user or service account at project scope using the new role-binding endpoint.

```
$ curl -X POST 'https://api.pinecone.io/organizations/<org-id>/role-bindings' \
  -H 'Api-Key: <PINECONE_API_KEY>' \
  -H 'Content-Type: application/json' \
  -d '{"principal": "user:<user-id>", "resource": "project:<project-id>", "role": "editor"}'
```

— Enumerates every endpoint area with two runnable curl examples `snapshot-20261013`

03 **Cloud storage integrations for bulk import** NEW 60

Adds storage integrations with Amazon S3 (via IAM role, used for bulk import and audit log export), Google Cloud Storage (via service account key, used for bulk import), and Azure Blob Storage (via service principal, used for bulk import).

— Names all three providers and auth mechanisms but no setup command `snapshot-20261013`

THINNER COVERAGE BELOW

04 **Fetch data and List IDs record retrieval** NEW 55

A Fetch data capability retrieves documents or records from a namespace by ID or metadata filter, and a List IDs capability lists and paginates record IDs in a serverless index namespace with ID-prefix filtering.

— Names both capabilities and filtering options, no example given `snapshot-20261013`

05 **Search-quality diagnostic tool** NEW 40

A new diagnostic tool compares exact brute-force search results against index results to isolate whether issues stem from embeddings or from recall.

— Explains mechanism but no interface or command named `snapshot-20261013`

06 **Local Pinecone API mock for testing** NEW 35

Adds a local mock of the Pinecone API runnable from its OpenAPI specs, letting tests and CI run without a live account.

— Describes purpose but no setup command or flag shown `snapshot-20261013`

07 **Common error messages reference** NEW 30

Adds a Common error messages reference documenting Pinecone API and SDK errors along with their causes and fixes.

— Named doc resource but no content specifics given `snapshot-20261013`

08 **Monthly spend alerts** NEW 20

Adds monthly spend alerts to catch unexpected charges.

— Bare feature name with no configuration detail `snapshot-20261013`

WAS THIS USEFUL? 👍 👎

Volcengine OpenViking

Feedback ⓘ `python-sdk@0.1.15`

NOTES ↗ RANK

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

> OpenViking's Python SDK 0.1.15 adds fine-grained controls to batch writes and search, supporting conflict handling in VikingBot Compile workflows and depth-bounded grep searches.

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

01 **skip_conflicts option for batch writes** NEW 55

Adds `BatchWriteOptions.skip_conflicts` option to batch write operations, required for VikingBot Compile finalize workflows.

— Names the exact flag but not full mechanism `python-sdk@0.1.15`

02 **Depth-bounded grep search** NEW 52

Adds `level_limit` parameter to the grep() function for depth-bounded search.

— Names function and parameter, lacks behavior detail `python-sdk@0.1.15`

WAS THIS USEFUL? 👍 👎

GrowGraph OntoCast

Feedback ⓘ v0.6.8 CODE ↗ RANK

OntoCast generates RDF knowledge graphs from documents by co-evolving domain ontologies and fact graphs with an agentic LLM pipeline.

> OntoCast v0.6.8 introduces live token-budget enforcement across the pipeline, adding a run-level meter with hard caps and a dedicated error, per-document token limits on the processing endpoints, and separate accounting for cache-replayed tokens.

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

01 RunMeter live token tracking with enforceable caps NEW

95

Adds `RunMeter`, bound via `use_run_meter`, which counts provider and cache-replayed tokens per model across a run while it is in progress. It supports optional `token_caps` and a caller's `limit` policy checked before each provider call, plus a throttled `on_usage` callback; once a cap is reached it raises `BudgetExceededError` before the provider call, stopping the whole run after parallel units finish and carrying the meter's `snapshot`.

Enforce a token budget across a run so a long document batch stops before an expensive model call overshoots your limit.

python

```python
from ontocast.onto.run_meter import RunMeter, use_run_meter,
BudgetExceededError

meter = RunMeter(token_caps={"gpt-4o": 50000})
with use_run_meter(meter):
    try:
        result = await graph.ainvoke(state)
    except BudgetExceededError as exc:
        print(exc.snapshot)
```

— Full mechanism plus a runnable code example. v0.6.8

02 Per-document token caps on processing endpoints NEW

85

Adds `MAX_INPUT_TOKENS_PER_DOCUMENT` and `MAX_OUTPUT_TOKENS_PER_DOCUMENT` per-document token caps enforced on `ontocast process`, `/process`, `/process_unit`, and `make_ontocast_node`. A capped route answers `422` with `error_code` `budget_exceeded`, and a capped batch file is recorded as failed.

