NOW AVAILABLE The 512GB Mac Studio is on sale now, shipping February to March. See every configuration.

THE AI TOOLCHAIN

NO. 091

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED NOVEMBER 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 447 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 104 of 106 tools

▾ // THE BRIEF

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

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

vim ? help

Three unrelated things stand out today: opensandbox resolves secrets per sandbox request instead of handing credentials to code, TrueForge can pause an agent turn for human approval before tools run, and mlxtop detects slowdowns and stalls with replayable session evidence.

GOVERN

Let agents use credentials without seeing them

Both releases move secrets behind a vault boundary: opensandbox resolves them per request in the sandbox path, while codealta lets agents reference provider credentials without exposing the values. That removes the common workaround of injecting long-lived secrets into agent runtime state, where prompts, logs, or tool output can leak them.

opensandbox · codealta

GOVERN

Stop a risky agent action before it executes

TrueForge added paused turns for human tool approvals and questions; synthadoc added run-scoped confirm gates for agentic workflows. Teams can now make review a first-class runtime state instead of bolting on a separate ticket, Slack thread, or manual rerun after the agent has already acted.

TrueForge · synthadoc

OPERATE

Catch inference stalls with evidence you can replay

mlxtop now detects slowdown and stall anomalies and records them in a replayable session Journal. Instead of staring at live terminal metrics or trying to reconstruct a bad run from logs, operators get a captured trace of the failure window to debug runtime, cache, and adapter behavior.

mlxtop

DEPLOY

Ship cloud agent infrastructure with fewer hand edits

Crusoe Cloud added Terraform resources for external load balancers and Kubernetes node-pool autoscaling; agents-cli added deploy pre-flight checks plus flags to remove env vars and secrets during deploy. That turns more rollout state into declared, checked infrastructure, so bad configuration is caught before launch and stale credentials are not left behind by hand.

Crusoe Cloud

Does **GPT-5.5** read better? 👍 👎

OFFENSIVE SECURITY

◆ Exploitation & C2

xalgorix

Feedback ⓘ v4.6.188 NOTES ↗ RANK

The xalgorix platform runs AI pentesting agents for reconnaissance, vulnerability detection, and exploitation workflows.

> xalgorix hardened its HTTP tools and scan-submission API against SSRF, tightened validation of per-scan `discord_webhook` and `api_base` fields, made the automatic Go memory limit scale with host RAM, and fixed scan pause/resume edge cases.

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

01 RAM-scaled automatic Go memory limit IMPROVED

95

The automatic Go soft memory limit now scales with host RAM instead of capping at 4 GB: hosts up to 12 GB use 45% of RAM (up to 4 GB), larger hosts use one third of RAM with a 32 GB ceiling. `XALGORIX_GO_MEM_LIMIT_MB` and `GOMEMLIMIT` still take precedence in that order, and an empty `GOMEMLIMIT=` is now treated as unset — re-enabling the automatic limit, where previously an empty value disabled it entirely.

Override the automatic Go memory limit on a large host to give the engine more headroom than the default one-third-of-RAM formula provides.

```
$ export XALGORIX_GO_MEM_LIMIT_MB=20480
./xalgorix
```

— Exact formula, env var precedence, and a runnable override command v4.6.187

02 discord_webhook SSRF validation at scan submission BREAKING

78

A scan request that sets a `discord_webhook` pointing at the local machine, dashboard listener, link-local, multicast, or cloud metadata address now receives a 400 response (or a WebSocket `error` event) instead of being accepted, so misconfigured webhook URLs fail immediately rather than misfiring later.

Start a scan with a private LAN relay webhook — the engine now validates this at submission time, so a misconfigured URL fails immediately with 400 rather than silently misfiring later.

```
$ curl -X POST https://xalgorix.example.com/api/scan \
  -H 'Content-Type: application/json' \
  -d '{"targets":
["10.0.0.5"],"discord_webhook":"http://192.168.1.50:9000/relay"}'
```

— Named field, exact failure codes, and a runnable curl example v4.6.187

03 **SSRF protections for http_request and send_request** BREAKING

67

`http_request` and `send_request` now block requests that target the local machine or dashboard listener, matching the rule already applied to other gated tools. Each redirect target is checked before being followed, including port-matching, and a refused redirect returns an explanation instead of being followed. Hosts with an IPv6 zone-ID literal (e.g. `[::1%eth0]`) or a DNS response with no usable address are now refused instead of being treated as unresolvable and allowed through.

— Clear mechanism and scope, no runnable example given v4.6.188

04 **api_base validation against stored credentials** BREAKING

67

Scan requests setting `api_base` are now validated: without an `api_key` in the request, `api_base` must stay on the endpoint of the stored credential; when a request supplies its own `api_key`, that key replaces stored keys and the key pool for the scan. A request violating this now receives a 400 response (or a WebSocket `error` event) instead of being accepted.

— Named field and exact rule, but no runnable example v4.6.187

THINNER COVERAGE BELOW

05 **Scan pause and resume reliability fixes** IMPROVED

56

Resuming a refused or still-shutting-down paused scan now answers 409 with a retry hint instead of 200, so the paused scan stays resumable. A paused multi-target queue with no saved queue state now resumes from the target it paused on rather than restarting from the first target.

— Describes behavior change but no config or command to act on v4.6.187

└──▷ **BREAKING ON UPGRADE**

! `http_request` and `send_request` now refuse requests whose URL resolves to the local machine or dashboard listener — previously these calls were allowed through unchecked.

! `http_request` and `send_request` now refuse redirects that point at the local machine or dashboard listener instead of following them.

! Hosts with an IPv6 zone-ID literal (e.g. `[::1%eth0]`) or a DNS response with no usable address are now refused by the local-machine check; previously both were treated as unresolvable and allowed through.

! A scan request that sets a `discord_webhook` pointing at the local machine, dashboard listener, link-local, multicast, or cloud metadata address now receives a 400 response (or a WebSocket `error` event) instead of being accepted.

! A scan request that sets `api_base` to a host outside the stored credential endpoint without supplying its own `api_key` now receives a 400 response (or a WebSocket `error` event) instead of being accepted.

! An empty `GOMEMLIMIT=` environment variable is now treated as unset, re-enabling the automatic memory limit; previously, an empty value disabled the automatic limit entirely.

WAS THIS USEFUL? 👍 👎

BUILD

◆ AI App Builders

Vercel v0

Feedback ⓘ changelog · Nov 13, 2026 CHANGELOG ↗ RANK

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

> v0 adds ChatGPT Business plan billing integration and an on-demand security review feature for catching vulnerabilities before merging.

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

01 **ChatGPT Business plan support** NEW 75

Users can connect a ChatGPT Business account to v0 via Sign in with ChatGPT so that eligible AI responses count against the ChatGPT plan instead of v0 credits. Workspace admins can set usage limits, v0 saves progress when a limit is reached, and surfaces a Copy request for admin action.

— Names mechanism and admin controls but no limits specifics changelog-20261113-b61f3c4f

02 **On-demand security review of changes** NEW 60

v0 can run a security check on recent changes to catch issues like missing auth checks, injection flaws, or SSRF before merging, triggered by running 'Run a security check.'

Trigger an on-demand security review of your recent changes to catch missing auth checks, injection flaws, or SSRF before merging.

> 📍 Run a security check.

— Gives trigger command but limited mechanism detail changelog-20261113-b61f3c4f

WAS THIS USEFUL? 👍 👎

◆ AI Coding Agents

SST OpenCode

Feedback ⓘ changelog · Nov 13, 2026 CHANGELOG ↗

RANK

OpenCode is an open-source AI coding agent for terminal-based development tasks.

> OpenCode's latest release adds scripted worktree sessions with archiving controls, GitHub Copilot Auto model routing, shared MCP server connections, and several GUI/plugin and SDK refinements, alongside a breaking change to the image response field.

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

01 Scripted sessions in isolated git worktrees NEW 84

Run scripted sessions in a new git worktree with `opencode run -w` or name it with `--worktree=<name>`, and keep CI and scripted sessions out of the inbox by passing `--archived` to `opencode run`.

Run a scripted CI session in an isolated git worktree and keep it out of your inbox.

```
$ opencode run -w --archived
```

Name a specific worktree for a scripted session so it is easy to reference later.

```
$ opencode run --worktree=feature-branch-review
```

— Runnable commands and flags with examples given changelog-20261113-e52ca2a2

02 Plugin session instructions API NEW 70

Plugins can attach and update keyed session instructions with `ctx.session.instructions.entry.put`, `ctx.session.instructions.entry.list`, and `ctx.session.instructions.entry.remove`.

— Named API methods, mechanism briefly explained changelog-20261113-e52ca2a2

03 Faster SDK endpoint responses during plugin loading IMPROVED 69

Several SDK endpoints — `agent.list`, `integration.list`, `usage.limits`, `vcs.init`, `rpc.call`, and `artifact.*` — now return immediately on locations still loading plugins rather than waiting up to 5 seconds.

— Named endpoints and behavior change, no direct action step changelog-20261113-e52ca2a2

04 **ImageResponse.image field removed** BREAKING 67

The `ImageResponse.image` field is removed; code that reads `response.image` must be updated to read `response.images[0]`.

— Exact migration path given for the breaking change changelog-20261113-e52ca2a2

05 **GitHub Copilot Auto model routing** NEW 63

OpenCode now supports GitHub Copilot Auto model routing — selecting Auto in the model picker lets Copilot's router choose the model per session at the Auto discount, falling back to the account's default chat model if routing fails.

— Mechanism and fallback explained, only a UI path to act changelog-20261113-e52ca2a2

THINNER COVERAGE BELOW

06 **Session and tab management improvements** NEW 58

Mark any session, project row, or inbox card as unread via its tab menu to keep the notification dot until next visited; side panel tabs also now support keyboard shortcuts to close tabs to the right, close the session tab with `Ctrl+Alt+W`, and reopen the last closed tab.

— Clear UI path given, no deeper mechanism Named shortcut given, others described but unnamed

changelog-20261113-e52ca2a2

07 **Shared MCP server connection pooling** NEW 46

OpenCode now supports shared MCP server connections — enabling pooling lets one connection be reused across projects with the same setup, reducing process count and startup time.

— Benefit described but no exact setting named changelog-20261113-e52ca2a2

08 **GUI plugin theming and code block actions** NEW 35

GUI plugins can now add color themes, render or add actions to code blocks, and use native settings and menu components.

— Lists additions but no APIs or surfaces named changelog-20261113-e52ca2a2

09 **Mobile composer touch dock** IMPROVED 25

On mobile, composer actions now appear in a touch dock above the keyboard with larger buttons.

— Describes UI change only, no further mechanism changelog-20261113-e52ca2a2

└──▷ BREAKING ON UPGRADE

! The `ImageResponse.image` field is removed; code that reads `response.image` must be updated to read `response.images[0]`.

WAS THIS USEFUL? 👍 👎

OpenAI Codex CLI

Feedback ⓘ rust-v0.169.0 NOTES ↗ CODE ↗

RANK

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

> OpenAI Codex CLI's terminal UI gained a full side-chat workspace, custom model-provider and gateway sign-in flows, WebRTC-based realtime voice calls, and per-conversation speed control for cloud turns, alongside audio attachments, animated GIF generation, and OSC 777 terminal notifications.

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

01 Side conversation panes and tabs in TUI workspace IMPROVED

88

Enable a side conversation pane beside the main chat with `features.tui_workspace`, dragging the divider to resize or pressing `leader z` to zoom the focused pane; the opt-in workspace supports multiple side-chat tabs switchable by keyboard or mouse, with tab close controls and local side chats that persist and reopen after restart.

Split the TUI into a main chat and a side conversation pane for running a parallel investigation without leaving your current context.

toml

```
features.tui_workspace = true
```

— Exact config key, keybindings and example given rust-v0.168.0 rust-v0.171.0-alpha.2 rust-v0.169.0

03 Custom model provider management and gateway sign-in NEW

75

Add or edit custom model providers from the terminal, with custom gateway onboarding that supports device-code authorization for SSH or browserless environments; the TUI sign-in flow adds custom gateway selection, saving the selection as `model_provider` and completing gateway OAuth before the first chat, allows creating custom providers during sign-in, and enables model catalog discovery with gateway OAuth. The default model provider for new chats can also be browsed and selected under Sessions → Model providers.

— Names config key and OAuth flow but no runnable command rust-v0.168.0 rust-v0.171.0-alpha.2 rust-v0.169.0

05 **Animated GIF generation and editing** NEW 72

Enable animated GIF generation and editing with optional frame count and timing controls using `features.image_generation_animation`.

Generate or edit animated GIFs directly from the chat with control over frame count and timing.

```toml
features.image_generation_animation = true
```

— Exact config key and example provided `rust-v0.168.0`

06 **Save diagnostic ZIP from terminal report editor** NEW 65

Save a diagnostic ZIP locally by pressing Ctrl+S in the terminal report editor, even when uploads are disabled.

— Exact keybinding and behaviour named `rust-v0.169.0`

08 **Image previews in tmux on Ghostty and Kitty** NEW 62

View image previews and pets in tmux on Ghostty and Kitty terminals in fullscreen mode with `allow-passthrough all`.

— Named terminals and exact flag given `rust-v0.168.0`

THINNER COVERAGE BELOW

11 **Collapse turn details in full-screen transcript** NEW 50

Enable Collapse turn details in the full-screen transcript to fold completed activity beneath each prompt while keeping answers visible and tool output searchable.

— Named UI toggle but no exact path `rust-v0.169.0`

12 **Audio attachments for audio-capable models** NEW 50

Attach audio recordings when using an audio-capable model; Codex skips confirmed rejected recordings on later turns and uses clip duration to keep more short recordings in context.

— Describes behaviour but no config key or command `rust-v0.168.0`

13 **Task search in agent command center** NEW 45

Search older tasks in the agent command center by name, prompt preview, or working directory.

— Names the UI area but no command or key `rust-v0.169.0`

17 **Cloud image messages and editable draft handoffs** NEW

30

Support Cloud image messages and editable draft handoffs.

— Thin one-line description across two releases rust-v0.171.0-alpha.2 rust-v0.169.0

WAS THIS USEFUL? 👍 👎

yetone magpie

Feedback ⓘ v0.1.4032 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.

> This window's headline additions are configurable web search across ten provider APIs (exposed to agents via MCP), per-key billing price rates, and Claude Code advisor tool support with per-key model restriction enforcement, alongside new installer flags, provider presets, an isolated Claude Code folder for Desktop, and per-host connection routing.

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

01 **Per-key billing price rates** NEW 95

Each API key under a provider can now have its own price rate set via `SetKeyPriceRate`, stored in `KeyPriceRates` on the provider, so usage accounting on the Usage page, spend caps, and typed balances reflect the per-key multiplier rather than the shared `PriceRate`. Retrieve a key's effective rate with KeyPriceRate(keyID), which falls back to the provider's `PriceRate` when unset, and pass `keyPriceRates` in the JSON body of a `POST /api/provider/save` request to set, update, or clear per-key rates in bulk.

Assign a 0.5x rate to a cheap key and 1.2x to a premium key on the same relay endpoint so spend tracking reflects actual billing.

```
$ POST /api/provider/save
Content-Type: application/json

{
  "id": "relay",
  "from": "relay",
  "name": "Relay",
  "key": "",
  "chat": "https://relay.example/v1",
  "icon": "generic",
  "models": [],
  "headers": {},
  "keyPriceRates": {
    "<plain-key-fingerprint>": 0.5,
    "<premium-key-fingerprint>": 1.2
  }
}
```

— Named functions, fields, and a runnable endpoint with example. v0.1.4032

02 Installer flags for build type, network routing, uninstall

NEW

85

The install script gains `--gui` or `--tui` flags to choose a desktop or terminal-only build, `--proxy` or `--mirror` flags to route installs through a corporate firewall, and `--uninstall` to remove magpie and restore agent configs, with an optional `--purge` flag to also delete settings in `~/.config/magpie`.

Force a terminal-only install on a headless server or when you want no desktop dependency.

```
$ curl -fsSL https://usemagpie.ai/install.sh | sh -s -- --tui
```

Remove magpie behind a firewall while keeping settings, or add `--purge` to wipe ~/.config/magpie too.

```
$ curl -fsSL https://usemagpie.ai/install.sh | sh -s -- --
uninstall --purge
```

— Names every install flag with runnable commands. product docs

03 New search and inference provider integrations NEW

85

magpie adds configurable web search across Tavily, Brave, Exa, Firecrawl, You.com, DeepSeek, Doubao Search on Volcengine, Yuanbao Search on Tencent Cloud, TinyFish, and SearXNG, with drag-to-reorder priority among providers, per-key round-robin or ordered selection via `pick=order`, and usage inspection via the `magpie search` command; it also adds provider presets for Amazon Bedrock Mantle, Vercel AI Gateway, GMI Cloud, Fireworks Nexus, Doubao Search on Volcengine, and Yuanbao Search on Tencent Cloud, alongside support for any OpenAI- or Anthropic-compatible URL.

Inspect Tavily key quota usage and trigger ordered (non-round-robin) key selection across multiple keys.

```
$ magpie search
```

— Names providers, selection flag, and inspection command. Lists named presets but no configuration steps.

product docs

04 **Claude Code integration enhancements** NEW 80

Enabling `DesktopIsolatedClaude` in settings sets `CLAUDE_CONFIG_DIR` at launch so Claude Code's Desktop Code tab writes its `.claude.json`, backups, plugins, and sessions to its own folder instead of `~/.claude`; separately, setting `CLAUDE_CODE_ENABLE_EXPERIMENTAL_ADVISOR_TOOL` (Claude Code 2.1.316+) enables Claude Code's advisor tool for any model magpie serves, with magpie running the advisor call itself when the upstream cannot, configurable via the `advisor` field in magpie settings, and keys restricted to a model list now have that restriction enforced on the advisor model too via the `keyAllows` function, which checks the calling key's permitted models against any advisor model the client names in the request so a restricted key cannot reach other models through the advisor path.

— Names the setting, env var, and affected files. Names env var, config field, and enforcement function.

v0.1.4029 v0.1.4023 v0.1.4025

---

05 **Per-host connect address routing via Resolve field** NEW 75

A new Resolve field on a provider record routes that provider's outbound TCP connections to specific IP addresses or DNS names while leaving the TLS server name and Host header unchanged — useful when a managed network routes domains through an internal SNI gateway.

— Names the field and explains mechanism and use case. v0.1.4012

THINNER COVERAGE BELOW

06 **Telemetry opt-out controls** NEW 55

Opt out of usage telemetry fully or partially in Settings → Privacy, or by setting the environment variable `DO_NOT_TRACK=1`.

— Names UI path and env var, nothing further. product docs

---

07 **Release binary provenance verification** NEW 50

Verify every release binary's build provenance with `gh attestation verify -R yetone/magpie-releases`.

— Runnable command but no mechanism explained. product docs

---

08 **Web search exposed as MCP server tools** NEW 50

Any agent can expose the configured web search as an MCP server, providing a `web_search` tool and a `web_fetch` tool.

— Names the two MCP tools but no setup steps. product docs

---

09 **Kimi Work session usage tracking** IMPROVED 45

magpie now reads API call usage from the Kimi desktop app's Work agent sessions, counting them under the `kimi` agent alongside regular Kimi Code calls.

— States what's tracked but no setup or limits. v0.1.4017

NEW SUPPORTED PLATFORM Gemini 4 Carbon and Hydrogen on Vertex (global, US, EU)

NEW SUPPORTED PLATFORM xhigh effort level for Gemini 4 Hydrogen on Gemini's own API

v0.1.4016 CODE ↗ STABLE

└──▷ GET THIS VERSION ▾

Docker

```
$ docker pull ghcr.io/yetone/magpie:0.1.4016
```

git

```
$ git clone --branch v0.1.4016 https://github.com/yetone/magpie.git
# already have the repo? check out this version:
$ git checkout v0.1.4016
```

published under the project's GitHub account · checked 13 Nov Wrong command?

WAS THIS USEFUL? 👍 👎

Letta Code

Feedback ⓘ v0.34.59 NOTES ↗ RANK

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

> Letta Code's latest releases focus on small operational upgrades: MCP reminders gain file-based argument input, the Monitor tool gets a new cloud data source and persistent state, and the watch-PR listener adds event filtering.

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

01 **MCP reminders support `--args-file` with exit code guidance**

IMPROVED 65

MCP reminders now accept `--args-file` to pass arguments from a file, with added exit code guidance surfaced alongside the reminder.

– Names exact flag but lacks file format or example v0.34.59

THINNER COVERAGE BELOW

02 **Monitor tool gains Cloud URL source and API-persisted state**

IMPROVED 45

The Monitor tool now supports a Letta Cloud URL source, expanding where Monitor can pull data from. Cloud-agent monitor state now persists through the API, so monitor status survives agent restarts and reconnects.

– Describes two additions but no config or endpoint names v0.34.57 v0.34.59

03 **PR event filtering in watch-PR listener** NEW 40

PR event filtering and workflow run watching are now available in the watch-PR listener.

– Brief mention without usage detail or config options v0.34.59

WAS THIS USEFUL? 👍 👎

forgedock

Feedback ⓘ v1.58.0 NOTES ↗ RANK

ForgeDock — autonomous AI dev pipeline for Claude Code. GitHub as agent memory: issue in, PR out.

> ForgeDock expanded its release pipeline with new `release status` / `release explain` commands, a verification engine, a gate catalogue, and ledger-backed release tracking, while also adding adaptive pipeline depth, on-demand test sandboxes, and new quality gates for docs and public pages.

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

01 New CLI commands for release inspection and stats NEW

85

ForgeDock adds `release status` and `release explain` commands with a stable JSON schema for inspecting release state and explaining why a release is blocked or passing directly from the CLI, plus `forgedock stats --card` to generate a shareable stats card that outputs aggregate-only metrics by default for reporting or dashboards without exposing per-issue detail.

Inspect the current release state and get a human-readable explanation of why a release is blocked or passing.

```
$ forgedock release status
forgedock release explain
```

Export aggregate pipeline metrics as a shareable card — useful for reporting or dashboards without exposing per-issue detail.

```
$ forgedock stats --card
```

— Named commands with runnable example and stable schema Runnable command with described default behavior

v1.58.0 v1.57.0

02 Disk usage reporting and garbage collection IMPROVED

70

ForgeDock's `gc` and `doctor` commands now report per-sandbox disk usage and agent free disk. ForgeDock also adds scheduled and disk-pressure garbage collection running outside batches, covering ForgeDock scratch space, lane compose volumes, and legacy worktrees.

– Named commands and GC targets, no flags given v1.57.0 v1.55.0

03 **Lean path pre-warming and batched prechecks** IMPROVED 70

ForgeDock's lean path now pre-warms lane sandboxes and worktrees — cutting up to 30 seconds of sandbox startup time per lane on small repos — overlaps lane setup (dependency installation, heavy shims, light eligibility checks, and context packing) with the investigation session, and runs precheck guards as a single batched test invocation concurrent with the review session.

– Concrete timing numbers and mechanism, no trigger command v1.55.0

04 **Release pipeline infrastructure and tracking** NEW 65

ForgeDock's release pipeline now handles deploy stages MERGED to LIVE, including `LIVE_CHECK` and `DEPLOY_FREEZE` conditions, adds a verification engine that closes issues automatically once verified, tracks release state on per-release branches using the `forge-ledger/<id>` and `forge-ledger/_active` naming scheme with configuration committed as `.forge/release.yaml` (via a vendored pinned YAML parser and unit map resolver), adds nightly release orchestration and CI minutes accounting, and adds a flake ledger that tracks same-SHA flips with auto-issue creation, expiry, and caps, governed by the `exempt` policy.

– Named stage conditions but no CLI entry point given Describes behavior but no mechanism detail or command Named branch scheme and config file, no CLI command shown Named exempt policy, mechanism described but no command

v1.58.0 v1.57.0

05 **Audited release control verb set** IMPROVED 65

Release control now uses one audited verb set — `hold`, `resume`, `cancel`, `lane`, `exempt`, and `freeze` — applied consistently across release commands.

– Named verbs, but no example command syntax given v1.57.0

06 **New release gates and gate system tooling** NEW 60

ForgeDock adds a gate catalogue, false-block telemetry, and a new-gate admission checklist to the release gate system, along with a design routing gate at `review-pr-staging` to enforce design review before staging merges, a docs link-integrity gate requiring a docs link before a milestone can close, and a page gate requiring an approved design, an intent-and-ICP SEO brief, and a GEO brief for every new or changed public page.

– Named components but no exact command or config shown Named gate location, opt-in described but no exact flag Clear enforcement point at milestone close, no command Named requirements, no exact config or command shown

v1.58.0 v1.57.0 v1.55.0

07 **New testing checks across the pipeline** NEW 60

ForgeDock's validate step now adds on-demand test sandboxes — DB, Redis, services, browser, and HTTP replay — created from the test plan when the issue requires them; its contract phase adds blast-radius regression sets and baseline suites and requires an executable post-deploy check before prod-only deferrals are accepted; and its review step adds a test-adequacy table per requirement, enforced for bugs, plus test-quality telemetry.

— Named sandbox types, no command to invoke directly Describes new requirement, no exact config or command Thin description of enforcement scope, no mechanism detail

v1.55.0

THINNER COVERAGE BELOW

08 **Adaptive pipeline depth** IMPROVED 45

ForgeDock's investigation phase now decides pipeline depth, letting trivial issues fast-forward directly to a single implement session instead of running the full pipeline.

— Explains behavior change, no named flag or config v1.55.0

09 **Issue-watch reconciliation with batch outcomes** IMPROVED 30

ForgeDock's issue-watch now reconciles claims with batch outcomes and drives parked lanes through remediation.

— Vague mechanism, no named surface or command v1.55.0

10 **Architecture decision register** NEW 20

Bind architecture decisions and interface contracts into the decision register with the new architecture feature.

— Thin prose description, no command or schema given v1.57.0

WAS THIS USEFUL? 👍 👎

Alibaba Qwen Code

Feedback ⓘ desktop-v0.27.3 NOTES ↗

RANK

Qwen Code is an open-source AI coding agent that lives in your terminal.

> Qwen Code expanded its channel integrations with a new DingTalk webhook alongside Email, WeChat and QQ, added manual PR binding via a new `/pr` command, and gave Mods plugins persistent state with `$.store`, while also extending managed-agent SPIs, session journal durability and an opt-in event-bus agent loop.

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

01 Mods plugin persistence and reply rewriting NEW

85

Mods gain `$.store -based` plugin state persistence that survives across sessions, the ability to rewrite displayed reply rows in both the Ink and OpenTUI terminals (in OpenTUI via `ui.render`, while preserving stored message text and `/copy` output), and the ability to run model completions directly.

Persist a Mods plugin's counter or state between sessions so it survives restarts.

javascript

```
// Inside a Mods plugin
const count = ($.store.count ?? 0) + 1;
$.store.count = count;
```

— Runnable `$.store` code example plus exact API names. v0.27.3 desktop-v0.27.3

02 Managed session storage and journal durability IMPROVED

80

Managed sessions now reuse the host shell in the web shell, store oversized resources and compacted closed journals in object storage while preserving recovery and normal reads, durably store self-paced `/loop` wakeups in the session journal so they survive restarts and takeovers, and support recovery of evicted journal records from journal packs via the `qwen.managed-agent.session-resources.deletion-enabled` setting.

— Names a config key and storage mechanism. desktop-v0.27.3 v0.27.3

03 New channel integrations: Email, WeChat, QQ, DingTalk NEW

75

Adds channel manager configuration in the web shell for the Email channel, in-browser QR login for the WeChat channel, a QR-login management contract plus a QQ config

surface for QR-based channel authentication, and a sending-only DingTalk webhook channel that posts notifications and successful task results as Markdown to configured DingTalk custom group robots, with DingTalk webhook destinations configurable in Web Shell and selectable by Scheduled Tasks as saved Channel destinations.

— Names each channel and config surface but no exact command. desktop-v0.27.3 v0.27.3

04 **Managed MCP live selection limits** IMPROVED 75

Managed MCP can now propose live-only selection capacity; it counts only up to 32 live selections, retains removed entries for recovery, keeps the 16-enabled limit, and caps retained snapshots at 96 entries.

— Precise numeric limits but no direct command. v0.27.3 desktop-v0.27.3

05 **PR Binding: Host Discovery, Ordered Shutdown, and Manual /pr Command**

NEW 70

Adds PR-watch host discovery and ordered shutdown to the serve layer, and introduces the new `/pr` command (plus a Web Shell dialog) to manually bind a session to a pull request.

Manually bind an open terminal session to a specific pull request so the agent can read PR context and post review comments.



— Single terse line with no mechanism detail. Runnable command demonstrated with usage example.

desktop-v0.27.3 v0.27.3

06 **Opt-in event-bus agent loop** NEW 70

Adds an `experimental.unifiedAgentLoop` experimental setting to enable the event-bus agent loop on headless CLI runs, and an `AgentLoopOptions.useEventBus` option to opt in to running the shared agent loop on event-bus handlers without changing observable behavior.

— Names exact settings though no usage example given. v0.27.3

07 **Managed-agent SPIs for Skill and extension import** NEW 60

Managed-agent gains SPIs that let embedding applications import exact Skill or extension revisions from external sources and store compiled archives in private storage, including Skills import from the original Runtime when initialising a managed agent.

— Names the SPI capability but not method signatures. v0.27.3 desktop-v0.27.3

THINNER COVERAGE BELOW

08 **Web Shell composer and session list updates** IMPROVED

55

Adds session speed and cache-hit rate display under the composer, unfinished background task chips above the composer, and sidebar session lists that now expand ten sessions per click with labels showing remaining count instead of showing all at once.

— Describes UI elements without exact navigation path. desktop-v0.27.3 v0.27.3

09 **Localized bound-session event prompts** IMPROVED 40

Bound-session event prompts are now localized by UI language, including Chinese localization for GitHub channel event prompts in PR-bound sessions when the channel UI language is Chinese.

— Thin description with no config key or steps. desktop-v0.27.3 v0.27.3

10 **Managed-agent session recap and observability metrics** NEW

40

Adds private BTW and session recap to managed-agent sessions, and Tool v4 execution measurements to the managed-agent observability surface.

— Lists additions without usage detail. desktop-v0.27.3

11 **Channel delivery segments and watch gating** NEW 40

Exposes channel delivery segments in the public read model and gates review/verify bound delivery on active watches in the channels layer.

— Abstract internal-layer description, no usage path. desktop-v0.27.3

12 **BTW session history threading in CLI** IMPROVED 35

Threads `/btw` session history and decouples its request lifecycle in the CLI.

— Single terse line, mechanism only hinted. desktop-v0.27.3

WAS THIS USEFUL? 👍 👎

GitHub Copilot CLI

Feedback ⓘ v1.0.110 NOTES ↗ RANK

GitHub Copilot CLI is a terminal-based coding agent that can plan, write, and test code in local repositories.