— Names exact config keys, endpoints and error response. v0.6.8

THINNER COVERAGE BELOW

03 **Cache-replay token accounting fields** NEW 55

Adds budget fields `replayed_reasoning_tokens`, `replayed_cache_read_input_tokens`, and `replayed_cache_creation_input_tokens` to track tokens replayed from disk cache separately from provider-call totals.

— Named fields but no usage path given. v0.6.8

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

LangChain LangSmith

Feedback ⓘ snapshot-20261013 NOTES ↗

RANK

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

> LangSmith's September 7–14 2026 release tightens trace retention and renames core permission scopes as breaking changes, while adding a model-pricing MCP tool, SSO redirect handling, finer-grained model access controls, and several operational and admin-visibility improvements.

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

01 **redirect_to parameter for SSO login** NEW 90

Adds `redirect_to` query parameter to the organization SSO login URL (`/sso/login/<slug>?redirect_to=<path>`) so SAML sign-in lands on the intended page instead of the workspace root.

Redirect users to a specific trace after SAML sign-in rather than the workspace root.

```
$ /sso/login/<slug>?
redirect_to=/o/<org>/projects/p/<project>/r/<run>
```

— Exact URL pattern plus runnable example snapshot-20261013

02 **Model access policy filtering on LLM Gateway** IMPROVED 85

`GET /v1/models` on the LLM Gateway now returns only the models permitted by the calling credential's model access policies.

Audit which models the LLM Gateway will actually serve to a credential — filtered by its model access policies.

```
$ curl -s https://<gateway-host>/v1/models \
  -H 'Authorization: Bearer <token>'
```

— Named endpoint with runnable curl example snapshot-20261013

03 **Workspace role permissions renamed** BREAKING 75

Built-in workspace roles no longer return the `repos:*` and `run-rules:*` permissions; they have been renamed to `prompts:*` and `rules:*`, and using the old names when creating a custom role is rejected.

— Names exact old and new permission scopes, no migration script `snapshot-20261013`

04 **Trace retention capped at 180 days** BREAKING 65

Extended trace retention on LangSmith SaaS is now capped at 180 days: existing settings above 180 days are silently applied as 180 days for new traces, and attempts to set a value above 180 days are rejected.

— Clear before/after and limit, no migration command given `snapshot-20261013`

05 **Workspace visibility into Agent Auth connections** NEW 65

Workspace administrators can view workspace-level Agent Auth connections from the Integrations settings page, including authentication type and update details, and can inspect an Agent Auth connection's associated credential metadata without exposing secret or token material.

— Gives exact UI location but no API or command `snapshot-20261013`

06 **Model configurations provider in Model Access policies** NEW 60

Model Access policies gain a Model configurations provider, letting a workspace explicitly permit all or a chosen subset of its saved model configurations.

— Describes mechanism but no config key or UI path given `snapshot-20261013`

THINNER COVERAGE BELOW

07 **Raw TCP access to allow-listed public IPs in sandboxes** NEW 55

Sandboxes now pass raw TCP connections to literal public IP addresses when the destination IP and port are explicitly allow-listed.

— Explains behaviour but no config surface for the allow-list `snapshot-20261013`

08 **Model pricing lookup via MCP tool** NEW 50

LangSmith MCP adds `list_model_price_maps`, a tool for searching and paginating model pricing rules configured in a workspace.

— Named tool but no invocation example shown `snapshot-20261013`

09 **IAM role assumption for ECR snapshot pulls** NEW 50

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

— States capability without setup steps or tag format `snapshot-20261013`

---

10 **Context-aware naming in LangSmith Chat** IMPROVED 50

LangSmith Chat now reads active filters, view, time window, and selection from the tracing project screen and names runs and threads by what was asked.

— Describes behaviour change, no config or trigger detail `snapshot-20261013`

---

11 **Higher file limit for Context Hub repositories** IMPROVED 40

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

— Concrete number but prior limit and usage unspecified `snapshot-20261013`

---

12 **PII option in data protection policy dialogs** NEW 40

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

— Names the UI addition but not its effect or scope `snapshot-20261013`

└──▷ BREAKING ON UPGRADE

! Extended trace retention on LangSmith SaaS is capped at 180 days: existing settings above 180 days are silently applied as 180 days for new traces, and attempts to set a value above 180 days are rejected.

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

WAS THIS USEFUL? 👍 👎

Comet Opik

Feedback ⓘ 2.2.99 NOTES ↗ RANK

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

> Opik 2.2.99 adds event-driven alerting via AWS EventBridge and improves dashboard query management for Ollie chart widgets.

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

01 **AWS EventBridge publishing for triggered alerts** NEW

40

Opik now publishes triggered alerts to AWS EventBridge, enabling event-driven workflows when monitoring rules fire.

— Names integration but no config or endpoint details 2.2.99

02 **Ollie chart widget queries run by reference** IMPROVED

33

Saved Ollie chart widget queries can now be run by reference rather than inline, improving dashboard query management.

— Describes change but no UI path or API detail 2.2.99

WAS THIS USEFUL? 👍 👎

Langfuse

Feedback ⓘ v4.58.0 NOTES ↗ RANK

Langfuse provides tracing, evaluation, and monitoring for LLM applications.

> Langfuse added per-project trace idle time settings and Topics configuration for LLM connections, improved its media upload pipeline, and nudged self-hosted v3 users toward upgrading to v4.

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

01 **Per-project trace idle time settings** NEW 55

Adds `tracing_readiness_settings` support for per-project trace idle time configuration in the worker.

— Names config key but gives no values or usage example v4.58.0

02 **Media upload pipeline improvements** IMPROVED 45

Adds an upload pool with concurrency and byte limits for media uploads, and extracts media before TypeScript parsing in the OpenTelemetry pipeline.

— Describes mechanism but omits concrete limit values v4.58.0

03 **Self-hosted v3-to-v4 upgrade hint** NEW 45

Adds an upgrade hint in the version menu to prompt self-hosted v3 deployments to migrate to Langfuse v4.

— Names UI location but no migration steps given v3.225.12

04 **Topics configuration for project LLM connections** NEW 30

Enables configuring Topics on a project's own LLM connections.

— Bare description, no mechanism or config surface named v4.58.0

WAS THIS USEFUL? 👍 👎

Arize Phoenix

Feedback ⓘ arize-phoenix-sqlean-v0.2.0 NOTES ↗

Arize Phoenix is an open-source platform that monitors LLM application traces and evaluates their outputs.

└──▷ GET THIS VERSION

```
$ git clone --branch arize-phoenix-sqlean-v0.2.0
https://github.com/Arize-ai/phoenix.git
# already have the repo? check out this version:
$ git checkout arize-phoenix-sqlean-v0.2.0
```

NEW SUPPORTED PLATFORM Python 3.15 support

UPDATED SUPPORTED PLATFORM SQLite 3.54.0 bundle

WAS THIS USEFUL? 👍 👎

Weights & Biases Weave

Feedback ⓘ v0.53.11 NOTES ↗

RANK

Weights & Biases Weave traces, evaluates, and monitors LLM applications for developers.

> Weave expanded its Agents/conversations tagging and insights surface, grew its Stainless-generated trace-server client with new write paths, and added federated identity authentication plus PII redaction for Call and Agent writes.

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

01 Stainless-based trace-server client expansion IMPROVED

65

Makes Stainless the default HTTP client for Weave's trace-server communication, adds the `calls_complete` write path to the generated trace-server client enabling call completion to be submitted through the auto-generated API layer, and binds the remaining Stainless TSI routes in the generated client, expanding the set of trace-server API operations accessible through the SDK.

— Names concrete write path and client change, no sample call v0.53.9 v0.53.10

02 Conversation tagging and insights in Agents UI NEW

60

Adds `conversation_tag_assignments` tables to support tag-based scoring of conversations, adds insights filters for the Agents/conversations UI, and adds agent identity to conversation tags so the UI surfaces which agent participated in each tagged conversation.

— Names the table and UI area but no usage steps v0.53.10 v0.53.8

THINNER COVERAGE BELOW

03 PII redaction in Call and Agent writes NEW

45

Redacts PII in Call and Agent writes to reduce sensitive-data exposure in traces.