> GitHub Copilot CLI v1.0.110 adds a masked dialog for agent password prompts, JSON-settings reads via `copilot config`, and several reliability fixes across MCP config handling, session management, and shell completion.

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

01 MCP config error handling across commands IMPROVED

85

`copilot -p` now warns on stderr and in JSON output when MCP servers cannot start because `mcp-config.json` cannot be parsed, instead of continuing silently; `copilot mcp remove` and `/mcp delete` now report missing MCP servers as not found rather than suggesting a disable command that cannot find them; and `copilot mcp enable` now works for most servers even when `mcp-config.json` is invalid.

— Names all affected MCP commands and exact before/after behavior. v1.0.110

02 JSON settings read via copilot config NEW

75

`copilot config <key>` or `/settings show <key>` can now read JSON-valued settings such as hook definitions, for example `copilot config hooks.postToolUse`.

Inspect a JSON-valued setting such as a hook definition from the CLI without opening a config file.

```
$ copilot config hooks.postToolUse
```

— Exact command and runnable example provided. v1.0.110

03 Masked prompts for agent password requests NEW

70

Agent commands running git, ssh, and sudo that request passwords are now answered through a masked dialog; this feature is experimental and enabled with `/experimental on`.

— Names enable command and scope but mechanism is thin. v1.0.110

04 List default-off plugin rules NEW

65

`copilot instruction list` now shows which plugin rules are disabled by default, letting users see them before selectively enabling for a project.

See which plugin rules are disabled by default before selectively enabling them for a project.

```
$ copilot instruction list
```

— Exact command and example, little further mechanism. v1.0.110

05 **Shell completion for new flags** BREAKING 65

Shell completion now works after `--effort`, `--no-remote-export`, and bare `--sync-session` flags; users must regenerate their completion script to apply these updates.

— Names flags and required migration action. v1.0.110

06 **Warning for ignored bypass-permission flags** IMPROVED 60

The CLI now shows a warning when `--yolo` or `--allow-all` flags are ignored because bypass permissions mode is disabled, including during non-interactive runs.

— Names exact flags but no further mechanism given. v1.0.110

07 **Show non-invocable custom agents in /subagents** IMPROVED 60

`/subagents` now lists custom agents marked `user-invocable: false`, so users can configure their models and toggle them on or off.

— Names field and command, limited further detail. v1.0.110

THINNER COVERAGE BELOW

08 **Skip in-use sessions during prune** IMPROVED 55

`/session prune` and its dry run now skip sessions open in other processes, reporting how many sessions were skipped.

— Names command but lacks deeper mechanism detail. v1.0.110

09 **Separate token count for custom instructions** IMPROVED 55

`/context` now shows loaded custom instructions as a separate token count instead of folding them into the System Prompt count.

— Names command and change but thin on mechanism. v1.0.110

10 **Credit limit enforcement in /compact** IMPROVED 55

`/compact` now respects the session AI credit limit, preventing additional charges once the limit is reached.

— Names command and limit behavior, no further mechanism. v1.0.110

└──▷ BREAKING ON UPGRADE

! Shell completion scripts must be regenerated to pick up completions after `--effort`, `--no-remote-export`, and bare `--sync-session` flags.

WAS THIS USEFUL? 👍 👎

Anthropic Claude Code

Feedback ⓘ v2.1.323 NOTES ↗

CODE ↗ RANK

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

> Claude Code's v2.1.323 release introduces an experimental Code Mode for agentic JavaScript-based tool calls, adds plugin version constraints and stricter TypeScript validation, and ships several environment-variable controls for session behaviour.

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

01 **Plugin version constraints via engines field** NEW 90

Add `engines["claude-code"]` to `plugin.json` to declare a version range: plugins outside that range are not installed, updated, or loaded, and install/update commands pick the newest compatible version.

Gate a plugin to only load within a supported Claude Code version range, so users on older or newer incompatible versions get the right fallback.

json

```
{
  "name": "my-plugin",
  "engines": {
    "claude-code": ">=2.1.323 <3.0.0"
  }
}
```

— Exact config key, file, and runnable example given v2.1.323

02 **TypeScript type-checking in plugin validate** IMPROVED 90

`claude plugin validate` now type-checks a plugin's TypeScript hooks module, treating each finding as a warning unless `--strict` is used to fail the command on any finding; it also reports the `$` nouns a mod's `engine.create` hook adds, withholds, or replaces.

Catch TypeScript type errors in a plugin's hooks module before shipping, failing CI on any finding.

```
$ claude plugin validate --strict
```

— Exact command, flag, and behavior with runnable example v2.1.323

03 **New environment variables for session and agent control** NEW

80

Set `CLAUDE_CODE_DISABLE_AGENT_PROGRESS_SUMMARIES=1` to turn off subagent progress summaries and the requests that write them; set `CLAUDE_CODE_RESUME_HINT_COMMAND` to have the exit hint name a custom resume command for programs that start Claude Code; and support `CLAUDE_CODE_AUTO_MODE_MODEL` in managed settings for Claude Desktop sessions on third-party providers.

Silence subagent progress summaries in a headless pipeline to cut API requests and noise in `claude -p` output.

```
$ CLAUDE_CODE_DISABLE_AGENT_PROGRESS_SUMMARIES=1 claude -p 'Run the full test suite and summarise failures'
```

— Three named env vars with one runnable example v2.1.323

04 **Plugin install JSON and pane coverage detection fields** NEW

70

Read `unsetOptionCount` from the `claude plugin install --json` result to see how many `userConfig` options are not yet set, and read `isPromptCovered` in the props a mod's pane is drawn with to detect when a dialog or field above the prompt covers it.

— Named fields and command but no usage example v2.1.323

THINNER COVERAGE BELOW

05 **Progress heartbeat for long-running hooks** IMPROVED

55

Sends a progress heartbeat every 30 seconds during long-running hooks in `claude -p` and SDK sessions so hosts like Claude Desktop do not end the turn as hung.

— Explains mechanism and interval but no direct action v2.1.323

06 **Code Mode for JavaScript-based tool calls** NEW 45

An experimental mode where Claude writes JavaScript to call its tools instead of direct tool invocations; opt in or organizations can disable it by following the steps at `https://code.claude.com/docs/en/code-mode`.

— Names the opt-in doc but not the mechanism's limits v2.1.323

07 **Artifact uploads support .docx files** IMPROVED 30

Uploads `.docx` files as Artifact tool assets alongside previously supported formats.

— Names the format but no further detail v2.1.323

WAS THIS USEFUL? 👍 👎

swttch Feedback ⓘ v0.37.0 NOTES ↗ RANK

Swttch is a claude Code GUI plugin for JetBrains IDEs. (ex - Claude Code with GUI)

> Swttch's v0.37.0 release adds session management tooling, transcript/plan visibility in the panel, and direct file action controls for Claude-sent files in JetBrains IDEs.

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

01 **File action controls for Claude-sent files** NEW 50

Users can act on files Claude sends directly in the IDE — save, open a context menu, or multi-select files — via new file action controls.

— Describes concrete UI actions but no exact navigation path v0.37.0

02 **Session management via MCP and session tree** NEW 40

swttch adds a session management MCP, a session tree view, and Desktop-style sessions, letting users browse and switch between Claude sessions directly in the IDE.

— Names three additions but no concrete command or endpoint given v0.37.0

03 **Transcript summary line and plan card** NEW 25

swttch now shows a transcript summary line and plan card in the panel, matching the desktop app experience.

— Bare description of a UI addition with no mechanism detail v0.37.0

WAS THIS USEFUL? 👍 👎

keploy

Feedback ⓘ v3.7.2 NOTES ↗ RANK

keploy is an open-source platform for creating safe, isolated production sandboxes for API, integration, and E2E testing.

> Keploy expanded Docker daemon compatibility to Rancher Desktop, began recording non-streaming HTTP responses as test cases, and added deterministic replay of timers, clocks, and IDs for mock-based e2e tests.

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

01 **Non-streaming HTTP responses recorded as test cases** IMPROVED

50

Keploy now records non-streaming HTTP responses as test cases when `keploy record` is used, capturing flows that previously produced no test output.

— Names exact command and prior gap it fixes v3.7.2

02 **Timer, clock, and ID replay in mock-based e2e tests** NEW

50

Keploy's mock replay now handles timers, clocks, and IDs during e2e test playback, letting time-sensitive and ID-dependent tests run deterministically without infrastructure changes.

— Describes what and why, no concrete usage steps v3.7.1

03 **Rancher Desktop Docker context support** NEW

40

Keploy now reaches the Docker daemon through the Docker CLI's current context, adding support for Rancher Desktop alongside existing Docker Desktop workflows.

— Explains mechanism but no command example v3.7.2

WAS THIS USEFUL? 👍 👎

mvschwarz OpenRig

Feedback ⓘ v0.7.13 NOTES ↗ RANK

OpenRig is a network manager for AI coding agents: build persistent teams with roles, shared context and owned work.

> OpenRig's v0.7.13 release adds an experimental hub connector, roster check-in visibility, and automatic usage-limit aware parking for Claude seats, alongside several breaking changes to schema migrations, SSH host status checks, and package install behavior.

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

01 **Schema migrations require database backup** BREAKING

80

Upgrading to this release runs three schema migrations (160, 161, 162) on first daemon start, and reverting to 0.7.12 afterward is not established, so back up the database first. Use `rig config get db.path` to find the database file.

– Exact migration numbers and command to locate the database v0.7.13

02 **Check-in status in roster commands** IMPROVED

70

`rig roster show` and `rig roster find` now print a Check-in line for connected assistants, driven by `check_in_minutes` or `check_in_on_request: true` set in a roster entry.

Inspect check-in status for all connected (hosted) roster members to see which are overdue or on-request.

```
$ rig roster show
```

– Exact commands and config keys with a usage example v0.7.13

03 **`rig package install` guidance file handling** IMPROVED

70

`rig package install` now leaves a guidance file alone when a recognized managed block has no complete marker pair, listing it under 'Left as it is:' and installing the rest, instead of appending as before.

– Clear before/after behavior but no runnable example v0.7.13

04 **`rig network` hub connector** NEW

67

Connect a local seat to an external hub with the new `rig network` command, marked experimental and built on the MCP SDK already bundled since 0.7.12.

— Names command and dependency but little on hub mechanics v0.7.13

05 **Passive SSH host status checks** BREAKING 65

Passive SSH host status now reads `unknown` — routine host listings no longer open a TCP connection to SSH hosts. Use `rig host doctor` to check actively.

— Named replacement command but limited mechanism detail v0.7.13

06 **Manual permission decision fallback** NEW 60

`rig seat decide-permission` lets you pass a permission decision manually when the automatic confirmation note reports it could not be delivered.

— Named command but no detail on decision format or options v0.7.13

THINNER COVERAGE BELOW

07 **Setup agent prompts person registration** NEW 50

Running `rig daemon start` now triggers the setup agent to offer to register you as the team's person when the person registry is empty.

— Describes trigger but not the registration flow itself v0.7.13

08 **Usage-limit aware parking for Claude seats** BREAKING 50

A usage-limited Claude seat now waits for its stated reset time automatically; it should not be released by hand on a timer as before.

— Behavior change stated but no mechanism for reset detection v0.7.13

09 **Handover retry caution after failure** BREAKING 40

A failed handover may have left its successor running; check the pane before retrying instead of assuming the successor is gone.

— Only a caution note, no command or mechanism given v0.7.13

└──▷ BREAKING ON UPGRADE

! Back up the database before upgrading: three schema migrations (160, 161, 162) run on first daemon start and reverting to 0.7.12 afterward is not established. Use `rig config get db.path` to find the database file.

! A usage-limited Claude seat now waits for its stated reset time automatically — do not release it by hand on a timer.

- ! A failed handover may have left its successor running; check the pane before retrying instead of assuming the successor is gone.
- ! `rig package install` now leaves a guidance file alone when a recognized managed block has no complete marker pair, listing it under 'Left as it is:' and installing the rest, instead of appending.
- ! Passive SSH host status now reads `unknown` — routine host listings no longer open a TCP connection to SSH hosts. Use `rig host doctor` to check actively.

WAS THIS USEFUL?  

zot

Feedback ⓘ v0.4.76 NOTES ↗ RANK

zot manages coding agent workflows from the command line.

> zot added fail-closed request transforms for extensions, changing transform failures to block requests instead of allowing unchecked pass-through.

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

01 **Fail-closed extension transforms** IMPROVED 54

Extensions now support fail-closed request transforms: when a transform fails, zot blocks the request rather than passing it through unchecked.

— Behavior described; no config surface v0.4.76

NEW INTEGRATION UberLLM provider

v0.4.75 NOTES ↗ STABLE

└──▷ **GET THIS VERSION** ▾

Go

```
$ go install github.com/patriceckhart/zot/cmd/zot@v0.4.75
```

git

```
$ git clone --branch v0.4.75
https://github.com/patriceckhart/zot.git
# already have the repo? check out this version:
$ git checkout v0.4.75
```

Go module path matches the repo · checked 13 Nov Wrong command?

WAS THIS USEFUL? 👍 👎

agentty

Feedback ⓘ v0.9.54 NOTES ↗ RANK

Agentty is an AI pair programming in your terminal — one static binary, sub-ms startup, any model

> Agentty focused this release on terminal review ergonomics: word-level diff review with richer file/change context, a live model-thinking indicator, and a startup hint for disabling sandboxing with `--sandbox off`.

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

01 **Sandbox disable hint** IMPROVED 82

The startup line now surfaces the `--sandbox off` flag to disable the sandbox entirely, printed as: `agentty: sandbox: active (claybin) · turn it off entirely with --sandbox off`.

— Exact flag and startup text v0.9.54

02 **Word-level review diffs** IMPROVED 74

The review pane (`^R`) now highlights only the changed words on an edited line instead of the whole line. It also labels off-screen changes with the function name, marks new and deleted files, expands the code window to the full terminal instead of 24 rows, and shows green/red `+N −N` pill counters in file tabs.

— Names shortcut, counters, and review behavior v0.9.54

03 **Live reader thinking status** IMPROVED 67

The `^U` reader shows a live `Thinking ⁖` indicator with a `● LIVE` badge and a per-second timer while the model reasons. It then changes to `Done thinking` in green and finally `Reasoned` with the elapsed time.

— Clear UI states and shortcut v0.9.54

WAS THIS USEFUL? 👍 👎

Amazon Kiro

Feedback ⓘ

Mobile: Kiro Mobile for Android (2026-11-12) BLOG ↗ RANK

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

> Kiro shipped an Android app for managing cloud sessions from a phone, plus IDE memory across chat sessions and CloudTrail auditing for Kiro service MCP tool calls. The window also added CLI copy and compaction controls, workflow-following UI, post-compaction skill reloads, MCP sign-in support, Mermaid diagram readability improvements, and Approve for me command-palette toggles.

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

01 **Configurable CLI auto-compaction threshold** NEW 88

The Kiro CLI now lets you set the context usage percentage where automatic conversation-history compaction begins by running `/settings` and opening the Compaction screen. The value is saved as `compaction.autoCompactionThreshold` in `cli.json`, and a workspace-level `cli.json` can override it.

— Includes command, screen, key, and override file

CLI: Copy Earlier Responses and Choose …

02 **Android cloud-session app** NEW 81

Kiro Mobile is now available on Google Play for starting, following, and steering Kiro cloud sessions from an Android phone without keeping a laptop on. It can pick up sessions started in Kiro Web, the IDE, or the CLI with connected repositories and model choices already present; supports Build with spec, Autonomous mode where the model selector shows Auto and cannot be changed, phone-sized diff cards for edited files, pull request status such as merged or open, image attachments, inline answers to agent questions, and live replies and tool activity.

Pick up an in-progress cloud session on your Android phone after starting it in the IDE or CLI, without leaving your laptop running.

📍 1. Install Kiro Mobile from Google Play. 2. Sign in with your Kiro account (Pro, Pro+, Pro Max, or Power plan required). 3. Open the session list — sessions started in Kiro Web, the IDE, or the CLI appear automatically with their repositories and model choices. 4. Tap a session to follow live replies and tool activity, attach images, or answer the agent's questions.

— Concrete mobile surfaces and setup steps Mobile: Kiro Mobile for Android (2026-1…

03 **IDE memory across chats** NEW 81

Kiro IDE adds a Memory view, accessible from the Memories icon in the activity bar, for reviewing what the agent remembers across sessions, including repository conventions, corrections, and approaches already ruled out. The `Kiro Agent: Memory` setting controls how the agent reads and writes memories, including Use only mode, and chat now shows when Kiro accesses or changes a memory during a session.

Prevent the agent from adding new memories while still benefiting from existing ones — useful when you want stable, curated context in a shared or sensitive project.

📍 In Kiro IDE, open Settings and set `Kiro Agent: Memory` to Use only. The agent will read existing memories but will not create or modify any during the session.

— Names UI entry point and setting IDE 1.4.178: Memory Across Chat Session…

04 **CloudTrail MCP tool-call auditing** NEW 79

Kiro now records calls to Kiro service MCP tools as AWS CloudTrail management events by default for profiles in `us-east-1` and `eu-central-1`. Events capture who made the request, which tool was called, when it ran, and whether it completed; administrators can also log tool listings and read-only tool calls by enabling CloudTrail data events.

— Names event type, regions, and audit contents `General: Audit Kiro Tool Calls with AWS…`

05 **Approve for me palette commands** NEW 79

Kiro IDE adds Command Palette commands `Kiro: Approve For Me (User)...` and `Kiro: Approve For Me (Workspace)...`. They show the current state of Approve for me for new chats and let you toggle it on or off at user or workspace scope.

Toggle Approve for me on or off per scope (user or workspace) without leaving the IDE.

> 📍 Open the Command Palette and run 'Kiro: Approve For Me (User)...' to see and change the setting for new chats at user scope, or 'Kiro: Approve For Me (Workspace)...' for workspace scope.

— Exact command names and scopes `IDE: Follow Workflow Runs, Skills After…`

06 **CLI response copy commands** NEW 77

The Kiro CLI can now copy an earlier response with `/copy N`, where `N` is the Nth-latest response, or copy the last several messages in order with `/copy last N`.

Copy the response from two turns ago without scrolling back through the conversation.

```
$ /copy 2
```

— Runnable slash commands included `CLI: Copy Earlier Responses and Choose …`

07 **Workflow run following controls** NEW 76

In the Workflow tab, Kiro now lets you follow workflow runs step by step: Go to active selects the oldest step that is running, paused, or waiting, and Follow keeps the active step selected as each step finishes. You can also open the chat that launched a workflow run by selecting the chat name after `Created by:` in the Workflow tab.

Track a long-running multi-step workflow in real time without manually clicking through steps as they complete.

> 📍 In the Workflow tab, click Go to active to jump to the oldest in-progress step, then enable Follow so the view advances automatically as each step finishes.

— Exact tab controls and behavior IDE: Follow Workflow Runs, Skills After…

08 **MCP sign-in without oauth block** IMPROVED 63

MCP sign-in now works without an `oauth` block when the authorization server supports Client ID Metadata Documents, offers no dynamic client registration, and accepts any port on a localhost callback. In that case, Kiro signs in with its own hosted document.

— Conditions named, but no config location IDE: Follow Workflow Runs, Skills After…

09 **Post-compaction skill restoration** IMPROVED 60

After Kiro compacts a conversation, the agent now reloads skills and steering files that the summary dropped. It also loads a large skill or steering file in full when it fits instead of using a cut-off preview, and skips re-adding a skill already present in the conversation.

— Mechanism described, no user control named IDE: Follow Workflow Runs, Skills After…

THINNER COVERAGE BELOW

10 **Sub-agent task visibility** IMPROVED 58

The Kiro CLI now shows each sub-agent's assigned task in the inline sub-agent panel and in the Ctrl+G agent monitor. The agent monitor also marks the sub-agent's final response.

— Names monitor shortcut and displayed state CLI: Copy Earlier Responses and Choose …

11 **Readable Mermaid diagrams in chat** IMPROVED 49

Mermaid diagrams in chat now stop shrinking at 80% of their natural size so labels stay readable. Larger diagrams support drag-to-pan, and the zoom toolbar is easier to read on hover.

— Behavior and threshold named IDE: Follow Workflow Runs, Skills After…

NEW INTEGRATION Okta identity provider support for credit pooling

> General: Credit pooling now works with Okta and Microsoft Entra ID (2026-11-10)
>
> BLOG ↗ STABLE
>
> Kiro adds Okta and Microsoft Entra ID support for enterprise credit pooling, with a Switch to credit sharing wizard and default pooling for new IdP-connected profiles.
>
> Share Kiro credits across a team when using Okta or Microsoft Entra ID as an identity provider, via Access management > Credit sharing in the Kiro console — in addition to the existing AWS IAM Identity Center support.
>
> Move existing individual subscribers to a shared pool using the Switch to credit

> sharing wizard in the Kiro console.
>
> New profiles connected to IAM Identity Center, Okta, or Microsoft Entra ID now start with credit pooling enabled by default.

WAS THIS USEFUL? 👍 👎

Charm Crush

Feedback ⓘ v0.101.0 NOTES ↗ CODE ↗ RANK

Crush is a terminal-based AI coding assistant that integrates LLMs, LSPs, and MCP servers into your development workflow.

> Crush's latest release adds a `CRUSH_DATA_DIR` environment variable for project-specific data storage, lets users toggle skills on or off from the command palette with clearer failure reporting, and introduces per-tool elapsed timers alongside Brotli-compressed requests to Hyper.

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

01 CRUSH_DATA_DIR environment variable NEW

75

Override the project data directory at runtime with the `CRUSH_DATA_DIR` environment variable, for example `CRUSH_DATA_DIR=/path/to/project/.crush crush`, useful when running multiple projects or in CI environments.

Isolate Crush's data storage to a project-specific path, useful when running multiple projects or in CI environments.

```
$ CRUSH_DATA_DIR=/path/to/project/.crush crush
```

— Exact env var with runnable command example v0.101.0

THINNER COVERAGE BELOW

02 Skill toggles in command palette NEW

50

Toggle individual skills on or off directly from the command palette; skills that have broken now display a red × marker along with the reason they failed.

— Names the UI surface but no exact command v0.101.0

03 Elapsed timers on tool calls NEW

35

Every tool call now displays an elapsed timer so you can tell whether a long-running task is slow or stuck.

— Describes behavior with no config or command to act on v0.101.0

04 Brotli compression for Hyper requests IMPROVED

35

Requests to Hyper are now compressed with Brotli, with Gzip used as a fallback for other agents.

— Names the compression mechanism but offers no reader action v0.101.0

└──▷ **ALSO FROM THESE RELEASES**

:
:

WAS THIS USEFUL?  

devoxxgenieideaplugin

Feedback ⓘ v1.18.0 NOTES ↗

RANK

Devoxxgenieideaplugin is a devoxxgenie is an agentic plugin for IntelliJ IDEA that uses local LLM's (Ollama, LMStudio, GPT4All, Jan and Llama.cpp) and Cloud based LLMs to help review, test, explain your project code. Latest version now also supports Spec Driven Development with CLI Runners.

> DevoxxGenie's v1.18.0 release consolidates LLM routing and local-provider handling into unified settings frameworks, adding OrcaRouter and UberLLM router support and a generic OpenAI-compatible local provider spanning Rapid-MLX, GPT4All, llama.cpp and Jan.

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

01 Unified Routers settings tab with OrcaRouter and UberLLM NEW

95

A new Routers settings tab unifies LLM routers behind `RouterDefinition`, `RouterRegistry` and `RouterChatModelFactory`, migrating OpenRouter, Requesty and DemonRoute onto it. New router entries are added for OrcaRouter (`https://api.orcarouter.ai/v1`, filtering its catalog to models whose `endpoint_capabilities` include `chat_completions`) and UberLLM (`https://api.uberllm.dev/v1`, reading display names, context windows and prices from its `/models` catalog and filtering out non-text-output models). Each router's base URL is now editable — a blank field falls back to its default, and changing it triggers a model-list refresh — enabling alternative endpoints such as Requesty's EU router.

Enable OrcaRouter for chat so prompt requests route across its catalog of chat-capable models.

> 📍 In IntelliJ, go to Settings › DevoxxGenie › Routers, check the OrcaRouter checkbox, paste your API key, and click Apply. OrcaRouter (endpoint: `https://api.orcarouter.ai/v1`) now appears in the chat provider list.

Point Requesty at its EU endpoint to keep prompts in-region, then refresh its model list.

> 📍 In Settings › DevoxxGenie › Routers, select Requesty, clear the base URL field and enter your EU endpoint URL, then click away — DevoxxGenie refreshes the model list automatically and uses the new URL for all catalog and chat requests.

— Names classes, two router endpoints, and exact UI steps v1.18.0

02 Generic OpenAI-compatible local provider framework NEW

90

Local OpenAI-compatible servers now share a generic provider built on `LocalProviderRegistry`, `LocalProviderDefinition` and `OpenAiCompatibleChatModelFactory`, consolidating Nativ, Rapid-MLX, GPT4All, llama.cpp and Jan. Rapid-MLX is now available as a local provider for Apple Silicon, defaulting to `http://localhost:8000/v1/` with context read from `max_model_len` (off by default). GPT4All now reads a context window (8,000 tokens by default, with an optional fallback setting). llama.cpp now lists the models `llama-server` actually serves with real context windows read from `meta.n_ctx` then `meta.n_ctx_train`, plus streaming chat and 'add project to context' support. Jan runs on this provider with a compact-JSON request workaround for older Jan versions carried over as a definition flag, tested against Jan 0.8.8.

Use Rapid-MLX on an Apple Silicon Mac to route local inference through the MLX server.

> 📍 In Settings › DevoxxGenie › Local, enable Rapid-MLX (defaults to `http://localhost:8000/v1/`), then start your Rapid-MLX server. DevoxxGenie reads available models and their context windows from `max_model_len` automatically.

— Names classes, defaults and config fields per provider v1.18.0

THINNER COVERAGE BELOW

03 **Expandable error details in chat error card** IMPROVED

50

The chat error card now shows full error text with selectable text, a Show full error / Show less toggle for errors over 300 characters, and a Copy button alongside Retry.

— Describes UI behavior but no config or API surface v1.18.0

04 **Exo moves to new Sharding settings tab** IMPROVED

35

Exo configuration moves to a new Sharding tab in settings, alongside planned distributed-inference options shown as Coming soon.

— Thin description, future options unspecified v1.18.0

WAS THIS USEFUL? 👍 👎

agents-cli

Feedback ⓘ v1.14.0 NOTES ↗ RANK

The CLI and skills that turn any coding assistant into an expert at creating, evaluating, and deploying AI agents on Google Cloud.

> agents-cli v1.14.0 adds pre-flight checks before deploys, flags to remove env vars and secrets on deploy, PyPI-based extension installs, manifest scaffolding for existing ADK projects, and Claude agent memory via Vertex AI Memory Bank, alongside a breaking default binding change for the Go playground server.

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

01 Pre-flight checks before agent deploys

BREAKING 91

`agents-cli deploy` now runs pre-flight checks before source builds, catching unreachable package indexes, missing or stale locks, ignored build inputs, out-of-tree path dependencies, and Python version mismatches; skip them with `--skip-preflight`. As a result, `deploy --dry-run` now exits 1 on pre-flight errors, so CI jobs or scripts that run it may need updating.

Deploy to Cloud Run while catching build-breaking issues early, without uploading anything.

```
$ agents-cli deploy --dry-run
```

— Lists exact checks and flags, includes a runnable dry-run example v1.14.0

02 Claude agent memory via Vertex AI Memory Bank

NEW 91

`agents-cli scaffold create -a claude --memory-bank` scaffolds a Claude agent with long-term memory in Vertex AI Memory Bank: it recalls memories at turn start, learns after each turn, and exposes a `search_memory` tool. Setting `OTEL_INSTRUMENTATION_GENAI_CAPTURE_MESSAGE_CONTENT` additionally records recalled memories in the system prompt as `gen_ai.system_instructions` on the turn span.

Scaffold a Claude agent with Vertex AI Memory Bank so it recalls user context across conversations.

```
$ agents-cli scaffold create -a claude --memory-bank my-memory-agent
```

— Describes recall/learn mechanism and gives exact scaffold command v1.14.0

03 **Env var and secret removal on deploy** BREAKING 86

`agents-cli deploy` gains `--remove-env-vars KEY,...`, `--clear-env-vars`, `--remove-secrets KEY,...`, and `--clear-secrets` for Cloud Run and Agent Runtime deployments. Note that `--clear-secrets` removes secrets set outside agents-cli (such as by Terraform) and secrets mounted as volumes, not only those managed by agents-cli.

Remove specific secrets from a deployed agent that are no longer needed, without touching ones managed outside agents-cli.

```
$ agents-cli deploy --remove-secrets DB_PASSWORD,OLD_API_KEY
```

— Names every flag and the deployment targets, with a runnable example v1.14.0

04 **Playground host binding for Go projects** BREAKING 86

For Go projects on ADK Go v2.5.0 or later, `agents-cli playground` now honors `--host` and binds `127.0.0.1` by default instead of all interfaces; pass `--host 0.0.0.0` to reach it from another machine. When running `agents-cli scaffold upgrade` on a Go project with a custom Dockerfile, add `"--host", "0.0.0.0"` after `"web"` in its `ENTRYPOINT` or the container will receive no traffic.

— Gives exact migration step for custom Dockerfiles v1.14.0

05 **Install CLI extensions from PyPI** NEW 83

`agents-cli extension add pypi@<package>[==<version>][#<folder>]` installs extensions from PyPI or a custom index set via `UV_DEFAULT_INDEX`, `UV_INDEX_URL`, or `PIP_INDEX_URL`, pinning to the wheel's sha256. Move the pin later with `agents-cli extension update <name> --to <version>` or `--to :latest`.

— Exact command syntax and env vars given, no usage example v1.14.0

06 **Manifest generation for existing ADK projects** NEW 75

`agents-cli scaffold enhance --from-adk-project` generates a minimal `agents-cli-manifest.yaml` for an existing ADK project without rendering a full template; preview the result with `--dry-run` or overwrite an existing manifest with `--force`.

— Flags and generated file named, no example given v1.14.0

07 **Opt-out for Claude Agent SDK usage telemetry** NEW 65

Claude Agent SDK model calls now send an `x-goog-api-client: agents-cli-app/<version>+claude-agent-sdk` usage header; opt out by setting `DO_NOT_TRACK=1` or `AGENTS_CLI_TELEMETRY=0`.

— Names header and opt-out vars but little further detail v1.14.0

└──▷ **BREAKING ON UPGRADE**

! `deploy --dry-run` now exits 1 on pre-flight errors, so CI jobs or scripts that run it may need an update.

! `--clear-secrets` removes secrets set outside agents-cli (such as by Terraform) and secrets mounted as volumes, not only those managed by agents-cli.

! For Go projects on ADK Go v2.5.0 or later, `agents-cli playground` now binds `127.0.0.1` by default instead of all interfaces; when running `agents-cli scaffold upgrade` on a Go project with a custom Dockerfile, add `"--host", "0.0.0.0"` after `"web"` in its `ENTRYPOINT` or the container will receive no traffic.

WAS THIS USEFUL?  

octomind

Feedback ⓘ 0.71.0 NOTES ↗ RANK

Octomind is an open-source AI coding agent and agent runtime: one binary, any model, MCP-native. Runs in terminal, CI, or as a daemon.

> Octomind 0.71.0 adds automatic dependency expansion in prompts and post-turn session summaries, while removing the ACP hook option and global layer configuration in favor of per-agent settings.

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

01 **Global layer config removed from `config`** BREAKING 65

Global layer configuration has been removed from `config`; per-agent or per-role layer settings must be used instead.

– Names exact config surface and required migration path 0.71.0

THINNER COVERAGE BELOW

02 **ACP hook option removed** BREAKING 50

The ACP hook option has been removed; any configuration referencing it will be rejected.

– Names exact removed option and rejection behaviour 0.71.0

03 **Automatic dependency expansion in prompts** NEW 40

Prompts now expand declared dependencies automatically, surfacing the full dependency graph before the model sees the request.

– Describes behaviour but no config key or flag named 0.71.0

04 **Turn summaries after successful sessions** NEW 35

Sessions now print a summary after each successful turn, giving a quick audit trail of what the agent completed.

– Clear behaviour but no concrete format or command given 0.71.0

05 **Cross-domain agent and tap reference resolution** IMPROVED 15

Octomind now resolves agent and tap references across domains.

– Only a one-line summary mention, no mechanism detailed 0.71.0

└──▷ **BREAKING ON UPGRADE**

! The ACP hook option has been removed; any configuration referencing it will be rejected.

! Global layer configuration has been removed from `config`; per-agent or per-role layer settings must be used instead.

WAS THIS USEFUL? 👍 👎

kane-cli

Feedback ⓘ 0.10.4 NOTES ↗ RANK

Kane CLI by TestMu AI (Formerly LambdaTest)

> kane-cli's 0.10.4 release adds structured lifecycle and pipeline observability, a `@manual` step marker for replay-only steps, CI-friendly exit codes, and self-healing replay, alongside always-on telemetry with no opt-out.

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

01 **Lifecycle events and pipeline observability** NEW 83

Every command — `run`, `testmd run` and `explore` — now emits start and end lifecycle events, making a run's shape visible even when it dies early. Session start records the `surface` and `target` while session end records the `outcome`, including sessions that exit without finishing; each pipeline stage now reports how long it took, making slow steps easy to spot; and a test run coordinator is now its own session, with each member run named and tagged by `run type` so a parallel suite reads as one tree.

— Names concrete fields and session structure but no new flag 0.10.4

02 **`@manual` step marker for replay-only steps** NEW 75

Add `@manual` to tag test steps that need a human; replay treats them as replay-only instead of attempting automation.

Tag a step that requires human interaction so replay skips automating it and treats it as replay-only.

text

```
# In a _test.md file:
# @manual
# Step: Physically confirm the hardware token LED is blinking
```

— Named marker plus a usage example 0.10.4

03 **Exit codes distinguish test vs CLI failures** IMPROVED 70

Exit codes now distinguish test failures from CLI failures so scripts and CI can tell them apart without parsing output; an unauthenticated command now reports the missing login explicitly instead of a generic failure; and an empty test result now exits non-zero instead of silently passing.

— Behavior change described but no exact codes given 0.10.4

THINNER COVERAGE BELOW

04 **Remote test run dispatch reporting** IMPROVED 55

`testrun --remote` now reports dispatch, outcome and failures, with every HyperExecute VM session naming the job that sent it.

— Named flag and integration, but description stays high-level 0.10.4

05 **Debug output held back until failure** IMPROVED 45

Debug lines are now held back and released only when a run fails, and a passing `test.md` run drops its held step output.

— Clear mechanism, no config surface to act on 0.10.4

06 **`query_js` supports function-evaluating snippets** IMPROVED 40

`query_js` now executes snippets that evaluate to a function instead of erroring.

— Named command, thin description of change 0.10.4

07 **Self-healing replay re-reads pages after actions** IMPROVED 40

Pages are now re-read after hover and drag actions so replay works from current content rather than stale state.

— Explains behavior but no named surface 0.10.4

08 **Telemetry always on, no opt-out** BREAKING 40

Telemetry is always on with no opt-out; usage before login is attributed by machine id.

— Clear breaking change, no mitigation path given 0.10.4

└──▷ **BREAKING ON UPGRADE**

! Telemetry is always on with no opt-out; there is no flag or setting to disable it.

WAS THIS USEFUL?  

opencode-telegram-bot

Feedback ⓘ v0.27.1 NOTES ↗

RANK

opencode-telegram-bot is a Telegram client for OpenCode that lets you run and monitor AI coding tasks from your phone while they run locally.

> opencode-telegram-bot's v0.27.1 release improves session tracking in Telegram, adding unread-reply indicators and clearer context in notifications.

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

01 **Unread-reply indicator in /recent** IMPROVED 50

The `/recent` command now marks idle sessions with an indicator when their latest reply has not yet been delivered to chat, clearing it once the reply is shown.

— Names exact command but no further mechanism detail v0.27.1

02 **Richer context in background reply notifications** IMPROVED 40

Background reply notifications now include the session title and project name, so you can identify the source without having to recognise a session ID.

— Clear behaviour change, no named config surface v0.27.1

WAS THIS USEFUL? 👍 👎

hax

Feedback ⓘ v0.7.0 NOTES ↗ RANK

hax is a minimalist, terminal-native coding agent written in C.

> hax v0.7.0 adds offline man pages, changes the default llama.cpp provider port, and improves conversation compaction to recover from context-limit overflows.

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

01 **llama.cpp provider default port change** BREAKING 85

The llama.cpp provider now defaults to port 9931 instead of the previous 8080. Running an llama.cpp server older than v0.7.0 requires setting `HAX_LLAMACPP_PORT=8080` or `providers.llamacpp.port` to restore the old port.

Point hax at an older llama-server that still listens on port 8080 instead of the new default 9931.

```
$ HAX_LLAMACPP_PORT=8080 hax --provider llama.cpp
```

— Exact env var, config key, and old/new port values given v0.7.0

02 **Offline man pages for hax** NEW 80

Offline documentation is now available via `man hax` for usage, hax-config(5) for configuration, hax-sessions(5) for sessions, and hax-providers(7) for providers, shipped from both packages and source builds.

Read the man page for hax configuration options without an internet connection.

```
$ man hax-config
```

— Names exact man page sections and a runnable command v0.7.0

03 **Compaction recovers overflowed conversations** IMPROVED 65

Compaction via `/compact` and automatic compaction can now recover conversations that already exceed the context limit, summarizing a shortened copy and sacrificing older detail first. Automatic compaction also accounts for new tool output and the next message, leaves room for the summary on small context windows, and suspends after repeated failures instead of retrying indefinitely.

— Explains mechanism and behaviour but no concrete config surface v0.7.0

└──▷ **BREAKING ON UPGRADE**

! The llama.cpp provider now defaults to port 9931 instead of 8080; set `HAX_LLAMACPP_PORT=8080` or `providers.llamacpp.port` to restore the old port when using a server older than v0.7.0.

WAS THIS USEFUL?  

Sourcegraph Amp

Feedback ⓘ Bye Bye, Dial (2026-11-12) 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 replaces its Dial model-switching UI with a per-thread model selector that shows billing source, supports mid-conversation switching, and defaults new threads to a Staff Pick model.

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

01 Per-thread model selector with billing source NEW

84

A new model selector lets you pick the model and its effort level per thread, showing whether each model bills a connected subscription, API key, or Amp credits. You can switch models mid-thread by opening the selector during an active conversation; Amp records the change in the transcript (e.g. `Switched to GPT-6.1 Sol · High`) and applies the new model's system prompt and tools from the next message onward. Staff Pick (currently Claude Opus 5.5) is the default starting model, and `New Threads Start On` in `Settings → Model Selector` can be configured to start on Staff Pick, the last-used model, or a fixed model. Custom agents previously placed on the Dial via Build Dial now appear under `Other Models` in the selector.

— Names config path and behavior but no screenshots of switching flow

Bye Bye, Dial (2026-11-12)

02 Dial removed in favor of model selector BREAKING

65

The Dial and its four modes (low, medium, high, ultra) are removed; all threads now use the model selector instead. Dial settings do not carry over: custom model mappings set in Tune Modes and custom agents placed via Build Dial must be reconfigured in the new model selector, and new threads start on Staff Pick regardless of which Dial mode was previously used last, unless a default Dial mode had been explicitly configured.

— Clear migration note but no exact reconfiguration steps Bye Bye, Dial (2026-11-12)

└──▷ **BREAKING ON UPGRADE**

! The Dial and its four modes (low, medium, high, ultra) are removed; all threads now use the model selector instead.

! Dial settings do not carry over: custom model mappings set in Tune Modes and custom agents placed via Build Dial must be reconfigured in the new model selector.

! New threads start on Staff Pick regardless of which Dial mode was previously used last (unless a default Dial mode had been explicitly configured).

WAS THIS USEFUL? 👍 👎

codealta

Feedback ⓘ 4.3.0 NOTES ↗ RANK

Codealta is an agentic Development Environment

> CodeAlta 4.3 adds vault-backed secrets spanning fourteen providers that agents can reference but never see, plus committed monorepo sub-project maps, agent-driven website sign-in, and fleet-wide provider setup via altanode.

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

01 Vault-backed secrets referenced by agents NEW 90

Secrets can be stored in external vaults — 1Password, Bitwarden, Vaultwarden, KeePassXC, pass, gopass, system keychain, HashiCorp Vault, OpenBao, AWS Secrets Manager, Azure Key Vault, Google Secret Manager, Doppler, Infisical, or a custom command — and referenced in config as `${secret:NAME}`. Agents can inspect vault state with `alta credential` and `alta secret` but never receive the actual value.

Reference a vault-backed secret in a provider config so agents use the value without ever seeing it.

yaml

```
${secret:ANTHROPIC_API_KEY}
```

— Names every vault type, the syntax, and the inspection commands 4.3.0

02 Committed monorepo sub-project maps NEW 80

Teams can declare monorepo sub-projects by committing `.alta/projects.toml`, where each sub-project gets its own sessions, plans, tasks, skills, agents, MCP servers, and plugin inheritance from parent folders. The map can be written or updated by an agent via `alta project manifest`, or proposed and written automatically by the built-in `codealta-monorepo` skill through Set up with an agent….

Let an agent propose and commit a monorepo sub-project map for a large workspace instead of editing `.alta/projects.toml` by hand.

```
$ alta project manifest
```

— Names the config file, CLI command and built-in skill 4.3.0

03 Agent-driven website sign-in NEW 60

Agents can sign in to websites through the `fill_login` browser call, pulling credentials from a vault item; the agent receives only `signed_in`, never the password, and the fill is restricted to the exact sites configured under `Settings > Secrets`.

— Names the call, the return value and the restriction UI path 4.3.0

04 **Fleet provider sign-in and install via altanode** NEW

60

Claude Code can be signed into an `altanode` machine using Anthropic's own auth flow via `altanode providers sign-in`, and Claude Code, Codex CLI, and ACP registry agents can be installed on a fleet machine in one click from `Providers > Add a provider…` on that machine's page.

— Names the CLI command and named agents, but no mechanism detail 4.3.0

05 **Provider metadata and model list refresh** NEW 60

Provider metadata and model lists can be refreshed with `alta provider refresh` and `alta provider model refresh` after updating a provider's CLI.

Refresh a provider's model list after upgrading its CLI, so CodeAlta sees any newly available models.

```
$ alta provider refresh && alta provider model refresh
```

— Exact runnable commands given but mechanism unexplained 4.3.0

THINNER COVERAGE BELOW

06 **Private server hosting via Dev Tunnel or Tailscale** NEW

55

A CodeAlta Server can be hosted behind a private Dev Tunnel with no open router port, or published via Tailscale Serve, directly from the Create my server flow.

— Names both hosting options and the setup flow, not a command 4.3.0

07 **GitHub Copilot enterprise organization policy enforcement** NEW 55

A GitHub Copilot enterprise organization policy — covering sandbox, bypass permissions, and default mode — can be applied across every session automatically; locked settings then appear in `Settings > Permissions` and Sandbox.

— Names the policy fields and the locked-settings UI location 4.3.0

08 **Searchable agent provider list** IMPROVED 35

The `Agent` chip now has a search box for the provider list, surfacing favorites first and showing each provider's logo, key, and type.

— Thin UI description with no underlying mechanism `4.3.0`

WAS THIS USEFUL? 👍 👎

◆ Model Training & Fine-Tuning

DeepSpeed

Feedback ⓘ v0.20.0 NOTES ↗ RANK

DeepSpeed is a deep learning optimization library for distributed training and inference, providing ZeRO sharding, pipeline parallelism, and mixed-precision execution.

> DeepSpeed v0.20.0 ships a TileLang-based sparse attention module for GLM-5.2 MoE models, opt-in deterministic execution for reproducible runs, and a round of ZeRO, AutoTP and HybridEngine refinements covering gradient handling, offload transfers, and non-CUDA accelerator support.

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

01 **TileLang sparse attention for GLM-5.2** NEW 80

Adds `deepspeed.models.glm_moe_dsa`, a TileLang sparse attention module for GLM-5.2 MoE models, intended to reduce attention compute costs.

Use TileLang sparse attention for GLM-5.2 MoE models to reduce attention compute costs.

```python
import deepspeed.models.glm_moe_dsa
```

— Names module and model, includes runnable import example v0.20.0

02 **Vocabulary-parallel JSD via VocabTP** NEW 60

Vocabulary-parallel generalized Jensen-Shannon Divergence (JSD) is now available via Triton kernels under `VocabTP`.

— Names module and kernel backend, no usage shown v0.20.0

03 **ZeRO memory and gradient handling improvements** IMPROVED 60

ZeRO-1/2 now preserves gradients across request boundaries until the optimizer step, enabling multi-request accumulation, and ZeRO-3 now copies offloaded master weights back to the device through a pinned buffer, reducing host-device transfer overhead.

— Clear mechanism but no config surface named Mechanism and benefit stated, no tunable exposed

v0.20.0

THINNER COVERAGE BELOW

04 **HybridEngine graph capture and decode improvements** IMPROVED

55

The hybrid engine graph decode path now supports non-greedy sampling for rollout generation, and HybridEngine graph capture now works on non-CUDA accelerators, with an explicit accelerator opt-in required.

— States the capability without usage detail Names scope and requirement but not the opt-in flag

v0.20.0

05 **AutoTP support for frozen models** IMPROVED 50

AutoTP now supports frozen models and manages AutoTP mode per model, enabling inference-only use of tensor-parallel models.

— Explains mechanism but no concrete API named v0.20.0

06 **Transformers 5.20 cache query offsets in DeepSpeedStaticCache** IMPROVED 45

Transformers 5.20 cache query offsets are now supported in `DeepSpeedStaticCache`.

— Names the class and dependency version but little else v0.20.0

07 **CPU accelerator memory capped at cgroup limit** IMPROVED 40

CPU accelerator memory is now capped at the cgroup limit of the process.

— Behaviour stated plainly, no configuration detail v0.20.0

08 **Opt-in deterministic execution policy** NEW 25

A new opt-in deterministic execution policy enables reproducible training runs.

— No flag, config key, or mechanism given v0.20.0

WAS THIS USEFUL?  

training_hub

Feedback ⓘ v0.12.0 NOTES ↗ RANK

Training_hub is an algorithm-focused interface for common llm training, continual learning, and reinforcement learning techniques

> training_hub v0.12.0 adds a new `classifier_sft` trainer for classification-head fine-tuning and substantially improves the memory estimator's per-GPU accuracy, while introducing two breaking changes to estimator semantics and cache location.

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

01 Classifier fine-tuning trainer (classifier_sft) NEW

95

The new `classifier_sft` trainer (also exposed as `ClassifierSFTAlgorithm` and the `thub classifier-sft` CLI subcommand) fine-tunes a classification head on any HF-recognized backbone. The `task` parameter selects `sequence_classification` (default), `multiple_choice`, or `token_classification`; custom architectures load via `trust_remote_code`, `revision`, `model_class`, or `loss_fn`. Encoder backbones fully fine-tune while decoder backbones use LoRA via `use_peft`, and checkpoints include the label mapping and task for direct inference.

Fine-tune a DeBERTa encoder as a sequence classifier on a labelled dataset — the fastest path when your label set is fixed.

python

```python
from training_hub import classifier_sft

result = classifier_sft(
    model_path="microsoft/deberta-v3-base",
    data_path="./labels.jsonl",
    ckpt_output_dir="./classifier_out",
    task="sequence_classification",
    num_epochs=5,
    learning_rate=2e-5,
)
```

Fine-tune a decoder (e.g. Qwen3) as a token classifier using LoRA, loading a custom architecture from the Hub with revision pinning.

```python
from training_hub import classifier_sft

result = classifier_sft(
    model_path="org/custom-decoder",
    data_path="./ner_train.jsonl",
    ckpt_output_dir="./ner_out",
    task="token_classification",
    use_peft=True,
    trust_remote_code=True,
    revision="a1b2c3d",
    num_epochs=3,
    learning_rate=1e-4,
)
```

— Full API names, CLI form, and two runnable code examples v0.12.0

02 **Model storage cache relocated outside package** BREAKING

80

The memory estimator now caches model sizes in `$XDG_CACHE_HOME/training_hub/model_storage.csv` (default `~/.cache/training_hub/`) instead of inside the installed package; the path can be overridden via `training_hub.profiling.memory_estimator.MODEL_STORAGE_PATH`. Tooling that read or wrote the old in-package cache path will no longer find it there.

— Exact new path, default, and override variable given v0.12.0

03 **Per-GPU LoRA/QLoRA memory estimation accuracy** IMPROVED

75

The memory estimator now gives per-GPU LoRA/QLoRA estimates using transformers' real `device_map='balanced'` placement, counts LoRA parameters from actual linear layers (including fused MoE experts), and sizes QLoRA as bitsandbytes loads it. Mean error dropped from ~29% to ~5% on held-out runs against Gemma 4 31B, Muse Glimmer 30B, Nemotron 30B-A3B and Qwen3/3.5 on H100s.

— Concrete mechanism and error figures but no usage command shown v0.12.0

04 **FSDP gather overhead in memory estimator** NEW

70

The memory estimator for SFT/OSFT now adds FSDP gather overhead via the new `count_fsdp_gather` option, covering the largest FSDP unit gather, the backward-prefetched unit, and per-GPU runtime overhead.

— Names the exact flag but no example invocation v0.12.0

05 batch_size semantics changed to per-process micro-batch

70

BREAKING

In the memory estimator, `batch_size` is now interpreted as the per-process micro-batch (`micro_batch_size`) rather than a global batch divided by `num_gpus` ; existing multi-GPU estimation calls that relied on the old global-batch semantics will now produce different (and larger) per-GPU estimates.

— Describes exact semantic change but no migration example v0.12.0

└──▷ **BREAKING ON UPGRADE**

! In the memory estimator, `batch_size` is now interpreted as the per-process micro-batch (`micro_batch_size`), not a global batch divided by `num_gpus` — existing multi-GPU estimation calls that relied on the old global-batch semantics will now produce different (and larger) per-GPU estimates.

! The estimator model-size cache has moved from inside the installed package to `$XDG_CACHE_HOME/training_hub/model_storage.csv` (default `~/.cache/training_hub/`); any tooling that read or wrote the old in-package cache path will no longer find it there.

WAS THIS USEFUL? 👍 👎

fizgig

Feedback ⓘ v9.7.0 NOTES ↗ RANK

Fizgig — LoRA & Fine-tune Studio for Klein 9B, Krea 2, MiniMax H3, Qwen Image 2.1, Anima & Z-Image Turbo and Ace-Step XL: train, fine-tune, profile, repair and extract LoRAs & LoKR for Image, Video & Music Models

> Fizgig v9.7.0 switches likeness scoring everywhere to AdaFace ViT, adds a new Automagic v4 optimizer with a configurable learning-rate ceiling, and reworks the MiniMax H3 Image Standard preset's optimizer, schedule and training resolution.

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

01 Automagic v4 optimizer with learning-rate ceiling control

NEW 88

Fizgig adds Automagic v4, starting at 2.5e-4, as a new optimizer that beat Automagic v3 and Muon in likeness tests, most of all on datasets with few headshots. Its per-weight learning rate ceiling can be capped by setting `max_lr=` in Optimizer Args.

Tighten Automagic v4's per-weight learning rate ceiling below the default 2.5e-4 when you want more conservative updates on a sensitive dataset.

env

```
max_lr=1.5e-4
```

— Names optimizer, default value, config flag and runnable example v9.7.0

THINNER COVERAGE BELOW

02 'Your photos' likeness band in Samples gallery NEW

50

The Samples gallery's likeness chart now shows a green band spanning the middle half of the training photos' own likeness scores, with their median marked as a dashed line and a plain-language summary below, giving a realistic ceiling for what the best training epoch can achieve.

— Describes mechanism but it's view-only, no action v9.7.0

03 Element boost megapixel threshold BREAKING

50

Element boost now requires Target Megapixels of 0.75 or more; below that threshold the tick greys out with an explanation, and MOSS-Image's 0.25 MP presets (Fast and Turbo Train) no longer enable it.

— Names exact threshold and affected presets, UI-level only v9.7.0

04 **Smaller face-model downloads** IMPROVED 50

Fizgig now fetches only InsightFace's face finder (18 MB) plus AdaFace ViT (460 MB) instead of InsightFace's full 300 MB pack; the updater pre-fetches them along with Make masks' person segmenter and tokenizer files.

— Exact sizes named but change is automatic, not user-driven v9.7.0

05 **AdaFace ViT replaces ArcFace for likeness scoring** IMPROVED 40

Likeness scoring now uses AdaFace ViT instead of ArcFace in every tool, and existing ArcFace scores are automatically re-scored with AdaFace the first time you open them.

— Clear before/after but no user action required v9.7.0

06 **Samples gallery scoring scoped to current run** IMPROVED 35

The Samples gallery now scores only the run currently being viewed rather than every run's previews in the samples folder, and new previews are scored immediately as they appear.

— States the behavior change but offers no action to take v9.7.0

WAS THIS USEFUL? 👍 👎

◆ Coding Agent Add-ons

Diagram Design

Feedback ⓘ changes since 2026-10-20 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 adds PlantUML import with configurable size and detail, five new diagram types, and a live gallery browsing all 50 diagram variants.

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

01 **PlantUML import into design system** NEW 90

Import PlantUML diagrams and redraw them in the design system using `/diagram-design:import-plantuml`, accepting `.puml`, `.plantuml`, `.pu` files and fenced `plantuml` or `puml` blocks in Markdown, for sequence and class diagrams. The `--size` and `--detail` flags control output size and level of detail, e.g. `--size=slide-16x9 --detail=simplified`.

— Named command, file types and flags with example invocation product docs

02 **Five new diagram types added** NEW 60

Diagram Design adds five diagram types: SBGN (biological species/process maps with compartments and reaction arcs), Requirement Traceability (requirements to design to verification), Event Modeling (commands to events to read models), Steps (numbered ordered stages), and Unit Grid (equal discrete units).

— Names each type and purpose but no commands to invoke them product docs

03 **Live gallery of all diagram types** NEW 60

Browse all 50 diagram types with light, dark, and full-editorial variants in a live gallery at `cathrynlavery.github.io`/diagram-design, or locally at `skills/diagram-design/assets/index.html`.

— Gives exact URL and local file path to browse product docs

THINNER COVERAGE BELOW

04 **Update commands for Git install and Pi extension** IMPROVED 55

Update the standalone Git install by running `npx skills update diagram-design`; update the Pi extension by running `pi update --extensions`.

— Exact runnable commands, minimal explanation of change product docs

changes since 2026-10-20 CODE ↗

Diagram Design adds a box plot variant, an arc diagram variant for skip-span imports, and a geometry verifier bundled inside the skill package

Diagram Design adds a box plot variant for five-number summaries to its scatter chart type.

Diagram Design adds an arc diagram variant to its dependency chart type for visualising skip-span imports.

Diagram Design now ships the geometry verifier inside the skill package itself.

Diagram Design's self-check now flags boxed text that overflows the width budget.

WAS THIS USEFUL? 👍 👎

◆ Image & Video Generation

Runway

Feedback ⓘ updated · Nov 13, 2026 NOTES ↗ RANK

Runway is an AI video and image generation platform with models for text-to-video, image-to-video and editing, available in a web app and through an API.

> Runway introduced App Studio, a plain-language builder that creates, brands and hosts custom creative apps powered by Runway's models.

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

01 **App Studio for building creative apps** NEW 50

App Studio lets users describe a creative app in plain language, including their brand and files, and Runway builds it using Runway's models and hosts it. In the web app, this is accessed by going to App Studio, typing a description of the app (with brand assets and files), and selecting Build.

Create and host a branded creative app by describing what you need in plain language inside App Studio.

> 📍 In Runway, go to App Studio, type a plain-language description of the creative app you need (including your brand assets and files), then select Build. Runway generates and hosts the app for you.

— Clear UI steps given but no technical mechanism detailed snapshot-20261113

WAS THIS USEFUL? 👍 👎

AGENT

◆ Web Search & Scraping for Agents

Perplexity API

Feedback ⓘ updated · Nov 13, 2026 NOTES ↗

RANK

Perplexity API provides language models with real-time web search and retrieval for building applications that research and answer questions.

> Perplexity API shipped new v1.3 decision models that can answer multiple questions in one pass and introduced a `locate` question type for span-level extraction, while shifting default aliases to the new version.

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

01 `locate` question type for span extraction NEW 72

The new `locate` question type, available only on v1.3 models, returns character-offset spans and a confidence score for every span of a string state matching a given description. It is not compatible with web grounding.

Find every span in a contract clause that matches a description of a liability limitation, with character offsets and scores, using the new locate question type.

```
$ curl https://api.perplexity.ai/decisions \
  -H 'Authorization: Bearer <token>' \
  -H 'Content-Type: application/json' \
  -d '{
    "model": "pplx-decider-v1.3",
    "state": "Supplier liability shall not exceed the fees paid
in the prior 12 months under any circumstances.",
    "questions": [
      {"type": "locate", "description": "liability cap language"}
    ]
  }'
```

— Names mechanism and constraint, includes runnable example snapshot-20261113

02 pplx-decider-v1.3 and flash-v1.3 models NEW 67

New decision models `pplx-decider-v1.3` (27B) and `pplx-decider-flash-v1.3` (4B) can now answer multiple questions about the same state in one pass, with each question isolated. Pricing is $0.02 per million input tokens for the larger model and $0.01 for the flash model; output tokens remain free.

— Names models and pricing but no usage example snapshot-20261113

03 **Decider alias version bump** BREAKING 60

The aliases `pplx-decider-latest` and `pplx-decider-flash-latest` now point to v1.3 instead of v1.2, which may change probability outputs; callers should pin `pplx-decider-v1.2` or `pplx-decider-flash-v1.2` to stay on the previous version while revalidating thresholds.

— Names exact aliases and migration path `snapshot-20261113`

└──▷ **BREAKING ON UPGRADE**

! `pplx-decider-latest` now resolves to `pplx-decider-v1.3` and `pplx-decider-flash-latest` resolves to `pplx-decider-flash-v1.3`; callers using these aliases may see different probability outputs and must revalidate any thresholds tuned on v1.2.

WAS THIS USEFUL? 👍 👎

◆ AI Agent Frameworks

Arcade

Feedback ⓘ updated · Nov 13, 2026 NOTES ↗ RANK

Arcade provides an actions runtime that lets AI agents securely call external services with authorization and governance controls.

> Arcade's biggest addition this window is a safety-gated `purge-execution-logs` CLI command, alongside multi-account OAuth connection profiles and a cluster of breaking changes that tighten file-creation behavior and narrow Microsoft Graph and Figma scopes across its toolkits.

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

01 **purge-execution-logs CLI command** NEW 90

Run the new `purge-execution-logs` CLI command on the engine image to delete old tool-execution rows; without flags it only reports what it would delete, and an explicit `--confirm` flag together with a row cap and a `--reason` is required to actually delete rows.

Actually delete old tool-execution rows up to a cap, supplying a reason for the audit trail.

```
$ purge-execution-logs --confirm --cap <row_count> --reason 'quarterly log retention cleanup'
```

— Runnable command with exact flags and example given snapshot-20261113

02 **Narrower Microsoft Graph scopes for Word and Excel** BREAKING 75

Word and Excel tools now request the narrower `Files.Read.All` and `Files.ReadWrite.All` Microsoft Graph scopes instead of broader file permissions; this is a breaking change, so teams should review any pinned toolkit versions and stored Microsoft Graph app consent before upgrading.

— Names exact scopes and gives upgrade guidance snapshot-20261113

03 **Filter execution history by API key and gateway** NEW

65

Tool execution history can now be filtered and reported by the API key and gateway that ran each call, backed by new nullable `api_key_id` and `gateway_id` columns added to the tool-execution table; these fields only populate on a call once the database migration adding the columns has been applied.

— Names columns and migration dependency but no API surface snapshot-20261113

THINNER COVERAGE BELOW

04 **Fail-by-default file creation across Drive, Word, Excel**

55

BREAKING

Google Drive, Word, and Excel tools that create a file now fail by default when a file already exists at the target path instead of silently overwriting it; callers can still choose to overwrite or auto-rename. This is a breaking behavior change requiring opt-in for the old overwrite behavior.

— Explains before/after but no exact flag names given snapshot-20261113

05 **Figma toolkit major version breaking change** BREAKING

55

Figma toolkit has a breaking major-version change: the previous 1.x line is no longer served, and the toolkit now requires a broader API scope than Arcade will request on an end-user's behalf for two endpoints.

— Names version line and scope gap but not the two endpoints snapshot-20261113

06 **Multi-account OAuth connection profiles** NEW

50

Organizations can now turn on multiple named connection profiles per OAuth provider, letting an end-user authorize and switch between separate accounts (for example, a personal and a work Google account) for the same toolkit. Usage reporting now also records calls to the profile-management tool, capturing the action taken, its result, and hashed user and session identifiers.

— Describes mechanism but no config key or endpoint named snapshot-20261113

07 **Pagination for Google Drive SearchFiles** IMPROVED

45

Google Drive's `SearchFiles` tool can now page through more than the first 50 results.

— Names tool and prior 50-result limit snapshot-20261113

08 **LLM-friendly OAuth error explanations** IMPROVED

30

OAuth error responses can now include a developer-facing explanation formatted for pasting into an LLM prompt.

— Describes what but not how it's surfaced `snapshot-20261113`

09 **OneDrive targeting for Word and Excel tools** NEW 25

Word and Excel tools can now also target OneDrive.