— States mechanism and scope but no config or opt-out details v0.53.8

04 Federated identity support NEW

35

Adds federated identity support to Weave, enabling users to authenticate via federated credentials.

— Describes capability but no setup or provider specifics v0.53.8

05 Playground support for Dedicated Inference NEW

35

Adds support for Playground requests routed to Dedicated Inference.

– Names the two surfaces but no mechanism or steps v0.53.10

06 **Explicit token-total preservation** IMPROVED 35

Preserves explicit token totals passed by the caller instead of overwriting them with inferred values.

– Explains before/after behavior but no field or API name v0.53.10

07 **SDK attribution under wandb.sdk namespace** IMPROVED 35

Standardizes SDK attribution under `wandb.sdk` for consistent telemetry source identification.

– Names the namespace but no usage or impact detail v0.53.11

WAS THIS USEFUL? 👍 👎

UK AI Security Institute Inspect AI Feedback ⓘ

0.3.279 CODE ↗ RANK

Inspect is the UK AI Security Institute's framework for large language model evaluations, covering datasets, solvers, scorers, and agentic eval tasks.

> Inspect expanded its Sentinel monitoring system, giving monitors broader visibility into agent model calls and richer context about the evaluation being run.

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

01 **Expanded visibility for Sentinel monitors** IMPROVED 60

Sentinel monitors gain access to `step.input`, which holds the exact model input even after transcript events are evicted (or None when unavailable). Monitors can now watch every agent's model calls, not just the agent that registered the monitor, and they also receive the task and sample descriptions, giving context about what is being evaluated.

– Names `step.input` but no code example or full API shown. 0.3.279

WAS THIS USEFUL? 👍 👎

Braintrust

Feedback ⓘ snapshot-20261013 NOTES ↗ RANK

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

> Braintrust's October 2026 release centers on new dashboard tooling — a Coding Agent Insights dashboard, SQL-driven charts, and Loop-powered chart building — alongside SDK/CLI improvements for cancellable evaluations, batch workflow scoring, project group placement, and several breaking span metadata renames.

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

01 **Coding Agent Insights dashboard** NEW 85

A new dashboard at `Dashboards > Coding agent insights` shows usage by user, API key, model, tool, and branch with cost breakdowns (requires data plane v2.13.0+); a daily Loop automation inside it surfaces recurring problems and inefficiencies across team coding-agent sessions.

Navigate to the Coding Agent Insights dashboard to review per-user and per-model spend across your team's coding agents.

📍 In the Braintrust console, go to Dashboards › Coding agent insights, then filter by user, API key, model, or branch to drill into cost breakdowns.

— Names UI path, requirement, and example navigation steps snapshot-20261013

02 **Cancellable evaluations via cancel_event** NEW 85

Eval() and EvalAsync() now accept a `cancel_event` (`threading.Event`) parameter; tasks can check it via `hooks.cancel_event`, and a timeout also sets it, enabling early termination of long-running evaluations.

Stop a long-running evaluation early from a CI timeout or signal, using the new cancel_event hook.

```python
import threading
from braintrust import Eval

cancel = threading.Event()

async def my_task(input, hooks):
    if hooks.cancel_event.is_set():
        return None
    # ... task logic ...

await Eval('my-project', data=cases, task=my_task, scores=[...],
cancel_event=cancel)
```

— Includes runnable code example and exact parameter names `snapshot-20261013`

03 **SQL charts for dashboards and sandbox** NEW 80

Adds SQL charts for building dashboard charts from a SQL query, switching between query text and a visual builder that edits the same query (available on Pro and Enterprise plans, requires data plane v2.1.1+); the SQL sandbox also gains chart-from-query with save-to-dashboard.

— Named plan tiers and version requirement but no exact UI steps `snapshot-20261013`

04 **Project group name support when pushing functions** NEW 70

Adds `projectGroupName` (TypeScript SDK v3.36.0+) and `project_group_name` (Python SDK v0.44.0+) support in `bt functions push`, plus `project_group_name` support in projects.create() (Python SDK), to place new or existing projects into the correct project group.

— Names exact flags and SDK versions across two SDKs `snapshot-20261013`

05 **Batch workflow evaluation support in Python SDK** NEW 65

Adds `WorkflowBatchTask` and `WorkflowBatchScorer` to the Python SDK for submitting many cases per provider batch in workflow evaluations.

— Names classes but no usage detail or example `snapshot-20261013`

06 **Copy logs into datasets via MCP or Loop** NEW 65

Adds the ability to copy logs into datasets via the Braintrust MCP server or Loop, preserving input, metadata, tags, and source-log links without passing full contents through the assistant context; requires data plane v2.18.0+.

— Names mechanism and version requirement but no exact command `snapshot-20261013`

07 **Loop on dashboards** NEW 60

Loop can now be opened from an individual dashboard or chart editor to investigate data and build or refine charts in natural language, with automatic dashboard/chart context injection.

— Describes mechanism but lacks a concrete entry point or command `snapshot-20261013`

THINNER COVERAGE BELOW

08 **bt trace hook exit code on failure** IMPROVED 55

`bt trace hook` now returns a nonzero exit code when it cannot capture an event, such as invalid tracing settings or an unreachable background tracing process.

— Names exact command and failure conditions `snapshot-20261013`

09 **bt update now syncs tracing plugins** IMPROVED 45

Running `bt update` now also updates installed tracing plugins even when `bt` itself is already current.

— Clear command but minimal behavioral detail `snapshot-20261013`

└──▷ **BREAKING ON UPGRADE**

! 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

NoLabs nono

Feedback ⓘ v0.80.0 NOTES ↗ CODE ↗ RANK

The nono runtime runs AI agents in isolated environments with policy-controlled filesystem, network, and credential access.

> nono v0.80.0 strengthens command policy with executable integrity pinning, per-command AWS credential signing, and macOS keychain access grants for tool sandboxes.

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

01 SHA-256 executable pinning for commands NEW 85

Adds a `sha256` field to `CommandPolicyConfig` in command policy, letting operators pin a command to a specific SHA-256 executable digest so substituted or tampered binaries are blocked from running under that policy.

Pin a sandboxed tool to its known-good binary to prevent a tampered executable from running under its command policy.

json

```
{
  "command_policies": {
    "commands": {
      "gh": {
        "executable": "/usr/bin/gh",
        "sha256":
"e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
",
        "from": {
          "session": {
            "sandbox": {
              "fs_read": ["."]
            }
          }
        }
      }
    }
  }
}
```

— Named config field with a runnable example config v0.80.0

02 Per-command AWS SigV4 auth in sandboxes NEW 78

Adds an `aws_auth` field of type `AwsAuthConfig` to `CommandCredentialConfig` in command policy, enabling per-command AWS SigV4 request signing via tool-sandbox scoped-proxy routes so child processes only ever see dummy keys rather than real AWS credentials.

— Named config fields but no usage example given v0.80.0

THINNER COVERAGE BELOW

03 **macOS keychain access grants in command policy** NEW

55

Adds a command-level `keychain_access` grant for macOS, allowing per-command access to the system keychain to be scoped within tool-sandbox command policies.

— Names the grant but no config example or mechanism detail v0.80.0

WAS THIS USEFUL? 👍 👎

agent-safe-pipeline

Feedback ⓘ v0.3.8 NOTES ↗ RANK

Agent Safe Pipeline provides a security-focused framework for building AI agent workflows with policy checks and safety controls.

> Agent Safe Pipeline's v0.3.7 release significantly expanded distribution of the `govern` gate and `agentsafe` runtime across platforms, package managers, and container registries, added card hold routing and distributed hold state, and renamed `GateDecision` to `DecisionGate` as a breaking change; v0.3.8 added a signed standing signal for Koard's Tap to Authenticate.

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

01 **Multi-platform distribution for govern and agentsafe runtime**

NEW

92

`govern` 2.1.0 can be installed via `go install github.com/decionis/agent-safe-pipeline/govern/v2/cmd/govern@v2.1.0`, with static binaries for macOS, Linux, and Windows (`govern-<version>-windows-x64.zip`), each shipped with a CycloneDX SBOM (`govern-<version>.cdx.json`). The runtime container image `ghcr.io/decionis/agentsafe:0.4.0` (plus init sidecar `ghcr.io/decionis/agentsafe:0.4.0-init`) is published to both `ghcr.io` and `docker.io/decionis/agentsafe:0.4.0` and can be verified with `gh attestation verify`. A Helm chart ships as `agentsafe-<chart version>.tgz` and at `oci://ghcr.io/decionis/charts/agentsafe`, and the runtime is also published to npm as `@decionis/agentsafe` and `@decionis/agent-safe-pipeline@0.3.7`.