— One-line addition with no further detail `snapshot-20261113`

10 **Gateway linking for remote MCP App resources** NEW 25

A gateway can now link a remote MCP App's resources back to the gateway that served them.

— Bare mechanism statement, no further detail `snapshot-20261113`

11 **ListFiles tools removed** BREAKING 20

The `ListFiles` tools are removed entirely.

— Bare removal notice with no replacement guidance `snapshot-20261113`

└──▷ BREAKING ON UPGRADE

! The tool execution history API only reports `api_key_id` and `gateway_id` on a call once the database migration adding those nullable columns to the tool-execution table has been applied.

! Figma toolkit has a breaking major-version change: the previous 1.x line is no longer served, and the toolkit now requires a broader API scope than Arcade will request on an end-user's behalf for two endpoints.

! Word and Excel toolkits have breaking changes: review any pinned toolkit versions and stored Microsoft Graph app consent before upgrading, as scopes have narrowed to `Files.Read.All` and `Files.ReadWrite.All`.

! Google Drive, Word, and Excel file-creation tools now fail instead of silently overwriting an existing file at the target path — callers must explicitly opt in to overwrite or auto-rename behavior.

! The `ListFiles` tools are removed entirely.

WAS THIS USEFUL? 👍 👎

truefoundry TrueForge

Feedback ⓘ

@truefoundry/trueforge-ui@0.185.0 NOTES ↗ RANK

TrueForge is the open-source agent harness - the runtime layer that turns an LLM into a working agent.

> TrueForge 0.185.0 introduces a paused-turn workflow for human-in-the-loop tool approvals and questions, a tenant-wide memory toggle with pluggable providers, a built-in MCP server, and a new service-scoped turn endpoint, alongside sandbox environment admin controls and access-grant-based session sharing — shipped with a substantial set of breaking API, SDK, and UI-runtime changes.

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

01 Paused-turn workflow for tool approvals and questions

90

BREAKING

Turns now pause instead of finishing while waiting for user input (tool approvals, questions, MCP authentication); clients send answers as events on the paused turn and can reconnect to its event stream at any point. This breaks custom chat servers, which must now implement `sendTurnEvents` to deliver answers to a paused turn, and `CreateTurnRequest.input` is now required. In `@truefoundry/trueforge-assistant-ui-runtime`, `useTrueForgeResumeMcpAuth` is renamed to `useTrueForgeContinueMcpAuth`, `findCurrentPausedAssistantMessage` is renamed to `findPausedAssistantMessage`, and `collectRequiredActionInputs`, `collectApprovalInputs`, `toTrueForgeApprovalInputs`, `collectResponseInputs`, and `TOOL_RESPONSE_THREAD_ID_CUSTOM_KEY` are removed in favor of `useTrueForgeRespondToToolApproval` and `useTrueForgeRespondToToolResponse`; in `@truefoundry/trueforge-ui`, `ToolApprovalContainer` now takes `approvalId` and `onRespond` instead of `options` and `onSelectOption`.

— Names every renamed/removed hook and required server method

@truefoundry/trueforge-ui@0.185.0 @truefoundry/trueforge-sdk@0.185.0 @truefoundry/trueforge-core@0.185.0 @truefoundry/trueforge-assistant-ui-run…

02 Configurable page size for session and turn events

BREAKING

88

Session-event and turn-event page size is now configurable via `LIST_EVENTS_PAGE_LIMIT_MAX` (default 500), which sets both the default and maximum `limit` for `GET /sessions/{session_id}/events` and `GET`

`/sessions/{session_id}/turns/{turn_id}/events`; requests exceeding the limit return 400. This is breaking: the `listEvents`, `listTurnEvents`, and `listSessionEvents` server ports no longer accept a `limit` parameter.

Cap the maximum number of events returned per page for session and turn event streams to stay within downstream processing limits.

```env
LIST_EVENTS_PAGE_LIMIT_MAX=200
```

Tune how many events are returned per page on session and turn event endpoints to avoid hitting the 400 limit on high-volume sessions.

```env
LIST_EVENTS_PAGE_LIMIT_MAX=200
```

— Exact endpoints, config key, default, and failure behaviour given

`@truefoundry/trueforge-ui@0.185.0` `@truefoundry/trueforge-assistant-ui-run…`

03 **Sandbox environment admin controls** IMPROVED 88

Sandbox environments are listed with the default environment first, then sorted by most recently updated (agents are also sorted by most recently updated), with ordering applied in the backend before pagination. The default sandbox environment cannot be deleted (returns 409); the reserved platform-provisioned environment named `trueforge-sandbox-env` can only be enabled or disabled, and create/update reject the reserved name. API callers that omit the sandbox environment now resolve the flagged default at request time rather than the environment named `trueforge-sandbox-env`.

Disable a sandbox environment and promote a different one as the new default without touching its definition or history.

```
$ # Disable the old environment
curl -X PATCH https://<host>/api/v1/sandbox-environments/legacy-
env \
  -H 'Content-Type: application/json' \
  -d '{"enabled": false}'

# Promote the new environment as default
curl -X PATCH https://<host>/api/v1/sandbox-environments/prod-env
\
  -H 'Content-Type: application/json' \
  -d '{"is_default": true}'
```

Disable a sandbox environment that is no longer fit for selection while keeping existing chats and agents that reference it running.

```
$ curl -X PATCH https://<host>/api/v1/sandbox-environments/my-old-env \
  -H 'Content-Type: application/json' \
  -d '{"enabled": false}'
```

Promote a ready environment to the tenant default so new chats and agents pick it up automatically.

```
$ curl -X PATCH https://<host>/api/v1/sandbox-environments/my-new-env \
  -H 'Content-Type: application/json' \
  -d '{"is_default": true}'
```

— Runnable PATCH commands and exact 409/reserved-name behaviour given

`@truefoundry/trueforge-ui@0.185.0` `@truefoundry/trueforge-assistant-ui-run…`

04 **Service account turns via internal API** NEW 84

Service accounts can start non-streaming turns via `POST /api/internal/sessions/{session_id}/turns`, using the new `SessionSource` value `service`; this lets a backend service inject a turn non-interactively, bypassing memory ingest.

Call a turn non-interactively from a backend service, bypassing memory ingest, using the new service-scoped endpoint.

```
$ curl -X POST https://<host>/api/internal/sessions/{session_id}/turns \
  -H 'Authorization: Bearer <service-api-key>' \
  -H 'Content-Type: application/json' \
  -d '{"external_id": "svc-call-001", "source": "service", "input": "Summarise the latest report"}'
```

Inject a backend-service turn into a session without triggering memory ingest, using a service API key.

```
$ curl -X POST
https://<host>/api/internal/sessions/{session_id}/turns \
  -H 'Authorization: Bearer <service-api-key>' \
  -H 'Content-Type: application/json' \
  -d '{"external_id": "job-42", "source": "service", "input":
"Summarise the overnight batch report."}'
```

— Exact endpoint, auth header, and payload shown `@truefoundry/trueforge-sdk@0.185.0` `@truefoundry/trueforge-core@0.185.0`

05 **Memory toggle and provider configuration** NEW 75

Admins can configure memory in OSS via Settings → Memory (`/api/v1/settings/memory-providers`) or via `TRUEFOUNDRY_MEMORY_*` environment variables in TrueFoundry mode; a tenant-wide memory toggle gates memory in chats, ingestion, and agent specs.

Configure the Zep memory provider in OSS so the tenant memory toggle and agent Memory option become available.

> 📍 In the TrueForge admin console, go to Settings → Memory (`/api/v1/settings/memory-providers`), enter your Zep connection details, save, then enable the tenant-wide memory switch to make the Memory toggle appear in the agent builder.

Enable Zep as a tenant-wide memory provider in OSS so the Memory toggle becomes available in the agent builder.

> 📍 In Settings → Memory, add your Zep connection details. The toggle then appears in the agent builder for each agent.

— Names config endpoint and env var prefix with UI path `@truefoundry/trueforge-ui@0.185.0` `@truefoundry/trueforge-sdk@0.185.0`

THINNER COVERAGE BELOW

06 **Access grants replace session sharing** BREAKING 56

Session sharing is replaced with access grants; session granularity is now required at the tenant or subject level across the API and UI, and subject granularity enables user invitations. The deprecated session `shared` field and PATCH sharing flow are removed, the `session.shared` database column is dropped, and the exported schema/type `CreatedBySubject` is renamed to `Subject` .

— Names dropped field, column, and type rename but no example

`@truefoundry/trueforge-ui@0.185.0` `@truefoundry/trueforge-core@0.185.0`

`@truefoundry/trueforge-assistant-ui-run…`

07 **Parallel tool call grouping in Agent steps** IMPROVED

50

Parallel tool calls issued in the same model step are now grouped under an N tool calls in parallel row in Agent steps, including nested sub-agent transcripts; sub-agent steps show a tree branch per step.

– Describes UI behaviour but gives no path or command `@truefoundry/trueforge-ui@0.185.0`

08 **Built-in TrueForge MCP server** NEW

50

Adds the built-in TrueForge MCP server for managing agents and supporting resources from draft sessions; it now uses a stable id and name `trueforge:mcp` and labels its tools as `truefoundry-system`.

– Names the server id but gives no usage steps `@truefoundry/trueforge-ui@0.185.0`

`@truefoundry/trueforge-assistant-ui-run…`

NEW INTEGRATION Serply MCP server preset

UPDATED INTEGRATION Zep memory SDK migration

NEW SUPPORTED PLATFORM Model catalog expansion

└──▷ **BREAKING ON UPGRADE**

! In `@truefoundry/trueforge-assistant-ui-runtime`: `useTrueForgeResumeMcpAuth` is renamed to `useTrueForgeContinueMcpAuth`, and `findCurrentPausedAssistantMessage` is renamed to `findPausedAssistantMessage`.

! In `@truefoundry/trueforge-assistant-ui-runtime`: `collectRequiredActionInputs`, `collectApprovalInputs`, `toTrueForgeApprovalInputs`, `collectResponseInputs`, and `TOOL_RESPONSE_THREAD_ID_CUSTOM_KEY` are removed; use `useTrueForgeRespondToToolApproval` and `useTrueForgeRespondToToolResponse` instead.

! In `@truefoundry/trueforge-ui`: `ToolApprovalContainer` now takes `approvalId` and `onRespond` instead of `options` and `onSelectOption`.

! Custom chat servers must implement `sendTurnEvents` to deliver answers to a paused turn, and `CreateTurnRequest.input` is now required.

! The deprecated session `shared` field and PATCH sharing flow are removed in favor of access grants, and the `session.shared` database column is dropped.

! The exported schema/type `CreatedBySubject` is renamed to `Subject`.

! The `listEvents`, `listTurnEvents`, and `listSessionEvents` server ports no longer accept a `limit` parameter; page size is controlled by the `LIST_EVENTS_PAGE_LIMIT_MAX` server setting.

! API callers that omit the sandbox environment now resolve the flagged default at request time rather than the environment named `trueforge-sandbox-env`.

- ! Custom chat servers must now implement `sendTurnEvents` to deliver answers to a paused turn, and `CreateTurnRequest.input` is now required (omitting or sending an empty message is rejected).
- ! The session `shared` field and PATCH sharing flow are removed in favor of access grants; the `session.shared` database column is dropped.
- ! The exported `CreatedBySubject` schema/type is renamed to `Subject`.
- ! In `@truefoundry/trueforge-assistant-ui-runtime`: `useTrueForgeResumeMcpAuth` is renamed to `useTrueForgeContinueMcpAuth`; `findCurrentPausedAssistantMessage` is renamed to `findPausedAssistantMessage`; `collectRequiredActionInputs`, `collectApprovalInputs`, `toTrueForgeApprovalInputs`, `collectResponseInputs`, and `TOOL_RESPONSE_THREAD_ID_CUSTOM_KEY` are removed — use `useTrueForgeRespondToToolApproval` and `useTrueForgeRespondToToolResponse` instead.
- ! Custom chat servers must now implement `sendTurnEvents` to deliver answers to a paused turn; `CreateTurnRequest.input` is required (omitting or sending an empty value is rejected).
- ! `useTrueForgeResumeMcpAuth` is renamed to `useTrueForgeContinueMcpAuth` in `@truefoundry/trueforge-assistant-ui-runtime`.
- ! `findCurrentPausedAssistantMessage` is renamed to `findPausedAssistantMessage` in `@truefoundry/trueforge-assistant-ui-runtime`.
- ! `collectRequiredActionInputs`, `collectApprovalInputs`, `toTrueForgeApprovalInputs`, `collectResponseInputs`, and `TOOL_RESPONSE_THREAD_ID_CUSTOM_KEY` are removed from `@truefoundry/trueforge-assistant-ui-runtime`; use `useTrueForgeRespondToToolApproval` and `useTrueForgeRespondToToolResponse` instead.
- ! Custom chat servers must now implement `sendTurnEvents` to deliver answers to a paused turn, and `CreateTurnRequest.input` is required.
- ! `ToolApprovalContainer` in `@truefoundry/trueforge-ui` now takes `approvalId` and `onRespond` instead of `options` and `onSelectOption`.
- ! `listEvents`, `listTurnEvents`, and `listSessionEvents` server ports no longer accept a `limit` parameter; page size is controlled by `LIST_EVENTS_PAGE_LIMIT_MAX`.

`@truefoundry/trueforge@0.185.0` NOTES ↗ STABLE

TrueForge adds a built-in MCP server, paused-turn resumption with event-stream reconnection, a Memory toggle for agents, and sandbox environment admin controls including enable/disable and default-slot management.

└──▷ **GET THIS VERSION** ▾

pipx

```
$ pipx install trueforge-sdk==0.185.0
# already installed? upgrade to this version:
$ pipx install --force trueforge-sdk==0.185.0
```

npm

```
$ npm install -g @truefoundry/trueforge@0.185.0
```

git

```
$ git clone --branch @truefoundry/trueforge@0.185.0
https://github.com/truefoundry/trueforge.git
# already have the repo? check out this version:
$ git checkout @truefoundry/trueforge@0.185.0
```

PyPI attestation · signed build provenance · checked 13 Nov Wrong command?

Adds `POST /api/internal/sessions/{session_id}/turns` (service API key) for non-streaming turn creation with `external_id`, non-null `source`, and a `service` SessionSource arm that skips memory ingest.

Adds `PATCH /api/v1/sandbox-environments/{name}` with `enabled` and `is_default` fields so admins can disable environments and choose which one is the default.

Adds a Memory toggle to the agent builder, configured via Settings → Memory (`/api/v1/settings/memory-providers`) in OSS or via `TRUEFOUNDRY_MEMORY_*` environment variables in TrueFoundry mode; the tenant-wide switch now defaults to on.

Adds the built-in TrueForge MCP server (`trueforge:mcp`) for managing agents and supporting resources from draft sessions.

Adds `LIST_EVENTS_PAGE_LIMIT_MAX` (default 500) to make the page size for `GET /sessions/{session_id}/events` and `GET /sessions/{session_id}/turns/{turn_id}/events` configurable; requests above the limit return 400.

+6 more

└──▷ **BREAKING ON UPGRADE**

- ! The session `shared` field and PATCH sharing flow are removed in favor of access grants; the `session.shared` database column is dropped.
- ! The exported `CreatedBySubject` schema/type is renamed to `Subject`.
- ! `@truefoundry/trueforge-assistant-ui-runtime`: `useTrueForgeResumeMcpAuth` is renamed to `useTrueForgeContinueMcpAuth`, and `findCurrentPausedAssistantMessage` to `findPausedAssistantMessage`; `collectRequiredActionInputs`, `collectApprovalInputs`, `toTrueForgeApprovalInputs`, `collectResponseInputs`, and

`TOOL_RESPONSE_THREAD_ID_CUSTOM_KEY` are removed — use `useTrueForgeRespondToToolApproval` and `useTrueForgeRespondToToolResponse` instead.

! `@truefoundry/trueforge-ui`: `ToolApprovalContainer` now takes `approvalId` and `onRespond` instead of `options` and `onSelectOption`.

! Custom chat servers must implement `sendTurnEvents` to deliver answers to a paused turn; `CreateTurnRequest.input` is now required.

! `GET /api/v1/mcp-servers/{name}/authorize` now rejects an empty `return_to` with 400.

! `listEvents`, `listTurnEvents`, and `listSessionEvents` server ports no longer accept a `limit` parameter; page size is controlled by `LIST_EVENTS_PAGE_LIMIT_MAX`.

WAS THIS USEFUL?  

Microsoft Agent Framework

Feedback ⓘ python-1.26.0

NOTES ↗ RANK

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

> Microsoft Agent Framework's python-1.26.0 release adds fine-grained control over loop responses and message classification, structured output support for the GitHub Copilot integration, and two breaking changes tightening CLI and execution-timeout requirements for the GitHub Copilot and Hyperlight integrations.

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

01 **Hyperlight execution timeout enforcement** BREAKING 90

The `agent-framework-hyperlight` integration now interrupts cancelled guest execution and applies a 30-second execution budget by default using Sandbox.interrupt_handle(), which requires Hyperlight SDK 0.9.0 or later at runtime. Set `execution_timeout_seconds` on `HyperlightCodeActProvider` or `HyperlightExecuteCodeTool` for workloads that need more than 30 seconds.

— Mechanism, default, SDK version, and config key given python-1.26.0

02 **GitHub Copilot CLI version requirement** BREAKING 85

The `agent-framework-github-copilot` integration now requires GitHub Copilot CLI 1.0.85 or later and fails startup when the CLI version is older or cannot be determined. Upgrade any custom runtimes selected by `cli_path`, `GITHUB_COPILOT_CLI_PATH`, or `COPILOT_CLI_PATH` to CLI 1.0.85 or later.

— Exact version, env vars, and migration step named python-1.26.0

03 **Final-only loop response control** NEW 75

Non-streaming callers can receive only the final loop response by passing `loop_return_final_only` to create_harness_agent(), avoiding intermediate loop messages being surfaced.

Return only the final loop response to a non-streaming caller, avoiding intermediate loop messages.

```python
from agent_framework import create_harness_agent

agent = create_harness_agent(
    client=my_client,
    instructions="You are a helpful assistant.",
    loop_return_final_only=True,
)
```

— Named flag with a runnable example `python-1.26.0`

04 Structured output for GitHub Copilot integration NEW

60

The GitHub Copilot integration now accepts Pydantic models and JSON Schema mappings via `response_format`, with parsed response values returned in both streaming and non-streaming runs.

— Names field and behaviour, lacks an example `python-1.26.0`

THINNER COVERAGE BELOW

05 Message labeler callback for context providers NEW

45

Context-provider messages can be classified at the application layer with the optional experimental `SecureAgentConfig.message_labeler` callback.

— Names config field but no usage shown `python-1.26.0`

06 Task-scoped authority middleware sample NEW

30

Adds task-scoped authority middleware guidance and an executable sample to the samples collection.

— Vague description with no named surfaces `python-1.26.0`

└──▷ BREAKING ON UPGRADE

! The `agent-framework-github-copilot` integration now requires GitHub Copilot CLI 1.0.85 or later and fails startup when the CLI version is older or cannot be determined. Upgrade any custom runtimes selected by `cli_path`, `GITHUB_COPILOT_CLI_PATH`, or `COPILOT_CLI_PATH` to CLI 1.0.85 or later.

! The `agent-framework-hyperlight` integration now interrupts cancelled guest execution and applies a 30-second execution budget by default using Sandbox.interrupt_handle(), which requires Hyperlight SDK 0.9.0 or later at runtime. Set `execution_timeout_seconds` on `HyperlightCodeActProvider` or `HyperlightExecuteCodeTool` for workloads that need more than 30 seconds.

WAS THIS USEFUL?  

openclacky

Feedback ⓘ v1.5.24 NOTES ↗ RANK

openclacky is a token-efficient open-source AI agent framework.

> OpenClacky v1.5.24 adds opt-in automatic memory recall, WebSocket event support for extensions, and streaming assistant replies, alongside smaller UI and model-picker refinements.

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

01 **WebSocket events for extensions** NEW 50

Extensions can now follow WebSocket events and run before the first think step, giving extension authors a hook into the event stream earlier in the processing pipeline.

– Names the mechanism but no API or flag details v1.5.24

02 **Model picker hints and new model support** IMPROVED 50

The model picker now shows hints for audio, speech-to-text, and video understanding capabilities, and adds Claude Fable 5.2 to the hosted model picker.

– Names a specific added model but little mechanism detail v1.5.24

03 **Opt-in automatic memory recall** NEW 40

A new opt-in auto-recall feature retrieves relevant memories automatically before each message is sent, rather than requiring the user to invoke recall manually.

– Describes behaviour but no config key or flag to enable it v1.5.24

04 **Temp-directory file preview in UI** NEW 40

Files that an AI turn leaves in the system temp directory can now be previewed directly in the UI instead of requiring a separate download.

– Clear what it does, no exact UI path given v1.5.24

05 **Interface polish: sidebar, backup, promo, extension icon** IMPROVED 40

Several small UI improvements shipped together: the Config and My Data sidebar sections can be collapsed to reduce clutter, backup packing and downloading now shows progress, the OpenClacky AI Keys promo can be dismissed and stays dismissed across sessions, and the secretary extension gained an icon and a detail page.

– Enumerates each UI change but no mechanism beyond description v1.5.24

06 **Streaming assistant replies** IMPROVED 35

Assistant replies are now streamed as they are written instead of waiting for a complete response before displaying it.

— Explains behaviour change with no further mechanism detail v1.5.24

WAS THIS USEFUL? 👍 👎

persona

Feedback ⓘ @runtypelabs/persona@4.38.0 NOTES ↗ RANK

Persona is a create agentic front-end experiences for the web in VanillaJS. Lightweight, extensible, and WebMCP-native.

> Persona shipped a pluggable streaming transport layer with a built-in WebSocket implementation, real-time multi-tab sync over Runtype's client-session socket, and a new server-pinned copilot driving the UI SDK Agent Bridge demo.

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

01 Pluggable stream transport with WebSocket implementation NEW

88

All streaming requests (send, resume, approval, durable reconnect) can now be routed through a custom transport by setting `config.transport` with a `PersonaTransport` implementation, falling back to HTTP + SSE when omitted. A ready-made `createWebSocketTransport` function (lazy-loaded from `@runtypelabs/persona/websocket-transport`, or via AgentWidget.loadWebSocketTransport() on CDN builds) connects any WebSocket backend using a user-supplied codec.

Connect a widget to an OpenAI Realtime API WebSocket backend instead of the default HTTP + SSE transport.

typescript

```typescript
import { AgentWidget } from '@runtypelabs/persona';
import { createWebSocketTransport } from
'@runtypelabs/persona/websocket-transport';

const widget = new AgentWidget({
  clientToken: '<token>',
  transport: createWebSocketTransport({
    url: 'wss://your-backend/realtime',
    codec: myCodec
  })
});
```

— Names exact config key, function, import path and fallback behaviour

@runtypelabs/persona@4.38.0

02 Real-time multi-tab sync via client-session socket NEW

84

In client-token mode, widgets gain real-time multi-tab rendering over the Runtype client-session socket (`runtype-client-session-v1`) whenever `/v1/client/init` advertises it. Behavior is tunable via `realtime: false` to opt out, or `realtime.connect` set to `'load'` , `'open'` , `'focus'` (default), `'input'` , or a custom hook function.

Have the real-time client-session socket connect only when the user starts typing, rather than on page load, to reduce idle connections.

typescript

```typescript
import { AgentWidget } from '@runtypelabs/persona';

const widget = new AgentWidget({
  clientToken: '<token>',
  realtime: {
    connect: 'input'
  }
});
```

— Names socket protocol, endpoint, config key and all connect modes

`@runtypelabs/persona@4.38.0`

THINNER COVERAGE BELOW

03 **WEBMCP_UI_SDK_AGENT settings copilot** NEW 48

Adds `WEBMCP_UI_SDK_AGENT` , a server-pinned settings copilot that drives the `snapshot` , `query` , and `act` page tools published by the UI SDK agent bridge, for the UI SDK Agent Bridge demo.

— Names the agent and tools but no usage example or config detail

`@runtypelabs/persona-proxy@4.38.0`

WAS THIS USEFUL? 👍 👎

Databricks Omnigent

Feedback ⓘ v0.22.0 NOTES ↗

Omnigent is an 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 adds nested agents in the sidebar, upstream HTTP proxy support for sandbox traffic, desktop keyboard shortcuts, PR diff search and file-viewed tracking, and `@` tagging in the composer.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v0.22.0 https://github.com/omnigent-ai/omnigent.git
# already have the repo? check out this version:
$ git checkout v0.22.0
```

Agents spawned via the Omnigent MCP or created on the same managed sandbox now appear nested in the left sidebar for better organization.

Sandbox traffic can now route through upstream HTTP proxies.

NEW INTEGRATION Managed and sandbox sessions can now fork to any host or sandbox provider.

The desktop app now has a native menu and keyboard shortcuts covering new terminal and browser tabs, pin, archive, fork, sidebar toggles, workspace tab switching, reopening closed tabs, back and forward, and an optional global shortcut to bring Omnigent to the front.

The pull request Changes tab now supports filtering changed files, searching diff text with Cmd/Ctrl+F, marking files as viewed with progress tracking, and browsing per-commit diffs.

+1 more

NEW INTEGRATION Automation triggers for Jira, Slack, GitHub, Linear, and PagerDuty

WAS THIS USEFUL? 👍 👎

Vercel AI SDK

Feedback ⓘ ai@7.0.172 NOTES ↗ RANK

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

> Vercel AI SDK tightened file-download security around experimental extraction and decision APIs, added a `redactToolErrors` option to mask tool call errors on both UI streams and model history, and sped up UI message validation.

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

01 Mandatory download hook for remote files

BREAKING — 95

`createDownload` gained a `fetch` option so downloads can use `guarded-fetch` for opt-in DNS validation on Node.js; the function it returns can be passed as `experimental_download` to `generateText` and `streamText`, now also handles generated file URLs returned by the model, and the call's abort signal is passed through to custom download requests (including prompt assets and generated files) to preserve cancellation and timeouts. A later release made this hook mandatory: remote file URLs passed to `experimental_extract` and `experimental_decide` no longer download silently via global fetch — callers must supply an explicit `experimental_download` hook (e.g. createDownload({ fetch })) or the call fails; inline file data still works without a hook.

Use a guarded fetch transport with `createDownload` to enforce DNS validation when downloading file inputs in an untrusted environment.

typescript

```typescript
import { createDownload } from 'ai';
import { guardedFetch } from 'guarded-fetch';

const download = createDownload({ fetch: guardedFetch });

const result = await generateText({
  model: 'openai/gpt-6-astra',
  prompt: 'Describe this file.',
  experimental_download: download,
});
```

— Names exact flags, functions and before/after behavior with a runnable example

ai@7.0.172 ai@7.0.171

02 redactToolErrors option for tool call errors

BREAKING — 80

Setting `redactToolErrors: true` on `toUIMessageStream` and `toUIMessageChunk` masks failed tool call error text before sending it to the client, restoring the prior 'An error occurred.' behavior — needed because these functions now send the model's full error text by default. The same `redactToolErrors: true` option on model message history (used with `generateText` / `streamText`) replaces tool call error text with the result of `onError` instead of keeping the original error.

Restore the previous error-masking behavior so raw tool error details are never exposed to the client.

typescript

```
return toUIMessageStream(stream, {
  redactToolErrors: true,
});
```

— Names exact functions and option with a code example, minus deeper mechanism `ai@7.0.171` `ai@6.0.331`

THINNER COVERAGE BELOW

03 **Faster UI message validation** IMPROVED 45

`validateUIMessages` and `safeValidateUIMessages` now validate more efficiently, checking each part only against schemas for its type and stopping at the first invalid message or part.

— Explains mechanism but no example or config to act on `ai@6.0.331`

04 **Experimental extraction API** NEW 40

Adds an experimental extraction API and extraction model spec via `experimental extract api` for structured data extraction workflows.

— Names the surface but no usage detail or example `ai@7.0.171`

└──▷ **BREAKING ON UPGRADE**

! Remote file URLs passed to `experimental_extract` and `experimental_decide` no longer download silently via global fetch; callers must now supply an explicit `experimental_download` hook (e.g. createDownload({ fetch })) or the call will fail.

! `toUIMessageStream` and `toUIMessageChunk` now send the model's full error text for failed tool calls and invalid tool calls to the client by default, instead of 'An error occurred.'; set `redactToolErrors: true` to restore the previous behavior.

WAS THIS USEFUL? 👍 👎

HeyGen HyperFrames

Feedback ⓘ v0.9.41 NOTES ↗ RANK

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

> HyperFrames shipped CLI additions for cloud render inspection and catalog search, moved several Studio editor reads onto a message-channel-based preview, added a desktop-importable keys subpath, a live transform readout chip, per-layer history tracking, and new image-generation model defaults and pricing.

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

01 **JSON output for cloud render exclusions** NEW 70

Cloud render now accepts `--json` to list the files an upload leaves out, letting you inspect what gets excluded before committing to a render via `hyperframes cloud render --json`.

Inspect which local files would be omitted from a cloud upload before committing to a paid render.

```
$ hyperframes cloud render --json
```

— Exact flag and runnable command given v0.9.41

02 **Studio keys subpath for desktop hosts** NEW 65

Key parsing and defaults can now be imported in desktop or other host environments via the new `@hyperframes/studio/keys` subpath.

— Named import path but no runnable example v0.9.39

03 **Registry search command** NEW 60

Adds `hyperframes registry search` to run catalog lookups from the CLI, letting you find a ready-made component before building one by hand.

Search the block catalog from the terminal to find a ready-made component before building one by hand.

```
$ hyperframes registry search
```

— Exact command, thin on mechanism v0.9.40

04 **Image generation model updates** NEW 60

The `heygen generate` command now defaults image generation to the `nano-banana-2.1` model, adds support for the `minimax-h3-image` model, and prices and validates `minimax-h3-image` along with all image models at 1.5x.

— Named models and command, pricing figure given v0.9.38

THINNER COVERAGE BELOW

05 **Live transform readout chip** NEW 50

Studio now shows move, resize, and rotate values as a chip directly beneath the selected element while editing.

— Describes behaviour, no exact UI path v0.9.39

06 **Layer name recorded in edit history** NEW 45

Studio now records the layer name on each history entry's target, so edit history shows which layer each change touched.

— Clear mechanism but no concrete surface to act on v0.9.38

07 **Studio preview channel for editor reads** IMPROVED 40

Several Studio editor operations were migrated to read from an opaque preview over its message channel: overlays and timeline selections reading geometry, new runtime element handles for the editing channel, picker styles and composition size, and playback and timeline reads.

— Internal architecture change, no user-facing action v0.9.39 v0.9.40 v0.9.41

WAS THIS USEFUL?  

cotal

Feedback ⓘ v0.163.3 NOTES ↗ RANK

Cotal is the open standard for agent coordination

> Cotal shipped a `cotal login --cloud` shorthand for direct Cotal Cloud sign-in and made `cotal use` space selection global, alongside new callout account hibernation/renewal on hosted auth planes, workflow validation for empty programs, seamless credential-swap connection handling via `overlapOnConnectionRenewal`, and a `ps -free` `cotal update` on Linux.

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

01 Direct Cotal Cloud sign-in shorthand NEW

85

Sign in to Cotal Cloud from the terminal without knowing its provider URL using `cotal login --cloud` or `cotal daemon login --cloud`, which selects a personal or sole space after approval and opens a browser when a desktop terminal supports it.

Sign in to Cotal Cloud from a terminal environment without looking up the provider URL first.

```
$ cotal login --cloud
```

Sign in to Cotal Cloud through the daemon (e.g. in a headless server setup) using the same shorthand.

```
$ cotal daemon login --cloud
```

— Runnable commands with clear mechanism and scope given v0.163.0

02 `cotal update` works without `ps` on Linux IMPROVED

75

`cotal update` on Linux now reads `/proc` directly instead of requiring `ps` on PATH, so it works in slim container images or services with a narrow PATH.

Update cotal in a slim container image where `ps` is not installed.

```
$ cotal update --self
```

— Clear before/after with a runnable command example v0.163.2

03 **Seamless credential swap for serving connections** NEW

70

A serving endpoint now keeps command dispatch active across a planned credential swap using `overlapOnConnectionRenewal`, subscribing on a second connection and flushing it before reconnecting the resident connection.

— Mechanism detailed with named config key, no example v0.163.3

THINNER COVERAGE BELOW

04 **Callout account hibernation and credential renewal** NEW

55

Hosts can now hibernate inactive data and auth-callout accounts on a full-resolver broker, then wake them without changing account identity or authority. Hosted auth planes can also persist callout account claim transitions and renew callout credentials without changing account or user identities.

— Describes mechanism but no command or config path given v0.163.2

05 **Guidance on re-authentication after refused sign-ins** IMPROVED

55

Refused or expired sign-ins now tell you how to sign in again, using `cotal login` for Cotal Cloud or naming `--idp` for another provider.

— Names the commands involved but no worked example v0.163.2

06 **Workflow validation rejects empty programs** NEW

50

cotal now rejects workflows whose text contains only comments or whitespace, reporting validation error `L1031` ('program has no statements').

— Names error code but no usage example shown v0.163.3

07 **Global space selection with `cotal use`** IMPROVED

50

Selections made with `cotal use` now apply in every folder, overriding the folder's running space.

— Names the command and new scope but lacks example v0.163.0

WAS THIS USEFUL? 👍 👎

AWS Strands Agents

Feedback ⓘ harness-typescript/v0.4.0

NOTES ↗ RANK

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

> Strands Agents shipped a major vended-tools expansion (swarm, graph, read/write/edit, python_repl, a2a_client, subagent, mcp_router) alongside a new pluggable decision-model system spanning SageMaker and System One, plus a run of breaking changes to injected-text history, default plugins, agent stash sharing, and context-offload granularity across the Python and TypeScript SDKs.

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

01 Pre-built vendor clients in Anthropic and OpenAI providers

NEW 90

The Anthropic and OpenAI providers in both Python and TypeScript now accept pre-built vendor clients, enabling Bedrock InvokeModel and Mantle passthrough, and pass the caching setting to OpenAI and Bedrock Mantle as an SDK cache config.

Bring an already-configured Anthropic client (with custom retry or proxy settings) into the provider instead of letting the SDK build one.

python

```python
import anthropic
from strands.models.anthropic import AnthropicModel
from strands import Agent

client = anthropic.Anthropic(api_key='...', max_retries=5)
model = AnthropicModel(client=client, model_id='claude-opus-4-5')
agent = Agent(model=model)
agent('Summarize the last 10 CVEs in the NVD feed')
```

— Named providers, targets, and a runnable client-injection example.

harness-typescript/v0.4.0

02 Pluggable decision model system across providers NEW

80

Introduces DecisionModel and LLMDecisionModel primitives in Python and TypeScript, letting ClassifierStrategy and HumanInTheLoop route decisions through any decision model rather than only the built-in classifier. New concrete implementations include SageMakerDecisionModel for TypeScript (routing through

System One models on SageMaker AI endpoints) and `SystemOneDecisionModel` for Jev and decisionapi.net in Python, plus System One providers across both SDKs.

Route TypeScript agent decisions through a SageMaker-hosted System One model instead of the default Bedrock model.

typescript

```
import { SageMakerDecisionModel } from '@strands-agents/harness'
import { createHarness } from '@strands-agents/harness'

const agent = await createHarness({
  decisionModel: new SageMakerDecisionModel({ endpointName:
'<your-sagemaker-endpoint>' })
})
```

— Names the primitives and concrete models with a working SageMaker example. `mcp/v0.4.0` `harness-typescript/v0.4.0`

---

03 **New vended tools for agent workflows** NEW 75

Strands Agents ships seven new vended tools across Python and TypeScript: `swarm` (run multi-agent swarms from within an agent), `graph` (graph-based multi-agent execution), `read`, `write`, and `edit` (replacing the now-deprecated `file_editor` / `fileEditor`), `python_repl`, `a2a_client`, `subagent`, and `mcp_router`. The shell tool's default timeout is also now configurable in the vended-tools package.

Run a multi-agent swarm from inside an agent tool call, distributing subtasks across worker agents.

python

```
from strands import Agent
from strands.vended_tools import swarm

agent = Agent(tools=[swarm])
agent('Analyse each microservice log in parallel and summarise
the error patterns')
```

— Names seven new tools and a config change but little behavioral detail per tool.

`mcp/v0.4.0` `harness-typescript/v0.4.0`

---

04 **Auto mode for HumanInTheLoop pausing** NEW 75

`HumanInTheLoop` gains an `'auto'` mode that lets the harness decide when to pause for human review, rather than always interrupting on every turn.

Let the harness automatically decide when to pause for human review rather than always interrupting on every turn.

typescript

```typescript
import { HumanInTheLoop } from '@strands-agents/harness'
import { createHarness } from '@strands-agents/harness'

const agent = await createHarness({
  humanInTheLoop: new HumanInTheLoop({ mode: 'auto' })
})
```

— Exact config key and a runnable code example. harness-typescript/v0.4.0

05 **Cancel running agent invocations** NEW 75

Agents can be cancelled via agent.cancel(), stopping the event loop mid-run, for cancelling a long-running invocation from outside the agent loop (e.g. on a user interrupt signal).

Cancel a long-running agent invocation from outside the agent loop, e.g. on a user interrupt signal.

python

```python
import asyncio
from strands import Agent

agent = Agent()

async def main():
    task = asyncio.create_task(agent.invoke_async('Scan every
file in this repo for secrets'))
    await asyncio.sleep(5)
    agent.cancel()   # stops the event loop mid-run
    await task

asyncio.run(main())
```

— Runnable example with the exact cancel method shown. mcp/v0.4.0

06 **Injected text stored as typed InjectedBlock** BREAKING 65

Injected text is now written into conversation history as a typed `InjectedBlock` (Python and TypeScript) instead of plain text, making injection auditable in the message stream; existing code that reads raw injected text from history will see a different structure and must be updated.

– Clear before/after and migration note, no exact migration steps. `mcp/v0.4.0` `harness-python/v0.4.0` `harness-typescript/v0.4.0`

07 **find_integrations MCP tool for discovering integrations** NEW

60

Adds an MCP tool `find_integrations` backed by the integrations catalog feed, letting coding agents discover available integrations by query.

– Names the tool and its data source but no usage example. `harness-typescript/v0.4.0`

THINNER COVERAGE BELOW

08 **MCP server instructions, annotations, and search improvements** IMPROVED

55

The MCP server adds instructions, tool annotations, and recoverable `fetch_doc` errors, and improves `search_docs` to accept alternate phrasings and fuse the rankings.

– Names several surfaces but describes none in depth. `harness-typescript/v0.4.0`

09 **Summarization and compaction support in TypeScript** NEW

55

Model.summarize() gets a client-side base implementation in TypeScript, enabling summarization without server-side support, and TypeScript gains Anthropic on-demand compaction (server-side summary), porting the existing Python capability.

– Explains the mechanism but gives no code to invoke it. `harness-typescript/v0.4.0`

10 **Todos plugin removed from default built-in plugins** BREAKING

55

The todos plugin is no longer included in the default built-in plugins; harnesses relying on it being present by default must now add it explicitly.

– Clear migration action but no code snippet for re-adding it. `mcp/v0.4.0` `harness-python/v0.4.0` `harness-typescript/v0.4.0`

11 **Agent stash shared via caller-scoped storage view** BREAKING

55

The agent stash is now shared across agents through a caller-scoped storage view (Python and TypeScript); code that assumed per-agent stash isolation will see cross-agent visibility.

– Describes the mechanism but no migration steps or API. `mcp/v0.4.0`

12 **Subagent and swarm interrupts persisted on parent**

BREAKING

55

Subagent and swarm interrupts are now persisted on the parent agent in TypeScript; interrupt-handling code that expected them only on child agents must be updated.

— Clear before/after but no code showing the new structure. `mcp/v0.4.0`

---

13 **defaultEffort for inference effort control** NEW

45

Adds `defaultEffort` to shared model info (Python and TypeScript) for controlling inference effort across providers.

— Names the field but not how to set it or its range of values. `harness-python/v0.4.0`

---

14 **Offload conditions now control context granularity** BREAKING

45

Offload conditions now decide granularity for context management in TypeScript; existing context offload behavior and condition implementations may change or need to be revised after upgrade.

— States the change but not which conditions or how to adapt. `mcp/v0.4.0` `harness-python/v0.4.0` `harness-typescript/v0.4.0`

---

15 **Anthropic SDK version compatibility in TypeScript** IMPROVED

40

The TypeScript SDK now supports `@anthropic-ai/sdk` versions 0.110 through 1.x.

— A version range with no further behavior described. `harness-typescript/v0.4.0`

---

16 **Agent tool streaming logic consolidated** BREAKING

40

Shared agent tool streaming logic is consolidated in TypeScript; custom tools that depended on the previous streaming dispatch path may behave differently.

— Vague on what the consolidation changes in practice. `mcp/v0.4.0`

---

17 **Bidirectional streaming support** NEW

15

Adds bidirectional streaming, as named in the release summary; no further mechanism or API detail is given.

— Only named in a summary line, no detail or example. `mcp/v0.4.0`

---

18 **S3 Vectors search** NEW

15

Adds S3 Vectors search, as named in the release summary; no further mechanism or API detail is given.

— Only named in a summary line, no detail or example. `mcp/v0.4.0`

19 **OpenAI Live provider** NEW 15

Adds an OpenAI Live provider, as named in the release summary; no further mechanism or API detail is given.

— Only named in a summary line, no detail or example. `mcp/v0.4.0`

└──▷ BREAKING ON UPGRADE

! Injected text is now written into history as an `InjectedBlock` in the TypeScript harness; existing code that reads raw injected text from history will see a different structure.

! The todos plugin is no longer included in the default built-in plugins; harnesses relying on it being present by default must now add it explicitly.

! Offload conditions now decide granularity for context management in TypeScript; existing context offload behavior may change.

! Injected text is now stored in conversation history as an `InjectedBlock` rather than plain text (Python and TypeScript) — code that reads raw history entries and expects plain strings for injected content will break.

! The default built-in plugins no longer include the todos plugin — harnesses relying on the todos plugin being present by default must add it explicitly.

! Offload conditions now decide granularity for context in TypeScript — existing context offload configuration may behave differently after upgrade.

! Injected text is now written into history as an `InjectedBlock` (Python and TypeScript) — code that reads raw injected text from history will see a different structure.

! The harness drops todos from the default built-in plugins — existing setups relying on the todos plugin will lose it on upgrade.

! The agent stash is now shared across agents through a caller-scoped storage view (Python and TypeScript) — code that assumed per-agent stash isolation will see cross-agent visibility.

! Subagent and swarm interrupts are now persisted on the parent agent in TypeScript — interrupt-handling code that expected them only on child agents must be updated.

! Shared agent tool streaming logic is consolidated (TypeScript) — custom tools that depended on the previous streaming dispatch path may behave differently.

! Offload conditions now decide granularity (TypeScript) — existing offload condition implementations may need to be revised.

WAS THIS USEFUL? 👍 👎

lexmount Moli

Feedback ⓘ v1.1.24 NOTES ↗ RANK

Moli is a best headless browser for AI agents. Lite, Fast, High-Compatibility. Built in Rust

> Moli v1.1.24 adds typed page tasks and runtime hooks in its ACE layer, transactional IndexedDB store/index renames, extra CSS computed-style field reporting, and CSS stylesheet byte decoding.

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

01 **Extra fields in getComputedStyleForNode** NEW 65

Report extra CSS computed-style fields by using `extraFields` in `getComputedStyleForNode`.

– Names the exact flag and API but no mechanism detail v1.1.24

THINNER COVERAGE BELOW

02 **Typed page tasks and runtime hooks in ACE layer** NEW 35

Moli now exposes typed page tasks and browser runtime hooks in the ACE layer.

– Names the layer but no concrete API or usage path v1.1.24

03 **Transactional IndexedDB store and index renames** NEW 35

Moli adds transactional IndexedDB store/index renames.

– States the capability without mechanism or API names v1.1.24

04 **CSS stylesheet byte decoding** NEW 25

Moli now exposes CSS stylesheet byte decoding.

– Bare description with no named surface or usage detail v1.1.24

WAS THIS USEFUL? 👍 👎

CrewAI

Feedback ⓘ 1.15.41 NOTES ↗ RANK

CrewAI is an open-source Python framework for building multi-agent workflows using role-based AI agent Crews and event-driven Flows.

> CrewAI 1.15.41 lets agents access approved private URL hosts, enabling crews to reach internal services and private endpoints.

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

01 **Approved private URL hosts** NEW 36

Agents can now access approved private URL hosts, allowing crews to reach internal services and private endpoints.

— No config key or setup path given 1.15.41

WAS THIS USEFUL?  

magic-context

Feedback ⓘ v0.49.53 NOTES ↗ RANK

magic-context is a memory system for AI agents that provides unbounded context and manages itself across sessions.

> magic-context's single release in this window adds a plugin version check to its doctor command and introduces a stable dashboard token file.

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

01 **OpenCode 2 plugin version check in doctor** NEW 70

Running `doctor --issue` now warns when the OpenCode 2 plugin is outdated, showing the installed version alongside the CLI version and instructions on how to update.

— Names exact command and behaviour, but no mechanism detail v0.49.53

THINNER COVERAGE BELOW

02 **Stable dashboard token file** NEW 25

magic-context adds a stable dashboard token file, per the release summary, though no further detail on its location or usage is given.

— Mentioned only in summary, no mechanism or path given v0.49.53

WAS THIS USEFUL? 👍 👎

Agno (formerly Phidata)

Feedback ⓘ v3.2.0 NOTES ↗

RANK

Agno is an agent framework and runtime that orchestrates and runs multi-agent systems.

> Agno v3.2.0 adds A2A v1.0 support, a Bolt-based Slack interface for AgentOS, and database-backed AgentOS metrics. It also expands stored external agents, session-run pagination, Company details, Super Agent task execution, and routine execution behavior.

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

01 **Slack Bolt AgentOS interface** BREAKING 95

The Slack interface now runs on Slack Bolt with `slack_bolt>=1.29.0,<2` and `slack_sdk>=3.43.0,<4`, adding streaming replies, a Stop button that cancels runs for agents, teams, workflows, and external agents, and native support for `ClaudeAgent` and `CodexAgent`. AgentOS can map Slack users to AgentOS principals and control run approval with `users=` and `approvers=` on Slack(...), and can serve Super Agent over Slack using Slack(team=agent_os.manager), with Company Task results posted back into the originating thread across restarts. Breaking changes require `pip install -U "agno[slack]"`; only the starter of a run can Approve or Deny it unless listed in `approvers=` or mapped via `users=`, AgentOS refuses startup when Slack serves `agent_os.manager` under AgentOS(authorization=...) or an authorization middleware without `users=`, and direct imports of removed internal Slack helpers `SlackEventHandler`, `HITLHandler`, and `verify_slack_signature` need updating.

Serve Super Agent in Slack with per-user role enforcement and a named approver who can approve or stop any run.

python

```python
from agno.interfaces.slack import Slack
from agno.agent_os import agent_os

slack = Slack(
    team=agent_os.manager,
    users={"U012AB3CD": "alice@example.com", "U098ZY7WX":
"bob@example.com"},
    approvers=["U098ZY7WX"],
)
slack.serve()
```

— Includes args, install command, and constraints v3.2.0

02 **A2A v1.0 protocol support** BREAKING 91

Agno now runs on the official A2A v1.0 protocol using `a2a-sdk[http-server]>=1.2.2,<2`, installed with the `a2a` extra, while standard v0.3 clients still connect through the compatibility wire. A2A tasks are stored in the database and support send, stream, get, cancel, list, and subscribe operations; Agent Cards advertise both v1.0 and v0.3 interfaces. Breaking changes remove the A2A `/v1/message:*` and `/v1/tasks:*` routes, require use of each entity's A2A endpoint instead, change stream events and response artifacts to the v1.0 shape, and require updating direct imports of removed A2A helpers.

— Names package constraint and removed routes v3.2.0

03 **Session-run cursor pagination** NEW 90

`GET /sessions/{session_id}/runs` can now retrieve session runs with cursor pagination using `limit`, `before_run_index`, and `after_run_index`. Calls without `limit` still return a bare list.

Page through runs in a long session without loading all of them at once — useful when inspecting a session with hundreds of historical runs.

```
$ curl -X GET 'https://<your-agentOS-host>/sessions/<session_id>/runs?limit=20&after_run_index=40' \
  -H 'Authorization: Bearer <token>'
```

— Exact endpoint and query parameters v3.2.0

04 **AgentOS metrics endpoints** NEW 86

AgentOS homepage metrics are now available through new `GET /os/metrics/*` endpoints, backed by new component, aggregate, and refresh tables. The metrics cover daily sessions, runs, tokens, users, models, cost, duration, time-to-first-token, and model latency.

Pull the AgentOS homepage metrics for the last 30 days to monitor token usage and model latency across the platform.

```
$ curl -X GET 'https://<your-agentOS-host>/os/metrics/aggregate?days=30' \
  -H 'Authorization: Bearer <token>'
```

— Endpoint family and metric set named v3.2.0

05 **OSManager default and tool-name changes** BREAKING 81

`OSManager.tool_call_limit` now defaults to 24 instead of 8, and `num_history_runs` now defaults to 5 instead of 7. AgentOS refuses startup when OSManager(root=...) is combined with custom tools named `read_file`, `list_files`, or `search_content`.

— Defaults and invalid names specified `v3.2.0`

06 **Super Agent direct execution** IMPROVED 66

Super Agent now executes work directly instead of routing through manager members. It can list and run agents, teams, and workflows, read platform activity and schedules, and use Company, docs, source-file, and custom OS Manager tools; Tasks now expose `last_run` and API-computed `allowed_transitions`, support direct start, and record durable attempt state so Super Agent can start an accepted Task on the durable queue without blocking the conversation.

— Mechanism described, limited usage surface `v3.2.0`

THINNER COVERAGE BELOW

07 **Stored Claude and Codex agents** NEW 55

`ClaudeAgent` and `CodexAgent` can now be saved, loaded, versioned, listed, and run as first-class stored components through the components API.

— API area named, no endpoint details `v3.2.0`

08 **Company details endpoint** NEW 54

Agno adds a new `/company/details` endpoint to set a Company name and description. The same Company name and description can also be set from config or from chat.

— Endpoint named, method absent `v3.2.0`

09 **Routine Mini Agent execution** IMPROVED 47

Routines now run on a dedicated Mini Agent with Company tools and packaged Routine skills. The Mini Agent is hidden from agent discovery, MCP, and A2A.

— Behavior explained without usage surface `v3.2.0`

└──▷ **BREAKING ON UPGRADE**

! `OSManager.tool_call_limit` now defaults to 24 (was 8) and `num_history_runs` defaults to 5 (was 7); AgentOS refuses startup when OSManager(root=...) is combined with custom tools named `read_file`, `list_files`, or `search_content`.

! The A2A `/v1/message:*` and `/v1/tasks:*` routes are removed; use each entity's A2A endpoint instead. Stream events and response artifacts now follow the v1.0 shape. The `a2a` extra now requires `a2a-sdk[http-server]>=1.2.2,<2`. Direct imports of removed A2A helpers need updating.

! The Slack interface now requires `slack_bolt>=1.29.0,<2` and `slack_sdk>=3.43.0,<4`; run `pip install -U "agno[slack]"` to upgrade. Only the person who started a run can Approve or Deny it unless they are listed in `approvers=` or mapped via `users=`. AgentOS refuses startup when Slack serves `agent_os.manager` under AgentOS(authorization=...) or an authorization middleware without `users=`. Direct imports of removed internal Slack helpers (`SlackEventHandler`, `HITLHandler`, `verify_slack_signature`) need updating.

WAS THIS USEFUL? 👍 👎

LangChain LangGraph

Feedback ⓘ `cli==0.4.43` NOTES ↗

RANK

LangGraph is a low-level orchestration framework for building, managing, and deploying long-running, stateful AI agents.

> LangGraph's CLI gained support for Agent Server flags, with the older deployment flag names kept as hidden aliases for backward compatibility.

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

01 **Agent Server flags in LangGraph CLI** IMPROVED 30

The LangGraph CLI now supports Agent Server flags, with the previous deployment-related flags retained as hidden aliases for backward compatibility.

— No specific flag names or commands given `cli==0.4.43`

WAS THIS USEFUL? 👍 👎

LangChain Deep Agents

Feedback ⓘ deepagents==0.7.36

NOTES ↗ RANK

LangChain Deep Agents provides an agent harness with planning, filesystem tools, subagents, and context management.

> LangChain Deep Agents improved middleware and subagent traceability by exposing the agent backend to middleware and recording subagent checkpoint namespaces on task tool messages.

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

01 **Subagent checkpoint namespace on task messages** IMPROVED 55

The subagent's checkpoint namespace is now recorded on the `task` tool message, making subagent state traceable across runs.

— Names concrete `task` tool message field deepagents==0.7.36

02 **Agent backend passed to middleware** IMPROVED 35

Middleware that requests it now receives the agent's backend, enabling backend-aware middleware logic.

— Describes capability but no API/flag name given deepagents==0.7.36

WAS THIS USEFUL? 👍 👎

openexecutive

Feedback ⓘ v0.7.3 NOTES ↗ RANK

Openexecutive is an ai-powered virtual executive team — a single coherent executive persona backed by 8 specialist agents (FastAPI + Next.js).

> OpenExecutive's v0.7.3 release focuses on tightening approval workflows inside chat and email, adding in-card actions so pending tasks can be handled without leaving the message.

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

01 **In-chat approvals on Google Chat cards** NEW 40

Google Chat cards now include Approve and Skip buttons, letting users act on pending tasks directly from the card.

— Names the UI surface but no command or API v0.7.3

02 **Consolidated email card controls** IMPROVED 33

Email cards now have one consolidated set of buttons and support answering questions directly from the card.

— Describes change but lacks specifics on buttons or mechanism v0.7.3

WAS THIS USEFUL? 👍 👎

Microsoft Skill Recorder

Feedback ⓘ v0.9.0 NOTES ↗

RANK

Skill Recorder is a desktop app that records your on-screen work session and uses the GitHub Copilot SDK to reconstruct it as an intent + ordered steps, then builds a reusable Skill or Automation for Microsoft Autopilots like Scout, Microsoft Copilot Cowork, or Copilot Studio.

> Skill Recorder's v0.9.0 release adds a full Bring your own model option, letting users power AI analysis, skill generation and automation generation with Microsoft Foundry, Azure OpenAI, OpenAI, Anthropic, or any custom OpenAI-compatible endpoint instead of GitHub/Copilot, backed by provider validation and OS-level credential storage.

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

01 **Secure storage and management of BYOM credentials** NEW

83

Provider API keys are stored in OS-protected storage (DPAPI on Windows, Keychain on macOS, secure desktop keyring on Ubuntu), falling back to in-memory storage only when secure storage is unavailable. Credentials can be removed at any time via Forget key, and connections can later be changed through AI connection in Sessions.

— Names exact storage mechanisms and UI controls for managing keys v0.9.0

02 **Bring your own model AI provider support** NEW

78

At first use, users choose between Bring your own model or GitHub authentication (recording remains available without either). BYOM connects to Microsoft Foundry/Azure OpenAI, OpenAI, Anthropic, or any custom OpenAI-compatible provider by configuring an API key, endpoint, and model or deployment name, with no GitHub login or Copilot subscription required. The selected provider is applied consistently across analysis, skill generation, and automation generation, with provider-aware data-destination disclosures and isolation of BYOM sessions from unrelated CLI configuration.

— Names providers and config fields but not exact UI flow steps v0.9.0

03 **Provider configuration validation before saving** NEW

65

A Test and save synthetic check verifies image and tool-calling support for a configured provider before the configuration is saved, with the check billed to the provider.

— Names the exact check and what it validates, not full flow v0.9.0

WAS THIS USEFUL? 👍 👎

thclaws

Feedback ⓘ v0.155.0 NOTES ↗ RANK

Thclaws is an open-source AI agent harness in native Rust — GUI, CLI, headless, and webapp from one binary. Multi-provider, MCP, skills, plugins, agent teams.

> thClaws added Keenable as a new web search backend option.

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

01 **Keenable backend for web search** NEW 35

thClaws adds Keenable as a backend option for the websearch feature.

— Names the backend but no config or usage detail v0.155.0

WAS THIS USEFUL? 👍 👎

LangGenius Dify

Feedback ⓘ CVE-2025-59422 CVE ↗

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

.
.
.
.

`CVE-2025-59422`: Broken Access Control on Log Message Endpoint Allows Reading Chats of Others (affects `1.8.1`)

WAS THIS USEFUL? 👍 👎

◆ MCP Servers & Tooling

canvas-mcp

Feedback ⓘ v1.18.0 NOTES ↗ RANK

Canvas LMS MCP server — up to 102 tools and 8 agent skills for students & educators. Works with Claude, Cursor, Codex, and 40+ agents.

> canvas-mcp v1.18.0 ships a sandbox network guard that restricts outbound sandbox connections to an explicit host allowlist, a breaking change for workloads relying on unrestricted egress.

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

01 Sandbox network guard with host allowlist

BREAKING

50

Every outbound connection from a sandbox workload must now target an allowlisted host; connections to non-allowlisted hosts are blocked, reducing lateral-movement risk from compromised sandboxes. This is a breaking change — sandboxes that previously made arbitrary outbound connections will have those connections blocked on upgrade.

– Explains mechanism and impact but no config key or flag to adjust allowlist v1.18.0

└──▷ BREAKING ON UPGRADE

! The sandbox network guard now requires every outbound connection to target an allowlisted host — sandboxes that previously made arbitrary outbound connections will have those connections blocked on upgrade.

WAS THIS USEFUL? 👍 👎

agentcad

Feedback ⓘ v0.8.0 NOTES ↗ RANK

Agentcad is a CAD CLI and MCP server for AI agents

> agentcad v0.8.0 introduces compact response snapshots with on-demand full-response retrieval, configurable cleanup of cached derived artifacts, removes the experimental gate from `inspect`, adds opt-in staged-change commands, and switches to a new daemon state directory that requires manually stopping old services before upgrading.

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

01 **Experimental staged-change commands** NEW 90

Experimental staged-change commands can be enabled by passing `--experimental-changes` on each CLI call or setting `experimental_changes: true` on each local MCP call, including `run --idempotency-key` and `run --status`.

Run experimental staged-change commands across a CI pipeline where each call must opt in explicitly.

```
$ agentcad run model.py --label first --experimental-changes --idempotency-key build-001
```

— Exact flags and subcommands with a runnable example. v0.8.0

02 **Compact response snapshots with full retrieval** NEW 88

Import and `inspect --report` replies now default to a compact summary with selected images, capped at 2,000 estimated text tokens for large parts. The complete saved response — PNGs, the full face-and-edge list, reports, and the normalized STEP — can be fetched through the CLI or local MCP connection using `response_snapshot.retrieve`, or by passing `--full-response` (CLI) or `full_response=True` (MCP) to get the previous complete response shape instead of the compact summary.

Get a compact summary on import and then fetch the complete saved response when the agent needs full detail.

```
$ agentcad import model.step
# agent follows response_snapshot.retrieve action to read the complete saved response
agentcad import model.step --full-response
```

— Names exact token limit, action, and flags; lacks retrieval cost detail. v0.8.0

03 **New daemon state directory requires manual stop** BREAKING

80

The new daemon uses a private state directory and does not discover or migrate old shared temporary paths; services must be stopped with `agentcad daemon stop` and `agentcad viewer stop` before upgrading.

— Exact migration commands given, clear before/after behavior. v0.8.0

04 **Configurable cleanup of derived artifacts** NEW

70

Cleanup of old derived reports and images can now be configured while CAD geometry, issued IDs, retry records, and resumable checkpoints are retained, with a per-build-root cache defaulting to 512 MiB and 30 days.

— Gives concrete defaults but no config key or flag name. v0.8.0

05 **Inspect gate removed, artifacts default-on** IMPROVED

65

`inspect` can now be used without an extra acknowledgment step — the post-0.7.2 experimental gate is removed, so import reports, IDs, images, jobs, and artifact retrieval are available by default.

— Names removed gate and newly default surfaces, no further mechanism. v0.8.0

THINNER COVERAGE BELOW

06 **Local feedback report before upload** NEW

35

A longer feedback report can be written to a file or sent through standard input to keep the full report locally before upload.

— No command or flag name given, thin description. v0.8.0

└──▷ **BREAKING ON UPGRADE**

! The new daemon uses a private state directory and does not discover or migrate old shared temporary paths — stop services with `agentcad daemon stop` and `agentcad viewer stop` before upgrading.

WAS THIS USEFUL? 👍 👎

gflow-cli

Feedback ⓘ v0.98.0 NOTES ↗ RANK

Gflow-cli is a command-line tool for Google Flow: generate Veo videos and Imagen images via script or batch. Ships an MCP server for coding agent integration.

> gflow-cli's v0.98.0 release adds model discovery commands, a capability-reporting surface exposed to both CLI and MCP clients, a new optimization skill, and richer error payloads for billing-aware error handling.

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

01 **Model listing and inspection commands** NEW 75

Adds `models list` to enumerate available models and `models show <id>` to inspect a specific model's details before using it in a generation.

Find the identifier of the model you want to target before passing it to a generation command.

```
$ gflow models list
```

Inspect details of a specific model by ID before using it in a generation.

```
$ gflow models show <id>
```

— Named commands with example usage, limited mechanism detail v0.98.0

02 **Capability discovery via CLI and MCP** NEW 75

The new `gflow capabilities` command reports yes/no/partial status per capability for the current session without calling Flow, and the same information is exposed to MCP clients via a new `capabilities://self` resource endpoint.

Check what generation surfaces are available before running a batch, without hitting Flow at all.

```
$ gflow capabilities
```

— Names both surfaces and explains behaviour, no further mechanism v0.98.0

THINNER COVERAGE BELOW

03 **Richer error fields in problem details** IMPROVED 55

Error responses now include `charged`, `next_action`, and `operation_id` fields in problem details, helping callers understand billing state and what to do next.

— Names exact fields and purpose but no usage example v0.98.0

04 **/gflow:optimize engineering skill** NEW 35

Adds `/gflow:optimize` as a new engineering skill alongside the existing plan and issue-resolve disciplines.

— Named but undescribed beyond its placement among skills v0.98.0

WAS THIS USEFUL? 👍 👎

mcp-server

Feedback ⓘ v2.6.7 NOTES ↗ RANK

BrowserStack's Official MCP Server

> BrowserStack's MCP server now remembers confirmed false positives in accessibility scans and adds a guided remediation prompt, while flagging an older accessibility tool for removal.

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

01 **False-positive memory and remediation prompt for accessibility reviews**

NEW 60

Accessibility scanning now remembers false positives a user confirms, so they are not re-reported in subsequent runs. A new `fix-accessibility-issues` prompt guides remediation of detected accessibility problems.

– Names the prompt but no usage example given v2.6.7

THINNER COVERAGE BELOW

02 **accessibilityExpert tool deprecated** DEPRECATED 40

The `accessibilityExpert` tool is deprecated and may be removed in a future release.

– Names the tool but no migration path given v2.6.7

└──▷ **BREAKING ON UPGRADE**

! The `accessibilityExpert` tool is deprecated and may be removed in a future release.

WAS THIS USEFUL? 👍 👎

context7

Feedback ⓘ @upstash/context7-mcp@4.7.0 NOTES ↗ RANK

Context7 maintains up-to-date documentation and context for AI agents through an MCP server.

> Context7 added pagination to its upgrade-guide tool so AI agents can consume long migration guides in structured sections.

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

01 **Paginated upgrade guides via `page` argument** NEW 80

The `get-upgrade-guide` tool now accepts a `page` argument so long migration guides load in structured, paginated sections. Page 1 always leads with summary, prerequisites, codemods, breaking changes, and deprecations, while subsequent pages cover behavior, config, and API changes.

— Names exact argument and page structure, no example shown @upstash/context7-mcp@4.7.0

WAS THIS USEFUL? 👍 👎

morluto REA

Feedback ⓘ rea-agents-6.9.0 NOTES ↗

REA is a reverse engineer anything with agents, from app behavior down to native binaries.

REA adds desktop task capture in build computers, paged Evidence results, offline Evidence record access, guided CLI/MCP workflows, and CSV/TSV analysis views.

└──▷ GET THIS VERSION ▾

npm

```
$ npm install -g rea-agents@6.9.0
```

git

```
$ git clone --branch rea-agents-6.9.0
https://github.com/morluto/rea.git
# already have the repo? check out this version:
$ git checkout rea-agents-6.9.0
```

signed build provenance · checked 13 Nov Wrong command?

Capture, export and witness desktop tasks inside a build computer; keep an agentenv computer available through MCP and prepare a build host from one command.

Install desktop apps from SHA-256-pinned AppImages or upstream archives, retain selected package dependencies, and isolate explicitly declared build secrets.

Choose seed/end checkpoints, open seed documents at launch, build several tasks from one capture, and evaluate unreadable app state through headless checks or a screen judge.

Read gzip and XML desktop documents and draft assertions for nested fields, attribute text, child text and member counts.

Replay captured environment writes, including variable member sets, sorted creates, references, toggles and subsequent reads that explain the resulting state.

+4 more

NEW SUPPORTED PLATFORM ReFS volume support for private runtimes

NEW SUPPORTED PLATFORM Offline NTFS MFT and USN journal acquisition

NEW SUPPORTED PLATFORM RISC-V ilp32 and Windows P0 DOS MZ admission

NEW SUPPORTED PLATFORM Linux core-dump, Android ELF and Apple IPS support

NEW DETECTION ELF/PE/Mach-O hardening and encryption reporting

NEW DETECTION V8 code-cache header and Electron cachedData boundary inspection

NEW DETECTION Firmware region inspection: Intel HEX, Motorola S-record, U-Boot, ESP flash, UF2, DfuSe

NEW DETECTION Android R8 mapping, JNI candidate linking and DEX caller tracing

NEW DETECTION IL2CPP v108 header/string Evidence retention

NEW INTEGRATION Playwright v10 trace Evidence import

NEW INTEGRATION OpenAPI and GraphQL SDL inventory with HAR comparison

NEW SUPPORTED PLATFORM New bundle formats: UnityFS, Godot 3 PCK, WeChat mini-program, Zstandard ZIP, iNES

NEW SUPPORTED PLATFORM Standalone executable formats: .NET single-file, Bun standalone, Deno compile

WAS THIS USEFUL? 👍 👎

concord-mcp

Feedback ⓘ v0.22.0 NOTES ↗ RANK

Concord-mcp is a live messaging for coding agents

> Concord-mcp added support for OpenCode as a recognized client in this release.

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

01 **OpenCode client support** NEW 32

concord-mcp now supports OpenCode as a client, identifying OpenCode sessions from MCP request metadata.

— Names the client but no mechanism detail beyond session identification. v0.22.0

WAS THIS USEFUL? 👍 👎

◆ Agent Sandboxes

opensandbox

Feedback ⓘ release-1.2.1 NOTES ↗ RANK

Opensandbox is a secure, Fast, and Extensible Sandbox runtime for AI agents.

> OpenSandbox shipped Credential Vault Exchange for per-request secret resolution across all five SDKs, multi-version Firecracker VMM support, MCP background-command polling, port-scoped egress enforcement, and stricter request validation with 422 rejection of unknown fields, while deprecating the server-proxied sandbox data path in favor of the ingress gateway.

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

01 **Controller deployment tuning options** IMPROVED 95

The controller deployment config gains `controller.extraArgs` for tuning leader-election and other startup parameters, plus namespace-scoped caches and configurable leader-election timings for Kubernetes deployment tuning. Operators can set exact timings via flags such as `--leader-elect-lease-duration=20s`, `--leader-elect-renew-deadline=15s`, and `--leader-elect-retry-period=5s` to tolerate a slow API server and avoid spurious re-elections.

Tune leader-election timing to tolerate a slow API server and avoid spurious re-elections after a 10-second stall.

```
$ opensandbox-controller \
  --leader-elect-lease-duration=20s \
  --leader-elect-renew-deadline=15s \
  --leader-elect-retry-period=5s
```

— Includes a runnable controller command with exact flags release-1.2.1 release-1.2.1-rc.2

04 **Background command polling in MCP server** NEW 75

The Python MCP server now exposes `command_get_status` and `command_get_logs` for polling background commands: `command_get_status` reports running state and exit code, while `command_get_logs` returns content plus an opaque cursor for retrieving incremental output.

— Names exact MCP tool methods and returned fields release-1.2.1 release-1.2.1-rc.2

05 **Credential Vault Exchange across SDKs** BREAKING 75

All five SDKs now support Credential Vault Exchange, resolving credentials per request without persisting resolved secrets in the vault, and expose optional detailed download APIs with status, headers, ranges, and partial-response detection without requiring changes to existing adapters. The underlying `CredentialSource` model is widened to cover both inline and exchange variants, so Kotlin and C# SDK consumers must recompile, and C# code with inline-only assumptions needs review.

— Mechanism and migration note given, no code sample `release-1.2.1`

---

07 **Multi-version Firecracker VMM support** NEW 65

Fast Sandbox now supports multiple Firecracker VMM versions: new template builds select the highest installed version (1.16.1 or 1.17.0), while existing artifacts restore with their exact recorded VMM version and fail explicitly if that version is unavailable.

— Names versions and restore behavior, no direct action step `release-1.2.1`

---

08 **Disk resource limit enforcement for Docker sandboxes** IMPROVED 65

The server now enforces `resourceLimits.disk` on the Docker writable container layer, returning HTTP 400 for unsupported drivers or invalid limits instead of silently ignoring them.

— Names config key and exact error behavior `release-1.2.1`

THINNER COVERAGE BELOW

12 **Execd token discovery and staged uploads** IMPROVED 50

Execd now discovers Jupyter runtime tokens automatically when `JUPYTER_TOKEN` is unset, and staged uploads protect existing content by staging before rename, reducing overwrite data loss where atomic replacement is available.

— Names env var and upload mechanism briefly `release-1.2.1`

---

13 **SHA-256-pinned OCI template artifacts required** BREAKING 40

The server now requires OCI template artifacts to be SHA-256-pinned in the expected repository.

— Thin description of a requirement, no migration steps `release-1.2.1`

---

14 **Docker snapshots from paused sandboxes** NEW 25

Docker snapshots can now be taken directly from paused sandboxes.

— Single-line description with no mechanism `release-1.2.1`

└──▷ BREAKING ON UPGRADE

- ! Kotlin and C# SDK consumers must recompile after the `CredentialSource` model is widened to cover both inline and exchange variants; C# code with inline-only assumptions must be reviewed.
- ! Unknown fields in closed request objects are now rejected with 422 rather than silently discarded — `readonly: true` on a volume and `createIfNotExist: false` on volumes now behave as written; callers sending extra fields to image, auth, platform, volume, OSSFS, credential-proxy, or snapshot-renew request models will receive errors.
- ! The server-proxied data path (`/sandboxes/{id}/proxy/{port}` , the `use_server_proxy` parameter, matching SDK options, and CLI/MCP flags) is deprecated into maintenance mode and receives security and bug fixes only — sandbox traffic must be routed through the ingress gateway.

WAS THIS USEFUL? 👍 👎

◆ Agent Memory

claude-mem

Feedback ⓘ v15.18.3 NOTES ↗ RANK

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

> claude-mem v15.18.3 added token protection for its local worker API and MCP HTTP transport, plus Codex hook health/install support. It also shipped an n8n workflow integration and clearer quota fallback warnings.

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

01 **Local API token protection** NEW 96

Set `CLAUDE_MEM_LOCAL_API_TOKEN` in `~/.claude-mem/settings.json` to protect the worker API and MCP HTTP transport on shared machines. After restart, both services reject requests missing the token via `X-Claude-Mem-Token` or `Authorization: Bearer`, and the viewer accepts it as `/?token=...` or `/viewer?token=...`.

Lock the worker API and MCP HTTP transport to a secret token on a machine shared by multiple OS users.

```json
CLAUDE_MEM_LOCAL_API_TOKEN
```

— Exact config key, file, headers, and viewer URLs v15.18.3

02 **Codex hook registration support** IMPROVED 92

`npx claude-mem doctor` now includes a Codex hooks row that warns when the plugin is enabled but its hooks are absent from `hooks.json`. `npx claude-mem install` can register claude-mem's Codex hooks into `~/.codex/hooks.json` by selecting Codex, so hook approval survives future updates.

Verify that Codex hooks are correctly registered after updating, so session capture does not silently stop.

```
$ npx claude-mem doctor
```

Re-register Codex hooks into your own hooks.json once so upgrades no longer require re-approval in Codex.

```
$ npx claude-mem install
```

— Runnable commands and exact hook files included v15.18.3

03 **Quota fallback warning details** IMPROVED 68

The quota fallback warning now names both the selected provider and the provider actually doing the work. It also names `CLAUDE_MEM_QUOTA_FALLBACK_PROVIDER=off` to disable fallback.

— Includes exact environment variable setting v15.18.3

THINNER COVERAGE BELOW

04 **n8n workflow integration** NEW 50

The n8n workflow integration now saves a note, confirms the save, then searches and retrieves it within a chosen project. It is imported via the new n8n docs page.

— Workflow behavior named, but no URL or config v15.18.3

WAS THIS USEFUL? 👍 👎

Volcengine OpenViking

Feedback ⓘ python-sdk@0.1.20

NOTES ↗ RANK

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

> OpenViking's Python SDK gained a batch tag-update method for contexts, available in both sync and async clients.

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

01 **Batch context tag updates via batch_set_tags** NEW 72

The Python SDK adds a `batch_set_tags` method, available in both sync and async clients, to update multiple context tags—such as usage count and last-used-time—in a single call, replacing per-tag round-trips.

— Names exact method and sync/async scope, no usage example given python-sdk@0.1.20

WAS THIS USEFUL? 👍 👎

◆ Browser & Computer Use

browser-use

Feedback ⓘ 1.0.3 NOTES ↗ RANK

The browser-use Python library automates web browsers for AI agents.

> browser-use 1.0.3 improves CLI browser management with a persistent Chromium restart flag, container/PID-namespace connectivity, and faster failure on misconfigured remote debugging URLs.

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

01 CLI-managed persistent Chromium with --reload NEW

77

The CLI now automatically starts a persistent Chromium browser when no remote-debugging browser or display is available (as in `--cli-mcp` mode). Run `browser-use --reload` to close and restart this persistent browser on demand.

Restart the CLI's persistent Chromium after changing browser settings or recovering from a bad state.

```
$ browser-use --reload
```

— Exact command given and behaviour explained 1.0.3

02 Fast-fail on misconfigured BU_CDP_URL IMPROVED

65

A misconfigured `BU_CDP_URL` that nothing listens on now fails in 3 seconds with a message naming the value and both remediation paths, instead of the previous 30-second wait.

— Named env var and concrete before/after timing 1.0.3

THINNER COVERAGE BELOW

03 Connect to Chrome via DevToolsActivePort in other PID namespaces

NEW

50

browser-use can now connect to a Chrome instance running in another PID namespace — such as containers or Cursor on Linux — via its DevToolsActivePort file.

— Mechanism named but no exact command shown 1.0.3

WAS THIS USEFUL?  

windows-mcp

Feedback ⓘ v0.8.9 NOTES ↗ RANK

MCP Server for Computer Use in Windows

> Windows-MCP's v0.8.9 release adds desktop control gating with a confirmation prompt before AI takes over, a way to export Snapshot screenshots directly to disk, and a significant Snapshot performance fix, alongside smaller UI-tree and safety improvements.

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

01 **Snapshot screenshot export via save_path** NEW 70

The Snapshot tool can now write its screenshot directly to disk using the new `save_path` parameter, saving the output as a PNG file instead of returning base64 data.

Persist a Snapshot screenshot to a PNG for later review or offline analysis without piping base64 output.

```json
{ "save_path": "C:\\Users\\analyst\\screenshots\\snapshot.png" }
```

— Names exact parameter and shows a runnable usage example v0.8.9

THINNER COVERAGE BELOW

02 **Snapshot collection performance fix** IMPROVED 55

Snapshot collection time was cut from 15 s to 0.6 s by skipping collection of control handles for the server's own windows.

— Concrete before/after metric but nothing for reader to invoke v0.8.9

03 **Desktop control gating and idle timeout** NEW 50

New desktop gate policies require a confirmation prompt before AI starts a new desktop control period, and desktop control can now be made optional with a configurable idle timeout.

— Describes the gating mechanism but no config key or flag given v0.8.9

04 **Word node merging in UI tree** NEW 40

The UI tree can now merge per-glyph Word nodes into line runs via a new optional setting in feat(tree).

— Names the setting area but not its exact key v0.8.9

05 **AI click tagging for auditability** NEW 35

The cursor glow is now tagged with `AI` and each AI-driven click is visually marked, improving auditability of automated input.

— Explains the visual change but no UI path or setting named v0.8.9

06 **Safer symlink and junction deletion** IMPROVED 35

Deleting symbolic links and junctions no longer removes their underlying targets.

— Clear before/after behaviour but no command shown v0.8.9

07 **Region-filtered scope reporting in Snapshot** IMPROVED 30

Snapshot responses now report the region-filtered UI tree scope, so callers know which portion of the tree was returned.

— Thin description, no field name or example given v0.8.9

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ ML Platforms & SDKs

AWS Amazon Bedrock AgentCore

Feedback ⓘ

Runtime: VPC mode now supports use1-az6 on platform version V2 (2026-11-13)

BLOG ↗ RANK

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

> Amazon Bedrock AgentCore Runtime expanded its V2 agent platform this window with lifecycle hooks for faster cold starts, much larger container images, committed burst-capacity resource configurations, and a new supported Availability Zone for VPC mode.

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

01 **Committed resource configurations with burst capacity** NEW

70

V2 agents now support committed baseline vCPU and memory reservations that can burst up to 4 times the baseline, capped at 8 vCPU and 32 GB of memory per session, billed at a lower rate than default consumption-based pricing.

— Concrete limits and pricing behaviour but no config key or command shown

Runtime: Resource configurations for V2…

THINNER COVERAGE BELOW

02 **Build warmup lifecycle hook for V2 agents** NEW

58

AgentCore Runtime now supports lifecycle hooks — endpoints in your container that Runtime calls at points in the agent runtime's lifecycle — for agents on platform version V2. The first hook, build warmup, lets your agent prepare memory and disk while Runtime prepares the agent so that sessions cold start faster.

— Explains mechanism and benefit but no exact hook endpoint name given

Runtime: Build warmup lifecycle hook fo…

03 **Container images up to 10 GB for V2 agents** IMPROVED

52

V2 agents now support container images up to 10 GB (compressed), up from 2 GB; images larger than 2 GB require a resource configuration.

— Clear before/after numbers but no setup steps given

Runtime: Container images up to 10 GB f…

04 **VPC mode support for use1-az6** NEW SUPPORTED PLATFORM NEW

Amazon Bedrock AgentCore Runtime in VPC mode now supports Availability Zone `use1-az6` in US East (N. Virginia) for agents on platform version V2.

— Names exact AZ and region but is a thin single-line addition

Runtime: VPC mode now supports use1-az6…

> Runtime: Platform version V2 in nine more Regions (2026-11-12) BLOG ↗
>
> STABLE

WAS THIS USEFUL? 👍 👎

Google Gen AI JavaScript SDK

Feedback ⓘ v3.3.0

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 v3.3.0 adds an extra-high thinking depth option and a new Gemini model to its typed enums, along with smaller enum completeness fixes.

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

01 **XHIGH thinking level for generation requests** NEW 75

Adds `XHIGH` and `xhigh` to the `ThinkingLevel` enum so callers can request extra-high thinking depth from supported models via `thinkingConfig: { thinkingLevel: ThinkingLevel.XHIGH }` in `generateContent` calls.

Request maximum thinking depth when generating content by setting `ThinkingLevel` to `XHIGH`.

typescript

```typescript
import {GoogleGenAI, ThinkingLevel} from '@google/genai';
const ai = new GoogleGenAI({apiKey: process.env.GEMINI_API_KEY});
const response = await ai.models.generateContent({
  model: 'gemini-2.5-pro',
  contents: 'Solve this step by step: ...',
  config: { thinkingConfig: { thinkingLevel: ThinkingLevel.XHIGH
} },
});
console.log(response.text);
```

— Names exact enum and shows runnable config usage v3.3.0

02 **gemini-4-hydrogen model added to Model enum** NEW 67

Adds `gemini-4-hydrogen` to the `Model` enum so it can be selected for generation using `Model.GEMINI_4_HYDROGEN` instead of a raw string, e.g. ai.models.generateContent({ model: Model.GEMINI_4_HYDROGEN, ... }).

Target the new `gemini-4-hydrogen` model using the typed `Model` enum instead of a raw string.

```typescript
import {GoogleGenAI, Model} from '@google/genai';
const ai = new GoogleGenAI({apiKey: process.env.GEMINI_API_KEY});
const response = await ai.models.generateContent({
  model: Model.GEMINI_4_HYDROGEN,
  contents: 'Summarize the following text: ...',
});
console.log(response.text);
```

— Names new model and exact typed usage v3.3.0

THINNER COVERAGE BELOW

03 **Additional FinishReason and ThinkingLevel enum values** NEW

28

Adds missing enum values to `FinishReason` and `ReinforcementTuningThinkingLevel`.

— Names enums but no detail on which values or usage v3.3.0

WAS THIS USEFUL?  

Google Gen AI Python SDK

Feedback ⓘ v3.3.0 NOTES ↗

RANK

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

> Google Gen AI Python SDK v3.3.0 adds a new maximum thinking intensity level and expands enum coverage for responses and model selection.

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

01 XHIGH thinking level for generation requests NEW

75

Pass `ThinkingLevel.XHIGH` or `ThinkingLevel.xhigh` in `ThinkingConfig` when configuring thinking in generation requests to request the maximum thinking intensity from supported models.

Request the highest thinking budget when you need maximum reasoning depth from a model that supports it.

python

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

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

response = client.models.generate_content(
    model='gemini-2.5-pro',
    contents='Solve this multi-step reasoning problem: ...',
    config=types.GenerateContentConfig(
        thinking_config=types.ThinkingConfig(
            thinking_level=types.ThinkingLevel.XHIGH
        )
    )
)
print(response.text)
```

— Named config value with a runnable example, but no mechanism detail. v3.3.0

THINNER COVERAGE BELOW

02 Additional enum values for finish reason and tuning IMPROVED

30

The SDK now includes missing enum values in `FinishReason` and `ReinforcementTuningThinkingLevel` for more complete response introspection.

— Names two enums but not which values were added. v3.3.0

03 **gemini-4-hydrogen model enum entry** NEW 30

Adds `gemini-4-hydrogen` to the Model enum for use in interactions EAP builds.

— Names the model value but gives no usage context. v3.3.0

WAS THIS USEFUL? 👍 👎

OpenAI Python SDK

Feedback ⓘ v3.50.0 NOTES ↗ RANK

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

> OpenAI Python SDK v3.50.0 adds support for reasoning input transformations on the Responses API.

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

01 **Reasoning input transformations on Responses API** NEW 40

The SDK now exposes reasoning input transformations on responses via `client.responses.create`.

— Names the exact method but no mechanism or scope detail v3.50.0

WAS THIS USEFUL? 👍 👎

◆ Inference Providers & GPU Clouds

Crusoe Cloud

Feedback ⓘ updated · Nov 13, 2026 NOTES ↗ RANK

Crusoe Cloud provides GPU infrastructure and managed inference for AI training and deployment.

> Crusoe Cloud's Terraform provider gained new resources for external load balancers and Kubernetes node-pool autoscaling, plus safer in-place credential rotation for pull-through cache registries.

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

01 In-place pull-through cache credential rotation

BREAKING

91

Pull-through cache credentials can be rotated in place on `crusoe_registry_repository` by changing `credentials_version`; `upstream_registry_credentials.username` and `.password` are write-only and never stored in plan or state (requires Terraform 1.11 or later). Separately, changing `upstream_registry.provider`, `upstream_registry.url`, or `project_id` now replaces the repository instead of failing, and existing state is upgraded with the previously stored `username` and `password` dropped from it.

Rotate pull-through cache credentials without storing secrets in Terraform state by bumping `credentials_version`.

hcl

```hcl
resource "crusoe_registry_repository" "my_cache" {
  # ... other required fields ...

  upstream_registry_credentials {
    username           = var.registry_username
    password           = var.registry_password
    credentials_version = "v2"
  }
}
```

— Names exact fields, version requirement and state migration behavior snapshot-20261113

02 Terraform resource for external load balancers

NEW

90

The new `crusoe_load_balancer` Terraform resource manages external load balancers by specifying `vpc_id`, `protocol` (`tcp` or `udp`),

`listen_ports_and_backends` (each `listen_port` paired with `backends` of `ip` and `port`), optional `health_check_options`, and exposes a read-only `vip`.

Define an external TCP load balancer fronting two backend VMs in a VPC.

```hcl
resource "crusoe_load_balancer" "web" {
  vpc_id   = crusoe_vpc.main.id
  protocol = "tcp"

  listen_ports_and_backends {
    listen_port = 443
    backends {
      ip   = "10.0.1.10"
      port = 8080
    }
    backends {
      ip   = "10.0.1.11"
      port = 8080
    }
  }

  health_check_options {
    port = 8080
  }
}
```

— Full field list and runnable HCL example given `snapshot-20261113`

---

03 **Kubernetes node pool autoscaling via Terraform** NEW

87

Node pools can now autoscale by adding an `autoscaling_config` block (`enabled`, `min_node_size`, `max_node_size`) to `crusoe_kubernetes_node_pool`, letting the cluster autoscaler size the pool within those bounds. Note: a node pool whose autoscaling was enabled outside Terraform with no `autoscaling_config` block in configuration will have autoscaling disabled on the next `apply`.

Enable autoscaling on a Kubernetes node pool so the cluster autoscaler can grow or shrink it between 2 and 10 nodes.

```hcl
resource "crusoe_kubernetes_node_pool" "workers" {
  # ... other required fields ...

  autoscaling_config {
    enabled       = true
    min_node_size = 2
    max_node_size = 10
  }

  lifecycle {
    ignore_changes = [instance_count]
  }
}
```

— Exact config block plus behavioral caveat on apply `snapshot-20261113`

└──▷ BREAKING ON UPGRADE

! In `crusoe_registry_repository`, changing `upstream_registry.provider`, `upstream_registry.url`, or `project_id` now replaces the repository instead of failing; existing state is upgraded and the previously stored `username` and `password` are dropped from it.

! A `crusoe_kubernetes_node_pool` whose autoscaling was enabled outside Terraform with no `autoscaling_config` block in the configuration will have autoscaling disabled on the next `apply`.

WAS THIS USEFUL? 👍 👎

Eigen Labs Darkbloom

Feedback ⓘ v0.10.12 NOTES ↗

CODE ↗ RANK

Darkbloom is a private Inference Network on Idle Macs

> Darkbloom changed startup and serving modes, adding a personal-only `darkbloom local` path while making `darkbloom start` and `darkbloom cluster` non-interactive by default. It also shipped typed refusal codes, idle TTFT probe controls, broader two-Mac pair resilience, larger Gemma 4 contexts, higher tensor-parallel concurrency, and automatic selected-model reloads.

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

01 **Non-interactive serving startup** BREAKING 93

`darkbloom start` no longer asks how the Mac serves; on a Mac with no saved choice it defaults to both the Darkbloom network and local API, saving `[local] mode = "both"` and `me_first = false`. `darkbloom cluster` also no longer asks how a new pair serves; new setups always use the Darkbloom network with the leading Mac's local API and never pause, while a stopped pair restarts in its last saved mode.

— Exact commands, defaults, saved config keys v0.10.12

02 **Idle TTFT probe control** NEW 90

The idle speed-reprobing behavior can be disabled per model with `DARKBLOOM_TTFT_REFUSAL_PROBE=off`. The probe runs at most once every 2 minutes and never runs on battery.

Suppress the idle TTFT speed-reprobing on a Mac where you want to avoid any extra inference during quiet periods.

```
$ DARKBLOOM_TTFT_REFUSAL_PROBE=off darkbloom start
```

— Environment variable, limits, runnable example v0.10.12

03 **Typed refusal codes** NEW 84

Macs now return typed refusal codes: `media_memory_unavailable` when an image or video cannot be held, `pair_api_unavailable` with `Retry-After: 10` when a pair API is starting or stopping, `unsupported_media` for unsupported image formats, and `request_exceeds_node_budget` when one Mac of a pair runs short of memory mid-request.

— Concrete codes and header, no endpoint v0.10.12

04 **Status shows active work** IMPROVED 74

`darkbloom status` now shows answers in progress, which early-ended requests are paid, and whether fan control is on. It also lists answers in progress on a Mac that serves only you.

— Exact command and displayed fields v0.10.12

05 **Qwen presence-penalty defaults** BREAKING 74

Qwen3.5-9B no longer has a default presence penalty; the 35B Qwen builds keep the card's 1.5. A request's own `presence_penalty` still overrides.

— Exact field and changed defaults v0.10.12

06 **Personal-only local serving** NEW 72

Darkbloom added `darkbloom local` to serve only yourself without being prompted for a serving mode.

Start the provider for personal use only, without being prompted for a serving mode.

```
$ darkbloom local
```

— Runnable command but thin behavior detail v0.10.12

THINNER COVERAGE BELOW

07 **Two-Mac memory resilience** IMPROVED 58

Two-Mac pairs now keep serving when one Mac runs short of memory mid-request. Only that single request ends, and the pair continues.

— Clear before/after, no user surface v0.10.12

08 **Automatic selected-model reloads** IMPROVED 52

Selected models that did not load at start, or were unloaded by Autopilot, now reload automatically when memory frees without evicting other selected models.

— Behavior described, no command or config v0.10.12

09 **GLM tensor-parallel concurrency** IMPROVED 50

GLM-5.3-Flash tensor-parallel pairs now serve up to 8, 6, 4, or 2 requests depending on available memory.

— Specific model and counts, thin mechanism v0.10.12

└──▷ BREAKING ON UPGRADE

! `darkbloom start` no longer asks how this Mac serves; on a Mac with no saved choice it now defaults to serving both the Darkbloom network and the local API, saving `[local] mode = "both"` and `me_first = false`. Scripts that relied on the interactive prompt will take this default silently.

! `darkbloom cluster` no longer asks how the pair serves for new setups (always Darkbloom network with the leading Mac's local API, never pausing); a stopped pair restarts in its last saved mode.

! Qwen3.5-9B no longer has a default presence penalty; the 35B Qwen builds keep the card's 1.5. A request's own `presence_penalty` still overrides.

WAS THIS USEFUL?  

Cloudflare Workers AI

Feedback ⓘ

Workers AI - EmbeddingGemma 2 now available on Workers AI (2026-11-12) BLOG ↗

STABLE

Cloudflare Workers AI runs open-weight models, including the Clef decision models, as serverless inference on Cloudflare's global network.

Cloudflare Workers AI now supports multimodal embeddings via `@cf/google/embeddinggemma-2`, accepting text, images, video, and audio in one call with configurable output dimensions.

NEW INTEGRATION Run multimodal embeddings over text (including 100+ languages and code), images, video, and audio in a single call using `@cf/google/embeddinggemma-2` via env.AI.run(), with all modalities sharing one vector space.

Pass images, video, and audio as base64 data URIs in `messages` entries using `type: 'image_url'` and `image_url.url` fields alongside `type: 'text'` entries.

Tune embedding quality for specific tasks by adding prefixes such as `task: search result | query: ...` in the `content` string of a message.

Reduce vector storage footprint by setting `dimensions` to 128, 256, or 512 (default is 768) with a small quality trade-off.

Use an 8,192-token context window shared across all content in a single input for long-document or mixed-media embedding.

NEW SUPPORTED PLATFORM Inkling 2 Small model availability on Cloudflare Workers AI

▾ 1 more release in this issue · 2026-11-12

Workers AI - Inkling 2 Small now available on Workers AI (2026-11-12) BLOG ↗

LATEST STABLE

Cloudflare Workers AI adds `@cf/thinkingmachines/inkling-2-small`, a 594B multimodal MoE model with vision, configurable reasoning, function calling, and a 1M-token context window.

Call the model via env.AI.run(), the REST API, the OpenAI-compatible endpoint, or AI Gateway using `@cf/thinkingmachines/inkling-2-small`.

Control reasoning depth per request with `reasoning_effort` set to `max`, `medium`, `low`, `minimal`, or `none`; reasoning is on at high effort by default.

Accepts text and image inputs in the same request for vision understanding.

> Provides a 1,048,576-token context window for long-document and multi-turn agentic workflows.
>
> Supports function calling for building agents that invoke tools and APIs across multiple conversation turns.

WAS THIS USEFUL? 👍 👎

◆ Inference Serving & Optimization

emmy

Feedback ⓘ v0.3.25 NOTES ↗ RANK

Emmy is an optimized GPU compiler for LLM inference. Choose from a list of optimized recipes or optimize your own model via kernel fusion, autotuning, and advanced scheduling.

> Emmy shipped a new `deploy` command for renting CloudRift compute, async scheduling for native serving that reaches vLLM parity, and several low-level kernel and compiler-IR optimizations.

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

01 Golden realize drop command NEW

75

The `emmy golden realize --drop` command automatically drops stale freeze rows, useful during a nightly refresh or after a tune to keep the measurement freeze clean.

Drop freeze rows the compiler no longer realizes during a nightly refresh or after a tune, to keep the measurement freeze clean.

```
$ emmy golden realize --drop
```

— Exact command with usage example and purpose v0.3.25

02 Async scheduling for native serving IMPROVED

60

Native serving now supports async scheduling, reducing decode-step host-time gaps on multi-request workloads and reaching parity with vLLM, with the previously executed tuning plan retired.

— Mechanism and benchmark comparison given, no runnable step v0.3.25

03 NVFP4 pair-table lookups in CUDA IMPROVED

60

NVFP4 pair-table lookups now execute in CUDA with packed `e2m1` runs loaded as one word, and the output head runs `W4A4`.

— Precise kernel mechanism but no user-facing action v0.3.25

THINNER COVERAGE BELOW

04 Deploy command for CloudRift compute NEW

55

Adds the `deploy` command to rent cloud compute on a named CloudRift node and instance type.

— Named command but no further mechanism given v0.3.25

05 **Native attention kernel improvements** IMPROVED 55

Native causal attention now folds on tensor cores, covering probabilities, the position mask, and the FP8 value cache; per-key-value-head decode attention now reads the cache once per group and combines a split over its partials.

— Detailed mechanism, but internal and not directly actionable v0.3.25

06 **IR internals rewritten in Rust** IMPROVED 35

Loop IR statements and normalization now run in a Rust crate, with one operation per definition as the normal form; kernel IR statement kinds now declare fields with one decorator that writes their protocol.

— Internal compiler change with no user-facing surface v0.3.25

WAS THIS USEFUL?  

PyTorch ExecuTorch

Feedback ⓘ v1.6.0 NOTES ↗

ExecuTorch runs PyTorch models on mobile, embedded, and edge devices through an ahead-of-time export path and a lightweight C++ runtime.

ExecuTorch adds a torch-free Python runtime for `.pte` files, a `.ptn` portable program format, batched GPU LLM serving with prefix caching and DFlash speculative decoding, and an experimental diffusion extension.

└──▷ GET THIS VERSION ▾

pip

```
$ pip install executorch==1.6.0
# already installed? upgrade to this version:
$ pip install --upgrade executorch==1.6.0
```

git

```
$ git clone --branch v1.6.0 https://github.com/pytorch/executorch.git
# already have the repo? check out this version:
$ git checkout v1.6.0
```

named in the project's README · checked 13 Nov Wrong command?

NEW SUPPORTED PLATFORM Run `.pte` files from NumPy arrays, buffers, or DLPack tensors without a PyTorch installation using the new torch-free Python runtime.

Use the new experimental ExecuTorch Native format (`.ptn`), a portable program lowered at load time by Vulkan or ggml engines.

Change the CUDA stream from Python using the new experimental `executorch.runtime.cuda` module.

Serve multiple LLM sessions on CUDA with batched KV caches, prefix-cache reuse, on-device sampling, and DFlash speculative decoding in CUDA graphs.

Accelerate LLM serving on CUDA and ROCm with new autotuned INT4/5/6/8 quantized GEMM kernels covering decode and small-batch prefill (M ≤ 64).

+14 more

RETIRED SUPPORTED PLATFORM Python 3.10 dropped

RETIRED SUPPORTED PLATFORM CUDA 12.6 and 13.0 wheels dropped

NEW SUPPORTED PLATFORM CUDA 13.4 wheels added

NEW SUPPORTED PLATFORM Cadence HiFi5 support

NEW SUPPORTED PLATFORM WebGPU browser execution

UPDATED SUPPORTED PLATFORM NXP Neutron backend promoted to production-ready

NEW INTEGRATION TensorRT backend in beta

UPDATED INTEGRATION Qualcomm QNN SDK updated to 2.51

UPDATED INTEGRATION NXP eIQ Neutron SDK updated to 3.2.3

NEW INTEGRATION Samsung backend adds Llama 3.2 1B, Gemma 3 1B, and YOLO26

└─▷ **BREAKING ON UPGRADE**

! Python 3.10 is no longer supported; supported versions are 3.11 through 3.15.

! CUDA 12.6 and 13.0 wheels are dropped; Jetson Orin now requires a source build.

! OpenVINO CPU inference now defaults to float32 instead of the previous default.

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

mlxtop

Feedback ⓘ v3.3.0 NOTES ↗ RANK

The mlxtop terminal UI monitors local LLMs on Apple Silicon, showing loaded models, memory use, GPU activity, and generation speed.

> mlxtop's latest releases center on anomaly-based slowdown and stall detection with a replayable session Journal, alongside new runtime adapter discovery (oMLX, llama.cpp, Strata, Jan), KV/prefix-cache metrics, and reworked memory and header displays.

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

01 **Anomaly-based slowdown and stall detection** NEW 90

Detects generation and prefill slowdowns by comparing each sample against earlier samples of similar work (same concurrent-request count and context within a factor of two), labeling slow runs in yellow across charts and panels; the Journal writes a `SLOWDOWN` row when a slowdown begins (naming the speed drop, the unusual signal, and a suggested check), a `RECOVERED` row when it ends, and a `SETTLED` note if the lower speed holds for 30 seconds and becomes the new baseline. Slow prefill is surfaced the same way, e.g. `PREFILL ↓61% (180.0→70.0 tok/s) · paging …`. Generation stalls are named after 1.5s without a new token (Generation stalled), reporting host activity such as `no new token for 2s · GPU idle · disk read 850 MiB/s · 38 SSD cache blocks loaded · SSD cache full`; disk-cache stalls are named from the runtime's storage traffic and oMLX's SSD-cache counters (blocks loaded, saved, evicted, fullness); stall blame is distinguished so a reading only explains a stall if it wasn't already present at normal speed; and advice names the specific reading and fix, e.g. `Wired memory is 31.0 GiB of the 32.0 GiB GPU limit: raise it (sudo sysctl iogpu.wired_limit_mb=…) or shorten the context`. Speculative decoding draft-token acceptance on Splash is shown as `draft 72%` and named as a slowdown reason when it falls, requests that start or end between two polls are skipped to avoid false positives, and the expanded generation chart (Enter) and Diagnostics view (`d`) show the exact thresholds judged against, e.g. `usual 50.0 · slow below 40.0`.

— Full mechanism and named thresholds, but no direct user command v3.3.0

02 **Runtime adapter discovery for oMLX, llama.cpp, Strata and Jan** NEW 90

The `oMLX` adapter discovers connectivity from `~/.config/omlx-coding/server.env` (highest priority), the `OMLX_PORT` environment variable, the default port 8000, and oMLX's `settings.json` (`server.port` , found via `OMLX_BASE_PATH` or the default `~/.omlx`), controlled by the `omx.port` config key

(default 8000). The `llama.cpp` adapter listens on port 9931 for builds from October 2026 (its default since then), falling back to 8080, and reads `predicted_per_second` and `cache_n` from llama-server's own `timings` field when present. A new `strata` adapter is auto-discovered on ports 8080 then 8095 (including a bare `strata serve` on 8095), reads `/metrics?format=prometheus` (switched from `/metrics/prometheus`) and `/v1/models`, and is also selectable with `MLXTOP_PROVIDER=strata`. A new `jan` adapter is auto-discovered on ports 1337 (desktop app) then 6767 (`jan serve`), reads `/v1/models`, and keeps whichever port last answered.

Point mlxtop at a non-default oMLX port without editing settings.json, useful in CI or when running multiple server instances.

```
$ OMLX_PORT=9100 mlxtop
```

— Names every adapter, port, env var and config key involved v3.3.0

03 **Session recording and replay** NEW 85

Record a live session to a file with `--record FILE` (one JSON line per sample) and replay it later with `mlxtop replay FILE` to re-run detection rules and print a full Journal of slowdown and stall findings, without storing prompts or output. Each recorded sample also includes `generating=` and `context=` values so a session replays against its own workload.

Capture a session where a model slows down, then replay it after a rules update to get a fresh Journal without re-running inference.

```
$ mlxtop --record session.jsonl
# later:
mlxtop replay session.jsonl
```

— Exact flag and subcommand with a runnable example v3.3.0

04 **Configurable GPU load warning and critical thresholds** NEW 75

Set GPU load warning and critical thresholds with the new `gpu_warn_load` (default 75) and `gpu_critical_load` (default 90) config keys.

Lower the GPU busy warning to 70% and the critical threshold to 85% so alerts fire earlier on memory-constrained hardware.

```yaml
gpu_warn_load: 70
gpu_critical_load: 85
```

— Named config keys with defaults and a working example `product docs`

05 **KV cache and prefix-cache hit-rate panel** NEW 75

Reads vLLM prefix-cache reuse from `gpu_prefix_cache_*` metrics (vLLM 0.9.1 and earlier) and SGLang cached-prompt-token counters, exposing prefix-cache reuse rates alongside KV occupancy. Adds an SGLang KV cache hit rate shown as cached prompt tokens over all prompt tokens (reported as session averages since SGLang's counters move only when a request finishes), displays the panel — retitled `KV cache / model I/O` — with a hit rate lead (`KV hit 91% · ≈1h 25m saved`) and, where supported, splits hits into RAM (green), SSD (cyan), and computed (dim) bars plus a second bar for the last prompt's hit rate, and adds a `weights` bar for engines that stream MoE experts from disk (such as Splash and Strata) showing the share of decode read from disk and the speed difference (`25% of decode from disk · 18 vs 42 tok/s`).

— Detailed panel mechanics but no command or exact nav `v3.3.0`

06 **Memory, header and status panel display improvements** IMPROVED 75

The memory panel shows an OS `PRESSURE` label, a GPU wired-memory meter with an orange threshold at 90% and red at 97%, a RAM meter broken into wired, app, compressed, and file-cache segments, and a time-trace coloured by pressure level (green/orange/red). It also shows TensorFold model memory on a Mac from the process footprint. Raw rate displays are replaced with plain summaries: the SSD fill bar becomes a session activity summary (`SSD 120.0 GiB · 1.5k blocks loaded · 1.3k saved · 327 evicted`), and the paging panel's rate repeat becomes a plain-language summary (`no paging` , `reading files · memory not tight` , `reading files back: memory is tight` , or `swapping: memory is short`). The header unconditionally shows thermal state and system power draw as `thermal ok · 61 W` (from macOS power telemetry without sudo), and GPU memory in the header is read against the GPU limit named in the memory legend (`iogpu.wired_limit_mb` , or the runtime's recommended working set) rather than the driver's current allocation.

— Names every meter and label but is display-only, no action `v3.3.0`

WAS THIS USEFUL? 👍 👎

◆ AI / LLM Gateways

BerriAI LiteLLM

Feedback ⓘ v1.108.7 NOTES ↗

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

LiteLLM v1.108.7 adds Gemini batch jobs via `/v1/batches`, image search on Bedrock knowledge bases via `query_image`, and gpt-image edits on Azure AI Foundry.

└──▷ GET THIS VERSION ▾

Docker

```
$ docker pull ghcr.io/berriai/litellm:v1.108.7
```

git

```
$ git clone --branch v1.108.7 https://github.com/BerriAI/litellm.git
# already have the repo? check out this version:
$ git checkout v1.108.7
```

published under the project's GitHub account · checked 13 Nov Wrong command?

NEW INTEGRATION Run Gemini batch jobs via `/v1/batches`, download output through `/v1/files`, and get spend tracked against the calling key.

NEW INTEGRATION Search Bedrock knowledge bases with images using the `query_image` field, with image hits included in results.

NEW INTEGRATION Submit gpt-image edits on Azure AI Foundry deployments, including deployments under a custom name.

Responses-only fields such as `max_tool_calls` now return 400 on providers bridged through chat completions unless `drop_params: true` is set.

└──▷ BREAKING ON UPGRADE

! Responses-only fields such as `max_tool_calls` now return 400 when sent to providers bridged through chat completions; set `drop_params: true` or remove those fields to avoid errors.

WAS THIS USEFUL? 👍 👎

Mozilla AI Otari

Feedback ⓘ v0.46.0 NOTES ↗ RANK

Mozilla AI Otari is an open-source AI gateway you run yourself. One endpoint for 40+ providers, with virtual keys, budgets, and usage tracking.

> Mozilla AI Otari added CORS support for direct browser calls to model endpoints, OAuth app connections with PKCE for third-party integrations, and expanded provider model filtering and vLLM catalog compatibility — alongside a breaking change to how disabled API keys are tracked.

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

01 **Model filtering and search in provider tables** IMPROVED 55

Provider model tables in the gateway and dashboard now support filtering a provider's offered models by serving switch (served or unserved) and searching offered models by name.

— Names both additions and the exact UI surfaces they apply to v0.45.0

02 **vLLM catalog compatibility via owned_by** IMPROVED 55

Otari's catalog now reads a vLLM-format model listing under any `owned_by` value, expanding self-hosted vLLM compatibility.

— Names the exact field and behaviour change driving compatibility v0.45.0

03 **API key disable-reason tracking replaces actor field** BREAKING 55

API keys now record why a key is disabled rather than who disabled it — the field tracking the disabling actor has been replaced.

— Names the exact field change but not the new field's name v0.45.0

04 **CORS support for browser-based model calls** NEW 40

Otari adds CORS support so browser pages can call model endpoints directly with an API key.

— Names the capability but no config key or header details v0.46.0

05 **OAuth app connections with PKCE** NEW 40

Otari adds OAuth app connections with PKCE for authenticating third-party app integrations.

— Names the auth mechanism but no endpoint or setup steps given v0.45.0

06 **Operator update notifications** NEW 40

Operators can now opt in to release and security notifications via an update-check prompt.

— Clear opt-in UI prompt but no further mechanism detail v0.45.0

07 **Auto-expiring uploaded files on Railway deploys** IMPROVED 35

Uploaded files on Railway deploys now expire automatically after one week.

— States the behaviour and timeframe but it's automatic, nothing to configure v0.46.0

08 **Org member provisioning provenance** NEW 35

Otari now marks which organization members were provisioned by the organization's own connection.

— States what is marked but not where it's surfaced v0.45.0

09 **Video price discrepancy warning** NEW 30

Otari now warns when a video's price differs significantly from the provider's actual charge.

— Describes the warning but not its threshold or location v0.45.0

10 **Web search and code execution out of beta** IMPROVED 30

Web search and Code execution are now out of beta in the web UI.

— Status change noted without new capability detail v0.45.0

└──▷ **BREAKING ON UPGRADE**

! API keys now record why a key is disabled rather than who disabled it — the field tracking the disabling actor has been replaced.

WAS THIS USEFUL?  

DATA

◆ Vector DBs & RAG

infino

Feedback ⓘ v0.11.30 NOTES ↗ RANK

Infino is an embedded retrieval library built on Parquet. Fast, efficient, and scalable.

> Infino's v0.11.30 release sharpens SQL diagnostics and filtering while reworking how compaction and deleted rows are handled to keep queries fast under bounded memory.

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

01 WHERE clause filtering for exact and token match NEW

75

Infino's SQL engine now supports `WHERE` clause filtering for `exact_match` and `token_match` queries, letting full-text predicates be combined with other column filters in a single `SELECT` statement.

Filter full-text search results with a SQL WHERE clause using `exact_match`, now supported in this release.

```rust
let results = docs.sql("SELECT * FROM docs WHERE
exact_match(body, 'disk full') AND category = 'storage'").await?;
```

— Named SQL functions with a runnable query example. v0.11.30

02 Bounded-memory compaction and deleted-row handling IMPROVED

60

Infino now runs compaction in bounded memory, processing one pass of columns at a time, rewrites full superfiles once a fifth of their rows are deleted to keep read performance consistent, and answers aggregates from statistics rather than full scans when superfiles carry deleted rows.

— Mechanism and thresholds given but it's internal, no user action. v0.11.30

THINNER COVERAGE BELOW

03 Index check on reader pool during query planning IMPROVED

35

Infino now checks local superfiles' indexes on the reader pool during query planning, reducing unnecessary work at plan time.

— Names the mechanism but no user-facing action. v0.11.30

04 **Closest column suggestion for unknown-column errors** IMPROVED

20

Infino now suggests the closest column name when a query references an unknown column, helping diagnose schema mismatches faster.

— Describes the behaviour only, no mechanism or example. v0.11.30

WAS THIS USEFUL? 👍 👎

synthadoc

Feedback ⓘ v1.5.2 NOTES ↗ RANK

Synthadoc: An open-source LLM knowledge compilation engine that turns raw documents into structured, local-first wikis. A transparent, human-readable alternative to traditional RAG, which can be self-managed and self-improved without the use of any tools.

> Synthadoc 1.5.2 adds run-scoped confirm gates for agentic workflows, cross-wiki knowledge-gap detection with suggested follow-up searches, richer search result scoring, and several reliability and UI fixes.

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

01 Run-scoped confirm gates for agentic workflows BREAKING

90

`POST /action/confirm` and `POST /action/confirm-scoped` now require a `run_id` field, scoping approval gates to a single workflow run and preventing approvals from leaking between concurrent runs; callers that do not supply it are rejected.

Confirm a pending agentic-workflow action scoped to a specific run, so approvals cannot bleed across concurrent sessions.

```
$ curl -X POST http://localhost:<port>/action/confirm-scoped \
  -H 'Content-Type: application/json' \
  -d '{"run_id": "<uuid>", "confirmed": true}'
```

— Named endpoints, required field, and a runnable curl example v1.5.2

02 Cross-wiki knowledge-gap detection NEW

80

Synthadoc now detects knowledge gaps in cross-wiki federated queries, surfacing `suggested_searches` for uncovered sub-questions instead of returning a fluent but empty answer, using a fused `gap_score_threshold` (default 2.0) guarded by _guard_c_suppress().

— Mechanism and default threshold named, no direct command shown v1.5.2

03 Startup Docker flag logging and image support IMPROVED

65

`synthadoc serve` now logs the exact `-p <port>:<port>` Docker flag needed at startup, and the Docker image `chenp/synthadoc:1.5.2` supports both amd64 and arm64.

— Names the exact flag and image tag, directly usable v1.5.2

04 **Search result scoring fields** IMPROVED 60

`synthadoc_search` results now expose `rank_score` and `similarity` fields, with embeddings now synced inline.

— Named fields but no usage example given v1.5.2

THINNER COVERAGE BELOW

05 **Safe rollback errors on pruned snapshot** IMPROVED 55

Safe rollback now raises a genuine error when an explicitly requested `snapshot_id` was pruned, rather than silently falling back to a different snapshot.

— Behavior change described but no reproduction steps v1.5.2

06 **Rendered wikilink citations in web UI** IMPROVED 40

Inline `[[wiki::Title]]` citations in the web UI now render as links rather than raw wikilink syntax.

— Describes UI change but no navigation path given v1.5.2

└──▷ **BREAKING ON UPGRADE**

! `POST /action/confirm` and `POST /action/confirm-scoped` now require a `run_id` field; callers that do not supply it will be rejected.

WAS THIS USEFUL?  

ThHanke Ontosphere

Feedback ⓘ v1.8.1 NOTES ↗ RANK

Ontosphere is a browser-based RDF/ontology knowledge graph editor — load RDF from files, URLs or SPARQL endpoints; author nodes and edges on the canvas; run OWL-RL reasoning; apply multi-algorithm layout and clustering. AI-native via MCP tool surface and relay bridge.

> Ontosphere's v1.8.1 release reworks in-browser OWL reasoning for roughly 6x faster entailment and a smaller WASM payload, adds a justification-mode control to `explainEntailment`, and significantly restructures SHACL validation defaults, info-level findings, and cross-graph target resolution.

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

01 Justification mode control for explainEntailment

BREAKING

95

`explainEntailment` now accepts a `justificationMode` input of `"exact"` (guaranteed-minimal) or `"fast"`; the response names the mode used and sets `justificationUnavailable: true` with a `reason` when neither mode finds a result within the 60 s limit. The default is now `"exact"`, which can take up to 60 s, and callers that previously read an empty `justifications` list as directly asserted must now check `justificationUnavailable` instead, since a directly asserted axiom returns as its own one-axiom justification.

— Exact parameter names, defaults, timeout and response fields given. v1.8.1

02 Default SHACL shapes reduced to labels only

BREAKING

85

Default SHACL shapes now check only labels (`rdfs:label`, `skos:prefLabel`, or `dcterms:title`; at most one per language; no `rdfs:subClassOf` cycles). Checks for comments, domain, range, and `skos:inScheme` have moved to the opt-in `ontology-documentation.shacl.ttl` preset (Ontology Documentation), and the SKOS Quality preset is removed.

— Names exact predicates, preset filename, and what was removed. v1.8.1

03 User prefixes preserved in loadRdf

BREAKING

70

`loadRdf` with inline Turtle now keeps a user-defined `@prefix` for a built-in prefix name (`ex`, `schema`, `dc`, `dcterms`, `foaf`, `skos`, `rdf`, `rdfs`, `owl`, `xsd`) instead of silently replacing it with the built-in IRI.

— Names the function and every affected prefix explicitly. v1.8.1

04 Reasoner upgrade speeds reasoning ~6x

IMPROVED

65

In-browser reasoning is roughly six times faster for large ontologies after upgrading the pinned reasoner to `rdf-reasoner-konclude` 0.9.5, with PMDCO reasoning dropping from 14.9 s to 2.5 s; the reasoner's WASM binary also shrank from 11.9 MB to 7.05 MB, reducing download size.

— Concrete before/after benchmarks but nothing for a reader to invoke. v1.8.1

---

05 **Cross-graph SHACL target resolution via read-only view** IMPROVED 65

SHACL validation now reads the N3 store through a read-only view that merges shapes, data, loaded ontologies, and inferred triples, with `sh:targetClass` and `sh:class` following `rdfs:subClassOf` across graphs.

— Names the exact SHACL predicates and merge mechanism, but automatic. v1.8.1

---

06 **sh:Info findings moved to Warnings tab** BREAKING 60

`sh:Info` findings are no longer treated as warnings and no longer draw canvas badges; they now appear in the Warnings tab on request and as `infoCount`.

— Clear before/after and UI location, no code surface to run. v1.8.1

---

07 **WebAssembly exception handling now required** BREAKING 60

Browser WebAssembly exception handling is now required to load the reasoner module: Chrome 95+, Firefox 100+, Safari 15.2+, Node 17+; older runtimes cannot load it.

— Exact version thresholds let readers check compatibility. v1.8.1

THINNER COVERAGE BELOW

---

08 **SHACL structural checks merged into single query** IMPROVED 45

SHACL label-uniqueness, class-cycle, and orphan-class checks now run as a single SPARQL query instead of one query per node.

— Names the three checks but offers nothing to act on. v1.8.1

└──▷ **BREAKING ON UPGRADE**

! `explainEntailment` defaults to `justificationMode: "exact"`, which can take seconds (up to 60 s); agents that previously read an empty `justifications` list as directly asserted must now read `justificationUnavailable` instead, because a directly asserted axiom is returned as its own one-axiom justification.

! Browser WebAssembly exception handling is now required: Chrome 95+, Firefox 100+, Safari 15.2+, Node 17+. Older runtimes cannot load the reasoner module.

! SHACL default shapes now check labels only (`rdfs:label` , `skos:prefLabel` , or `dcterms:title` ; at most one per language; no `rdfs:subClassOf` cycles); checks for comments, domain, range, and `skos:inScheme` moved to the opt-in `ontology-documentation.shacl.ttl` preset, and the SKOS Quality preset is removed.

! `sh:Info` findings are no longer treated as warnings and no longer draw canvas badges; they appear in the Warnings tab on request and as `infoCount` .

! `loadRdf` with inline Turtle now keeps a user-defined `@prefix` for a built-in prefix name (`ex` , `schema` , `dc` , `dcterms` , `foaf` , `skos` , `rdf` , `rdfs` , `owl` , `xsd`) instead of silently replacing it with the built-in IRI.

WAS THIS USEFUL?  

AWS Context Ontology Accelerator

Feedback ⓘ

v0.3.10 CODE ↗ RANK

Context Ontology Accelerator provides an open-source semantic layer that helps AI agents make consistent, explainable decisions using formal ontologies and knowledge graphs.

> Context Ontology Accelerator v0.3.10 adds per-request LLM token usage accounting and capability-based suppression of source and ontology schema details in responses.

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

01 **Per-request LLM token usage accounting** NEW 86

Setting `options.includeLlmUsage` on a request enables a tracker that records input/output tokens and model IDs across all Bedrock, GraphRAG, and Strands agent calls made for that request.

— Names exact flag and scope of tracked calls v0.3.10

02 **Capability-gated schema suppression in responses** NEW 81

Context Ontology Accelerator now suppresses source schema (table/column names, compiled SQL) and ontology schema (class/property IRIs, generated SPARQL) from responses for callers lacking `viewSource` or `viewOntology` capabilities, applying field blanking and per-value redaction across `query`, `translate`, and `graphTraverse` actions.

— Names capabilities, actions, and redaction mechanism v0.3.10

WAS THIS USEFUL? 👍 👎

ralforion OrionBelt Ontology Builder

Feedback ⓘ

v1.32.0 NOTES ↗ RANK

OrionBelt Ontology Builder is a browser-based ontology workbench for OWL ontologies and SKOS vocabularies. Streamlit + rdflib, no Java, no Protégé.

> OrionBelt Ontology Builder's biggest change this window is a second SPARQL query engine (Oxigraph) for large ontologies, paired with a reworked Dashboard overview that surfaces coverage, hierarchy and validation metrics up front, plus a broadened set of validation and quality checks.

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

01 **Oxigraph SPARQL engine for large ontologies** NEW 90

A new Engine setting on the SPARQL page lets users choose Automatic, Oxigraph, or rdflib as the query engine; Automatic runs Oxigraph for ontologies of 20,000 triples or more and rdflib below that threshold. The choice is remembered alongside row and time limits, and the result panel shows which engine ran. On a 213,000-triple ontology, moving to Oxigraph cut query time from 10.9s to 1.7s for Ancestors at any depth and from 2.7s to 0.7s for Everything about one resource.

Force Oxigraph for a specific SPARQL query on a large ontology when you want consistent engine behaviour regardless of triple count.

📍 In the SPARQL page, open the Engine setting and select Oxigraph. Run your query — the result panel will confirm Oxigraph as the engine used.

— Names exact setting, engines, and before/after query times. v1.31.3

02 **Dashboard overview with coverage and hierarchy metrics** NEW 90

The Dashboard now opens on an overview card showing the ontology's title, IRI, version, triple count, imports, and tiles for classes, object/data properties, individuals, and restrictions (or concept schemes and concepts for SKOS vocabularies). A Coverage section shows how many classes, properties, and concepts have a label and definition, and how many properties have a domain and range. A Class hierarchy section shows depth, root, leaf, and multi-parent classes plus a chart of classes per depth. A Checks section runs validation and model quality rules automatically after each edit for ontologies up to 10,000 triples; above that a Run checks button triggers them, and Open Quality navigates to the Quality section with the run loaded. Individual-per-class and label-language breakdowns are also shown.

After loading a large ontology, trigger quality validation manually from the new Dashboard overview when the ontology exceeds the 10,000-triple auto-run threshold.

📍 In the Dashboard, scroll to the Checks tile and click `Run checks`, then click `Open Quality` to view the full findings in the Quality section.

Quickly assess documentation completeness for an ontology before a review — no need to leave the Dashboard.

📍 Open the Dashboard; the Coverage tile shows the percentage of classes, properties, and concepts that have labels and definitions, and which properties are missing a domain or range.

— Describes every tile, threshold, and navigation button named. v1.32.0

---

03 **Expanded validation, model quality, and SHACL checks** NEW

60

Validation now catches orphan classes, domain/range mismatches, missing annotations, and SKOS broader/narrower cycles. Model quality rules flag redundant hierarchy edges, disconnected islands, and naming problems. SHACL shapes check data against custom constraints.

— Lists all new check types but no UI path to run them. product docs

THINNER COVERAGE BELOW

04 **Settings tab replaces Dashboard default forms** BREAKING

50

The Dashboard no longer opens on the metadata, imports, and namespace prefix forms by default; these have moved to a new Settings tab, from which namespace prefixes can now be managed directly.

— Clear UI relocation but no further mechanism detail. v1.32.0

└──▷ **ALSO FROM THESE RELEASES**

Pin a specific version when starting the tool locally to avoid unexpected changes from a rolling `latest` image.

```
$ docker run -p 8501:8501 orionbelt-ontology-builder:1.32.0
# Then open http://localhost:8501
```

└──▷ **BREAKING ON UPGRADE**

! The Dashboard no longer opens on the metadata, imports, and namespace prefix forms by default — these have moved to a `Settings` tab on the Dashboard.

WAS THIS USEFUL? 👍 👎

Crow Desktop

Feedback ⓘ v1.0.0 NOTES ↗ RANK

Crow Desktop is a crow: build your second brain from the documents on your computer. Downloads for macOS, Windows and Linux.

> Crow Desktop's 1.0 release adds offline local-model support alongside API-key sign-in, auto-organizes wikis from onboarding answers, and introduces manual drag-and-drop folder management plus a Mac App Store build.

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

01 **Local and API-key model connections** NEW 95

Crow adds new ways to connect AI models: use your own `OpenRouter` key without signing in via Connect → Use an API key instead (removable later via Settings → Sign out); or run Crow with no cloud account using local models via Connect → On this computer, which downloads `Ollama 0.40.16` plus three bundled models — `LFM3 1.7B` for writing notes, `Tev1 0.8B` for filing, and `EmbeddingGemma 2` for search — about 2.4 GB of models plus the engine. You can switch between `OpenRouter` and local models, or remove the local model package, at any time via Settings → Model.

Use Crow entirely offline — useful on air-gapped or sensitive workstations — by switching to local models from the Connect screen.

📍 1. Open Crow and go to Connect. 2. Choose On this computer instead of OpenRouter. 3. Click the download button; progress is shown and can be paused and resumed. 4. Once complete, Crow files and answers questions locally using LFM3 1.7B, Tev1 0.8B, and EmbeddingGemma 2. 5. To switch back or remove the package later, go to Settings → Model.

Connect Crow to OpenRouter with a personal API key instead of a full sign-in flow — useful for shared machines or service accounts.

📍 1. Open Crow and go to Connect. 2. Click Use an API key instead . 3. Paste your OpenRouter API key. 4. To revoke access later, go to Settings → Sign out.

— Exact UI paths, model names and versions given v1.0.0

02 **Drag-and-drop wiki organization** NEW 75

Crow adds drag-and-drop wiki organisation: drag a note onto a folder to move it, or a folder onto another to nest it; right-click a folder or note to rename, describe, or move it; links between notes follow every move.

Reorganise your wiki folder structure manually without breaking note links — useful after a project rename or client offboard.

📍 1. In the brain view, drag a note onto the destination folder; the folder lights up to confirm the target. 2. Crow prompts for confirmation before moving. 3. To nest a folder inside another, drag it onto the target folder (or onto the wiki bubble for top level). 4. Right-click any folder to rename, redescribe, or move it; all links between notes update automatically.

— Exact drag and right-click steps given with example v1.0.0

03 **Onboarding-driven wiki auto-filing** NEW 70

With local models, Crow's onboarding now asks what your wiki is for, what your documents are about, and the names of clients, projects, or courses, then generates top-level wiki folders from your answers. After filing, Crow organises notes into subfolders for each named client, product, or course, marking uncertain notes '(?)' and routing them to a To review folder, while keeping passwords, API keys, IBANs, and Italian tax codes out of note titles, summaries, and folder names.

— Mechanism described but no exact command or UI steps v1.0.0

04 **Mac App Store build and sandboxing** NEW 70

Crow now ships a Mac App Store build (via `scripts/mas/build.sh`, documented in `docs/mac-app-store.md`) running in the App Sandbox, with one-time home-folder permission that persists across restarts.

— Names build script and docs but no usage walkthrough v1.0.0

05 **Review card for uncertain notes** NEW 60

Notes Crow is uncertain about — marked '(?)' with local models, or unmatched on OpenRouter once the wiki has ten or more main folders — surface in a card over the brain so you can assign, create, or skip folders for each.

— Trigger and UI described but no exact navigation steps v1.0.0

THINNER COVERAGE BELOW

06 **Search tool for connected AI apps** NEW 40

Connected AI apps now get a `search` tool that finds notes without a model call.

— Thin description with no interface details given v1.0.0

WAS THIS USEFUL? 👍 👎

aegis-stack

Feedback ⓘ v0.20.0 NOTES ↗ RANK

Aegis-stack is a production-ready FastAPI platform with modular components and a built-in control plane.

> aegis-stack v0.20.0 adds a full evals system for AI assistants, Brave-powered web search tools, a merge-triggered continuous-deploy flag, and reference-tracked persistent storage.

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

01 **AI assistant evals system** NEW 91

Run AI assistant evaluations via `ai evals run`, `make evals`, a background job, or the API; eval datasets live in the database, support YAML import/export, and two runs can be compared. Running evals now requires a database backend — the `ai[evals]` package is refused when using the memory backend.

Run the full eval suite against the AI assistant from the CLI to validate tool calls and responses before shipping a change.

```
$ ai evals run
```

— Names exact commands, formats, and new backend constraint v0.20.0

02 **Continuous deploy on merge to main** NEW 80

Trigger automatic deployment on every merge to main with `aegis deploy-cd-setup --on-merge`, which deploys at the exact commit CI tested.

Set up continuous deployment so every merge to main that passes CI is deployed automatically at the tested commit.

```
$ aegis deploy-cd-setup --on-merge
```

— Runnable command with clear deploy behavior v0.20.0

03 **Web search tools for AI assistant** NEW 73

Enable web search for the AI assistant by setting `BRAVE_API_KEY`; this activates the `web_search` tool (Brave) and the `read_page` tool, a guarded fetch restricted to public hosts. Both tools remain off until the key is set.

— Names exact env var and the two new tools v0.20.0

04 **Reference-tracked persistent storage** IMPROVED 60

Persistent storage now tracks every object stored through store() with a row for key, size, and content type, and purges a file only when no part references it.

— Names store() function and purge mechanism v0.20.0

└──▷ **ALSO FROM THESE RELEASES**

.
.
.

└──▷ **BREAKING ON UPGRADE**

! `ai[evals]` is refused when using the memory backend — a database backend is now required to run evals.

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

LangChain LangSmith

Feedback ⓘ updated · Nov 13, 2026

NOTES ↗ RANK

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

> LangSmith added Assumed IAM Role authentication for Amazon Bedrock in its LLM Gateway and introduced thread-scoped automations built on SmithDB queries, while also fixing a WebSocket query-string bug, improving API key search, and deprecating the Turns view.

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

01 Assumed IAM Role auth for Amazon Bedrock in LLM Gateway NEW

80

LangSmith's LLM Gateway can now route traffic to Amazon Bedrock using an assumed IAM role instead of a static API key, so credentials rotate automatically. Configure it in the UI under `Settings > LLM Gateway > Providers > Amazon Bedrock`, selecting Assumed IAM Role and supplying a Role ARN and optional external ID.

Route LangSmith LLM Gateway traffic to Amazon Bedrock using an assumed IAM role instead of a static API key, so credentials rotate automatically.

> 📍 In the LangSmith UI, go to Settings › LLM Gateway › Providers › Amazon Bedrock and select Assumed IAM Role, then enter your Role ARN and optional external ID.

– Mechanism and exact UI navigation given, but it's a UI path not a command

snapshot-20261113

THINNER COVERAGE BELOW

02 WebSocket query strings preserved on sandbox URLs IMPROVED

50

WebSocket connections via sandbox service URLs now keep their full query string, unblocking dev servers that authenticate with a query parameter such as Vite's HMR.

– Explains the fix and an example case but gives no steps to verify snapshot-20261113

└──▷ **BREAKING ON UPGRADE**

! Thread annotation-queue and webhook automations now require SmithDB queries; existing automations using other query types will need to be migrated.

! The Turns view in thread detail is deprecated and will be removed from LangSmith Cloud on October 31, 2026, and from self-hosted in v17; use the Messages or Details view instead.

WAS THIS USEFUL? 👍 👎

agentic-os

Feedback ⓘ v1.8.39 NOTES ↗ RANK

agentic-os is a governance framework for AI coding agents. It runs them through a five-step workflow (plan, implement, validate, review, ship) with test evidence tracking, capabilities validation, and engineering guardrails.

> agentic-os shipped KB-declared reading plans that let knowledge bases control how the agent reads their pages, alongside a mandatory design-artifact gate for UI tasks, manual sign-off support as a test receipt when no runner exists, and removal of nine dead `.agent/config.yaml` keys plus an unused script.

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

01 KB-declared reading plans NEW

83

Knowledge bases can now declare a labeled reading plan at the head of their entrypoint file, and the agent uses it to control page choice, order, count, and stopping point instead of relying on framework defaults. Bootstrap records per-KB whether a plan was found, logging `<id>→OK plan=declared|default` to the Work Log, and the behavior is documented in a new spec `docs/specs/kb-seam-declared-reading.md` and ADR-009 Amendment (2026-11-13). The guide `connecting-a-knowledge-base.md` gains new sections Declare how your KB is read and 'What the framework reads from any KB' and no longer types read hints as true/false or single values; a new eval case validated against Claude Sonnet and Haiku checks the agent starts and stops where the plan says. Note: KBs built from the earlier guide that dropped `summary` from rows with a digest will see their fallback path change on upgrade, since the pre-v1.8.39 fallback read summary and headings differently when a digest was stale or unreadable.

— names log format, spec file, guide sections; no runnable example given v1.8.39

02 Design-artifact gate for UI tasks NEW

83

Adds a design-gate rule to `AGENTS.md` so that any UI edit in production directories — however the task is phrased — requires a named design artifact (a link or a committed `docs/design/<screen>.md`) before code is written, or the agent offers to draft the wireframe itself.

— names exact config file and artifact path pattern v1.8.38

03 Removed unused config keys and script DEPRECATED

80

Removes nine keys from `.agent/config.yaml` — `raw_intake_cleanup`, four keys under `cluster`, and four keys under `downstream_capabilities` — because validators hard-code those values and editing them had no effect. Deploy replaces `.agent/config.yaml`, saving any local edits as `.agent/config.yaml.acx-local` for comparison after upgrade, and also removes `verify_agent_evidence.py` and its test, which were never deployed.

— names every removed key and file path verbatim v1.8.38

04 **Manual sign-off as test receipt** NEW 70

When no test runner is present, `/test` now accepts your recorded sign-off as the acceptance receipt that `/ship` requires, with the Work Log noting the evidence as a manual trace.

— names commands involved but mechanism of recording unclear v1.8.38

└──▷ **BREAKING ON UPGRADE**

! KBs built from the earlier guide that dropped `summary` from rows that have a digest will have those rows' fallback path change on upgrade: on v1.8.38 and earlier the fallback reads the summary and headings, so a stale or unreadable digest triggers different behavior than on v1.8.39.

! Deploy replaces `.agent/config.yaml` and saves any local edits as `.agent/config.yaml.acx-local`; compare the two after upgrading.

! Nine keys in `.agent/config.yaml` are removed: `raw_intake_cleanup`, four keys under `cluster`, and four keys under `downstream_capabilities` — any config relying on editing these keys had no effect and those keys no longer exist.

WAS THIS USEFUL? 👍 👎

Arize Phoenix

Feedback ⓘ arize-phoenix-v21.1.0 NOTES ↗ RANK

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

> Arize Phoenix's v21.1.0 release adds CLI commands for listing dataset and prompt versions and surfaces a new model column in the spans table UI.

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

01 **Model column in spans table** NEW 35

The spans table in the Phoenix UI now shows a model column for each span.

— Names UI change but no further detail or navigation. arize-phoenix-v21.1.0

WAS THIS USEFUL? 👍 👎

MLflow

Feedback

CVE-2026-118628

CVE ↗

MLflow tracks, evaluates, and monitors machine learning and LLM applications, covering experiment tracking, tracing, prompt management, and a model registry.

`CVE-2026-118628`: Server-Side Request Forgery in the MLflow tracking server via client-supplied artifact locations and model version sources (CVSS 8.2, affects `< 3.18.0`)

WAS THIS USEFUL? 👍 👎

Langfuse

Feedback ⓘ v4.79.0 NOTES ↗

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

Langfuse adds MCP audit logging, monitor `SUBSTITUTE_OK` mode, trace transcript eval scheduling, audit event staging for self-hosted enterprise, and topics multi-rule support.

└──▷ **GET THIS VERSION** ▾

Docker

```
$ docker pull ghcr.io/langfuse/langfuse:4.79.0
```

git

```
$ git clone --branch v4.79.0 https://github.com/langfuse/langfuse.git
# already have the repo? check out this version:
$ git checkout v4.79.0
```

published under the project's GitHub account · checked 13 Nov Wrong command?

Langfuse adds a `SUBSTITUTE_OK` no-data mode for monitors.

Langfuse adds scheduling of trace transcript evaluations.

Langfuse's audit logs now record read-access events for MCP tools.

Langfuse now stages and relays audit events on self-hosted enterprise installs.

Langfuse adds MCP tools to manage the default evaluation model.

+9 more

NEW INTEGRATION OpenRouter AI gateway provider

NEW INTEGRATION xAI AI gateway provider

NEW SUPPORTED PLATFORM Vertex AI Flex PayGo Gemini pricing via trafficType service tiers

└──▷ **BREAKING ON UPGRADE**

! The `isAdmin` flag has been deleted.

WAS THIS USEFUL? 👍 👎

Braintrust

Feedback ⓘ updated · Nov 13, 2026 NOTES ↗ RANK

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

> Braintrust shipped on-device trace translation, configurable pattern rules for Loop, OpenCode 2 tracing support, LLM service-tier tracking across ten providers, and a generally available Terraform provider that adds new resource types alongside breaking changes to secret/env-var versioning and the removal of the automation resource.

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

01 Terraform provider breaking changes: versioned secrets and removed automation resource

95

BREAKING

The Terraform provider now requires Terraform or OpenTofu 1.11.1 or later. `braintrust_ai_secret` now requires `secret_version` alongside `secret`, and `braintrust_env_var` now requires `value_version` alongside `value` — you must add a version to each configured value and increment it to write a new value, since changing only the value no longer updates it, and references to `secret` or `value` return `null`. The `braintrust_org_automation` resource is removed because the Braintrust API no longer supports it; delete its blocks from your configuration, then run `terraform state rm` on each address before running `terraform plan`.

— exact config keys and migration commands make this directly actionable

snapshot-20261113

02 OpenCode 2 tracing via `bt trace run opencode` NEW

90

Trace OpenCode 2 using `bt trace run opencode`, which checks the OpenCode version before launching and runs OpenCode 2 with a private server. Requires `@braintrust/trace-opencode` v2.2.0 or later.

— exact command and version requirement make this directly runnable snapshot-20261113

03 Terraform resources for Patterns views and ephemeral service tokens

NEW

90

The Terraform provider is now generally available with new resource types, including a `braintrust_view` resource for saving Patterns configuration (e.g. `view_type = "patterns"`) in version control, and an ephemeral `braintrust_service_token`

resource for provisioning short-lived service tokens without persisting the key in state or plan files.

Create a Patterns saved view in Terraform so Loop schedule configuration is tracked in version control.

```hcl
resource "braintrust_view" "my_patterns_view" {
  name      = "Weekly error patterns"
  view_type = "patterns"
}
```

Provision a short-lived service token in Terraform without persisting the key in state or plan files.

```hcl
ephemeral "braintrust_service_token" "ci_token" {
  name = "ci-deploy"
}
```

— named resources with runnable HCL examples provided `snapshot-20261113`

04 **LLM service tier tracking in TypeScript SDK** NEW 73

Spans in the TypeScript SDK now record the requested tier as `requested_service_tier` and the provider-used tier as `served_service_tier`, covering OpenAI, Anthropic, Gemini, Bedrock, Groq, OpenRouter, the AI SDK, LangChain, LangGraph, and the OpenAI Agents SDK.

— field names and provider list given, no code sample `snapshot-20261113`

THINNER COVERAGE BELOW

05 **On-device trace translation in Chrome/Edge** NEW 56

Translate an entire non-English trace to English — including span fields, span names, thread messages, and custom trace views — using the browser's built-in translation API in desktop Chrome or Edge. Translation happens on-device, so trace text stays local and uses no model credits.

— explains mechanism but no flag or menu path given `snapshot-20261113`

06 **Pattern rules for scheduled Loop runs** NEW 52

Define pattern rules to tell Loop what to investigate in each scheduled Patterns run, with an optional SQL filter to narrow scope to specific traces and a list of problems for Loop to watch for.

— concept and filter type named but no concrete syntax shown `snapshot-20261113`

07 Capture controls for LangChain/LangGraph spans NEW

37

Omit inputs, outputs, or token usage from LangChain and LangGraph spans using new capture controls added to the TypeScript SDK.

— no exact flag or config key named for the new controls `snapshot-20261113`

└──▷ BREAKING ON UPGRADE

! The Terraform provider now requires Terraform or OpenTofu 1.11.1 or later.

! `braintrust_ai_secret` now requires `secret_version` alongside `secret`; `braintrust_env_var` now requires `value_version` alongside `value`. You must add a version to each configured value and increment it to write a new value — changing only the value no longer updates it. References to `secret` or `value` return `null`.

! The `braintrust_org_automation` Terraform resource is removed because the Braintrust API no longer supports it. Delete its blocks from your configuration, then run `terraform state rm` on each address before running `terraform plan`.

WAS THIS USEFUL?  

GOVERN

◆ AI Runtime Defense & Guardrails

Anthropic Sandbox Runtime

Feedback ⓘ v0.0.83

NOTES ↗ RANK

Sandbox Runtime is a lightweight sandboxing tool for enforcing filesystem and network restrictions on arbitrary processes at the OS level, without requiring a container.

> Sandbox Runtime's v0.0.83 release expands the Windows MXC backend with custom SDK injection and environment variable support, while making the underlying MXC SDK dependency optional—a breaking change for existing npm installs.

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

01 **Windows MXC backend customization and env support** NEW

74

The Windows MXC backend now accepts a `windowsMxcSdk` parameter passed to initialize() to supply a custom MXC SDK instance, with `MxcSdk` types exported for type-safe embedding. It also supports setting environment variables on the child process by passing `env` to spawnSandboxed().

— Names exact parameters and functions but no further mechanism detail. v0.0.83

02 **MXC SDK peer dependency now optional** BREAKING

67

`@microsoft/mxc-sdk` is no longer an automatic dependency for `windows.backend: 'mxc'`; npm users must now install it explicitly themselves or supply their own instance via `windowsMxcSdk`, otherwise the backend will fail to load.

— Clear before/after and migration note, but brief. v0.0.83

└──▷ **BREAKING ON UPGRADE**

! `@microsoft/mxc-sdk` is no longer an automatic dependency for `windows.backend: 'mxc'` — npm users must now install it explicitly or pass their own instance via `windowsMxcSdk`, or the backend will fail to load.

WAS THIS USEFUL? 👍 👎

◆ Model & Agent Protection

Minimal Software Research minimal

Feedback ⓘ

release-049c4fbd NOTES ↗ RANK

Minimal is a secure, declarative runtimes for Claude Code, Gemini, Codex, Pi and more, so you can run multiple agents safely on your stack from a verified software supply chain.

> Minimal's latest release adds secret storage, hash-chained audit trail inspection with tamper detection, network/egress controls for task and box runs (the new Box Egress Proxy), OpenTelemetry tracing spanning the CLI, daemon and VM guest, GitHub App-based login, and a dedicated vulnerability-reporting flag.

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

01 **Hash-chained audit trail inspection** NEW 88

Read a box's hash-chained audit trail with `min box audit`, merge parent chains with `min box audit --parent`, and have `min doctor` report when an audit chain has been edited.

Inspect the full hash-chained audit trail for a box, merging its parent chain, to verify no decisions were tampered with.

```
$ min box audit --parent my-box-name
```

– Named commands, flag, and a runnable example command release-049c4fbd

02 **OpenTelemetry tracing across CLI, daemon, and VM guest** NEW 76

Opt in to OpenTelemetry export via `mlog` with a local spool, covering one trace per request across the CLI and daemon; export VM host daemon spans and carry telemetry into the guest via `minvmd`, with guest telemetry riding the VM channel.

– Names mechanism and components but no enabling command shown release-049c4fbd

03 **Network and egress flags for task and box runs** NEW 65

Take network and egress control on `min task run` and `min box run` using the new `--network` and `--egress` flags, part of the new Box Egress Proxy.

– Names exact flags but no example invocation or defaults release-049c4fbd

04 **GitHub sign-in via published App** NEW 60

Sign in to GitHub through the published App with `min login`, which also deletes the sign-in on logout.

— Names command and behaviour but no setup steps release-049c4fbd

05 **Security vulnerability reporting flag** NEW 60

Send a suspected vulnerability report using the new `--security` flag, invoked as `min bug --security`.

Report a suspected vulnerability in minimal itself to the team via the diagnostics command.



— Includes a runnable example command for the flag release-049c4fbd

THINNER COVERAGE BELOW

06 **Secret storage via `min secret` command** NEW 53

Store, list, and remove secrets with the new `min secret` command, including its `set` subcommand and implied list/remove operations.

— Names command and subcommand but no example or full syntax release-049c4fbd

07 **Store reference listing via `min box spec`** NEW 50

List each store reference alongside its authorities with the new `min box spec` command.

— Names the command but gives no example or output detail release-049c4fbd

08 **QUIC-block helper in install finalization** NEW 35

Name the QUIC-block helper in `min finalize-install`.

— Bare mention with no explanation of mechanism or purpose release-049c4fbd

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Frontier Models

Mistral AI

Feedback ⓘ updated · Nov 13, 2026 NOTES ↗ RANK

Learn how to deploy and use Mistral AI's Large Language Models with our comprehensive documentation, guides, and tutorials.

> Mistral AI added enterprise identity provisioning and new controls to steer model outputs via context bias.

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

01 **SCIM provisioning with Microsoft Entra ID** NEW 40

Mistral AI adds SCIM provisioning support with Microsoft Entra ID for automated user lifecycle management.

— Names integration but no setup steps or endpoint. snapshot-20261113

02 **Context bias controls for model outputs** NEW 35

Mistral AI adds context bias controls accepting keyphrases and free-form text to steer model outputs.

— Describes input types but no API or parameter name given. snapshot-20261113

WAS THIS USEFUL? 👍 👎

Anthropic

Feedback ⓘ updated · Nov 13, 2026 NOTES ↗ RANK

The Claude Developer Platform is the API for building applications on Anthropic's Claude models that generate, analyze, and automate work.

> Anthropic added self-hosted Claude Managed Agents, letting developers run the full agents service on their own infrastructure instead of relying solely on Anthropic-hosted sandboxes.

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

01 **Self-hosted Claude Managed Agents** NEW 50

The complete Managed Agents service — not only sandboxes — can now run on your own infrastructure, with agents, sessions, and session history stored locally. Model requests are routed to a provider of your choice, with Anthropic as the default.

— Explains scope and data locality but no setup steps or config keys given.

snapshot-20261113

WAS THIS USEFUL? 👍 👎

Google Gemini 4 Hydrogen

Feedback ⓘ

released · Nov 13, 2026 MODEL ↗ NOTES ↗

Gemini 4 Hydrogen from Google was released on 12 Nov 2026. 1049K-token context; $1.50 in / $7.50 out per million tokens (OpenRouter).

Gemini API adds Gemini 4 Hydrogen (`gemini-4-hydrogen`) at GA with a new `xhigh` thinking level and default Antigravity agent support

Access `gemini-4-hydrogen` on the free tier or at $1.50 per 1M input tokens and $7.50 per 1M output tokens on the paid tier.

Use the new `xhigh` thinking level alongside the existing `low`, `medium`, and `high` levels when calling `gemini-4-hydrogen` to control reasoning depth.

`gemini-4-hydrogen` is now the default model for the Antigravity agent in Gemini Managed Agents.

WAS THIS USEFUL? 👍 👎

OpenAI GPT-5 Feedback ⓘ retirement scheduled · Nov 12, 2026

MODEL ↗

GPT-5 is a reasoning model from OpenAI.

OpenRouter lists a retirement date of 11 Dec 2026 for openai/gpt-5, first seen 12 Nov 2026.

WAS THIS USEFUL? 👍 👎

OpenAI GPT-5 Mini Feedback ⓘ

retirement scheduled · Nov 12, 2026 MODEL ↗

GPT-5 Mini is a reasoning model from OpenAI.

OpenRouter lists a retirement date of 11 Dec 2026 for openai/gpt-5-mini, first seen 12 Nov 2026.

WAS THIS USEFUL? 👍 👎

OpenAI GPT-5 Nano Feedback ⓘ

retirement scheduled · Nov 12, 2026 MODEL ↗

GPT-5 Nano is a reasoning model from OpenAI.

OpenRouter lists a retirement date of 11 Dec 2026 for openai/gpt-5-nano, first seen 12 Nov 2026.

WAS THIS USEFUL? 👍 👎

OpenAI GPT-5 Pro

Feedback ⓘ

retirement scheduled · Nov 12, 2026 MODEL ↗

GPT-5 Pro is a reasoning model from OpenAI.

OpenRouter lists a retirement date of 11 Dec 2026 for openai/gpt-5-pro, first seen 12 Nov 2026.

WAS THIS USEFUL? 👍 👎

OpenAI o3

Feedback ⓘ retirement scheduled · Nov 12, 2026 MODEL ↗

o3 is a reasoning model from OpenAI.

OpenRouter lists a retirement date of 11 Dec 2026 for openai/o3, first seen 12 Nov 2026.

WAS THIS USEFUL? 👍 👎

OpenAI o3 Pro

Feedback ⓘ retirement scheduled · Nov 12, 2026

MODEL ↗

o3 Pro is a reasoning model from OpenAI.

OpenRouter lists a retirement date of 11 Dec 2026 for openai/o3-pro, first seen 12 Nov 2026.

WAS THIS USEFUL? 👍 👎

◆ Open-Weight Models

Xiaomi

Feedback ⓘ v0.1.17 NOTES ↗ RANK

Xiaomi is a mimo Code: Where Models and Agents Co-Evolve

> Xiaomi's MiMo-Code v0.1.17 makes Codex-style prompting, web search, and reasoning-level configuration opt-in and config-driven, while adding caller-supplied session IDs, improved Chinese memory search, and dual-provider sign-in.

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

01 **Opt-in Codex-style prompts and toolset** BREAKING 75

Codex-style prompts and toolset can now be enabled on any model by setting `MIMOCODE_CODEX_MODE=true`. Previously GPT models switched to this automatically; now all models share a single default system prompt unless the flag is set.

— Names exact env var and before/after behavior v0.1.17

02 **Configurable web search via Exa or MCP** BREAKING 72

Built-in Exa web search can be enabled by setting `MIMOCODE_ENABLE_EXA=1`, or a search MCP server can be connected instead. The previous always-on Xiaomi gateway search has been removed.

— Names env var and MCP alternative, clear migration path v0.1.17

03 **Model parameters sourced from models.dev and config** BREAKING 63

Context and output limits, reasoning levels, sampling parameters, effort, and step termination now come from models.dev and user config rather than model-name heuristics. `temperature` and `top_p` are no longer sent by default, which may change model behavior on upgrade if defaults were relied upon.

— Explains mechanism shift but no config keys named v0.1.17

THINNER COVERAGE BELOW

04 **Named reasoning variants with subagent inheritance** IMPROVED 56

Reasoning levels for `/variants` can now be named directly in config, and subagents now inherit the parent session's reasoning level instead of resetting to defaults.

— Names config surface but mechanism detail is thin v0.1.17

05 **Caller-supplied session IDs** NEW 40

Callers can now supply a session ID at session creation time, rather than having one assigned automatically.

— Clear capability but no API detail given v0.1.17

06 **Config-driven always-direct MCP tools** IMPROVED 39

Always-direct MCP tools are now driven from config and made mutable, rather than being fixed.

— Names mechanism loosely, no config key given v0.1.17

07 **Simultaneous Xiaomi and Token Plan sign-in** NEW 35

Xiaomi API keys and Token Plan keys can now be signed in simultaneously under separate providers.

— Names the two providers but no further detail v0.1.17

08 **Chinese substring recall in memory search** IMPROVED 33

Memory search now recalls Chinese substrings and migrates existing indexes to support this.

— States behavior change but no mechanism detail v0.1.17

09 **Removal of cloud session sharing and uploads** BREAKING 26

Cloud session sharing and third-party uploads have been removed in this release.

— Bare statement with no replacement or detail v0.1.17

└──▷ **BREAKING ON UPGRADE**

! Built-in web search via the Xiaomi gateway is removed; set `MIMOCODE_ENABLE_EXA=1` or add a search MCP server to restore web search.

! GPT models no longer automatically switch to a Codex-style prompt and toolset; set `MIMOCODE_CODEX_MODE=true` to opt in.

! `temperature` and `top_p` are no longer sent by default; model behavior may change on upgrade if defaults were relied upon.

! Cloud session sharing and third-party uploads are removed.

WAS THIS USEFUL? 👍 👎

Thinking Machines Inkling 2 Small

Feedback ⓘ

released · Nov 13, 2026 MODEL ↗

Inkling 2 Small from Thinking Machines was released on 9 Nov 2026. 1049K-token context; $0.30 in / $1.00 out per million tokens (OpenRouter).

WAS THIS USEFUL? 👍 👎

vLLM Project d3.1-nano

Feedback ⓘ

released · Nov 13, 2026 MODEL ↗

d3.1-nano from vLLM Project was released on 12 Nov 2026. Open weights on Hugging Face.

WAS THIS USEFUL? 👍 👎

OTHER / UNCATEGORIZED

◆ UNCATEGORIZED

Tokenrip

Feedback ⓘ changes since 2026-10-14 CODE ↗

Tokenrip's command-line client.

Tokenrip adds workspace messaging, job/output/feedback commands, key access controls, and task queue ordering across versions 2.6.0–2.6.24

Adds task list filtering by type tag with `--type`, step tag with `--step`, and label with `--label` (repeatable or comma-separated).

Replace `rip comment` with `rip message` for workspace thread messaging.

New `rip task list --reviewer me` option surfaces In-review tasks addressed to the current user.

Adds new `rip job`, `rip output`, and `rip feedback` commands for managing job definitions, outputs, and feedback, including deleting a job with `rip job delete`.

Create limited-scope API keys and inspect key access with `rip whoami` and `rip keys show access`.

+5 more

└──▷ **BREAKING ON UPGRADE**

- ! `rip comment` is replaced by `rip message` for workspace threads — existing scripts using `rip comment` will break.
- ! `--audience`, `--membership`, and `folder share-contents` are removed from workspace access commands — existing scripts using these flags will break.
- ! `cancel` replaces `decline` in task review rounds — scripts using `decline` will break.
- ! `rip trash restore` syntax changed: `<type> <id>` is replaced by `<id>` alone — existing restore commands with two positional arguments will break.
- ! `billingUrl` is dropped from `DomainErrorDetails` — code or scripts that read this field will no longer find it.

WAS THIS USEFUL? 👍 👎

Foremerge

Feedback ⓘ v0.7.0 NOTES ↗ RANK

Foremerge is a coordination protocol for coding agents that detects intent conflicts before code conflicts.

> Foremerge's v0.7.0 release improves `status` accuracy and output while removing scope fields from conflict listings, requiring clients to fetch scopes separately.

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

01 **Scopes removed from status conflict listings** BREAKING 85

A listed conflict in `status` (CLI, MCP, and `GET /v1/status`) no longer carries `source_scopes` and `target_scopes`; clients that read those fields must read each party's scopes from `get_intent` instead.

– Names exact endpoint, fields, and migration path v0.7.0

02 **Stranded intent detection in status** IMPROVED 60

The `stranded` field in `status` now uses the same test as `foremerge work adopt`, so an intent still in INTENT state whose agent has gone silent is correctly reported as stranded.

– Explains the fix and its mechanism clearly v0.7.0

THINNER COVERAGE BELOW

03 **Pagination added to status output** IMPROVED 25

Foremerge adds pagination to `status` output.

– Only a brief summary mention, no detail v0.7.0

04 **MCP protocol revision negotiation** IMPROVED 25

Foremerge now negotiates MCP protocol revisions correctly.

– Only a brief summary mention, no mechanism v0.7.0

└──▷ **BREAKING ON UPGRADE**

! A listed conflict in `status` (CLI, MCP, and `GET /v1/status`) no longer carries `source_scopes` and `target_scopes`; clients that read those fields must read each party's scopes from `get_intent` instead.

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

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

▷ Subscribe Feedback on this issue