Install the `govern` workflow gate on any platform and verify its integrity before running it in CI.

```
$ go install github.com/decionis/agent-safe-
pipeline/govern/v2/cmd/govern@v2.1.0
shasum -a 256 -c SHA256SUMS
```

Verify the attested runtime container image before pulling it into a Kubernetes or CI environment.

```
$ gh attestation verify
oci://ghcr.io/decionis/agentsafe@sha256:fa683a9ddc383a274f162a004
74430d3ced9231715820cd38b057076bc6967f9 --repo decionis/agent-
safe-pipeline
```

— Names every install command, binary, image tag and package. v0.3.7

THINNER COVERAGE BELOW

02 **GateDecision type renamed to DecisionGate**

BREAKING

55

The `GateDecision` type is renamed to `DecisionGate`; any code referencing `GateDecision` will break on upgrade.

— Exact renamed type named, clear migration action. v0.3.7

03 **Card hold routing and distributed hold state** NEW 50

Card holds are now routed to the cardholder's tap and the issuer app can resolve them; holds are keyed by card and merchant and shared across replicas, giving the cards

pipeline distributed hold state. When no hold is present, the pipeline evaluates the authorization directly at the commit boundary as a fallback path.

— Explains mechanism but gives no commands, flags or endpoints. v0.3.7

04 **Signed standing signal for Koard Tap to Authenticate** NEW

25

Producers can carry a signed standing signal alongside a proposal, enabling Koard's Tap to Authenticate flow.

— Bare description with no mechanism or usage detail. v0.3.8

└──▷ **BREAKING ON UPGRADE**

! The `GateDecision` type is renamed to `DecisionGate`; any code referencing `GateDecision` will break on upgrade.

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ **Open-Weight Models**

DeepSeek V4 Flash 0731 Feedback ⓘ

price cut · Oct 12, 2026 MODEL ↗

DeepSeek V4 Flash 0731 is a chat model from DeepSeek.

Price cut on OpenRouter (deepseek/deepseek-v4-flash-0731): $0.0152 → $0.016 in, $1.28 → $0.64 out per million tokens (-50%), first seen 12 Oct 2026.

WAS THIS USEFUL? 👍 👎

DeepSeek V4 Pro 0813 Feedback ⓘ

price increase · Oct 12, 2026 MODEL ↗

DeepSeek V4 Pro 0813 is a chat model from DeepSeek.

Price increase on OpenRouter (deepseek/deepseek-v4-pro): $0.2088 → $0.9553 in, $0.4176 → $1.9105 out per million tokens (+358%), first seen 12 Oct 2026.

Price increase on OpenRouter (deepseek/deepseek-v4-pro-0813): $0.66 → $1.32 in, $1.98 → $3.96 out per million tokens (+100%), first seen 12 Oct 2026.

WAS THIS USEFUL? 👍 👎

Z.ai GLM 5.3 Feedback ⓘ price increase · Oct 12, 2026 MODEL ↗

GLM 5.3 is a reasoning model from Z.ai.

Price increase on OpenRouter (z-ai/glm-5.3): $0.039 → $0.079 in, $4.80 → $4.80 out per million tokens (+103%), first seen 12 Oct 2026.

WAS THIS USEFUL? 👍 👎

Google Gemma 4 26B A4B Feedback ⓘ

price increase · Oct 12, 2026 MODEL ↗

Gemma 4 26B A4B is a multimodal model from Google.

Price increase on OpenRouter (google/gemma-4-26b-a4b-it): $0.0675 → $0.09 in, $0.225 → $0.30 out per million tokens (+33%), first seen 12 Oct 2026.

WAS THIS USEFUL? 👍 👎

Meta Llama 4 Scout

Feedback ⓘ price increase · Oct 12, 2026

MODEL ↗

Llama 4 Scout is a multimodal model from Meta.

Price increase on OpenRouter (meta-llama/llama-4-scout): $0.10 → $0.18 in, $0.30 → $0.59 out per million tokens (+97%), first seen 12 Oct 2026.

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · October 13, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue