THE AI TOOLCHAIN

NO. 099

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED NOVEMBER 23, 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

INFORMATION DENSITY

everything



BRIEFED | EVERYTHING

Every tool in the issue, in full.

all 201 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 54 tools

▾ // THE BRIEF HIDE SAME ISSUE, SAME PROMPT, TWO WRITERS: Opus 5 | GPT-5.5

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

Three unrelated things stand out. LMCache now moves KV cache between nodes over RDMA and indexes borrowed peer memory, so a

vim ? help

prefill computed on one GPU can be reused on another. Milvus's Go SDK enforces row-level security on reads, writes and bulk imports. KoboldCpp adds local TTS and transcription with word-level timestamps.

DEPLOY

Reuse a prefill computed on another node instead of recomputing it

v0.5.8 adds peer-to-peer RDMA transfer with a shadow-L1 index over borrowed peer memory, plus Valkey, HF3FS and Device-DAX as L2 backends and VMM KV-cache support for ROCm and vLLM. A cache hit is no longer confined to the GPU that produced it, which is what makes long shared system prompts cheap across a fleet rather than per-replica.

LMCache

GOVERN

Let the vector store decide who sees which rows, instead of filtering in app code

Milvus's Go SDK v3.0.1 manages and enforces row-level security policies across read, write and bulk-import paths, so multi-tenant RAG stops depending on every caller remembering to add a tenant filter. FastMCP v4.2.5 tightens OAuth/OIDC defaults with refresh-token theft detection and stricter redirect URI matching, closing the two usual ways an MCP server's auth gets abused.

Milvus · FastMCP

DEPLOY

Run speech in and out on the same local box as the model

KoboldCpp v1.125.1 ships TTS via `--wavtokenizer` plus Qwen3ASR/SenseVoice transcription with word-level timestamp alignment, and now enforces API keys on generation endpoints. aegis-stack v0.21.0 adds on-device TTS/STT through Kokoro and NVIDIA Parakeet. Voice pipelines that previously meant a round trip to a hosted speech API can stay on the machine.

KoboldCpp · aegis-stack

BUILD

Read tool calls the same way regardless of which provider answered

ai-microcore v6.10.0 exposes provider-agnostic `tool_calls` and `finish_reason` on responses from OpenAI, Anthropic and Gemini, removing the per-vendor parsing branch most agent loops carry. cotal

> adds Agent Client Protocol support so clients like T3 Code and Zed can spawn, drive and interrupt agents through `cotal acp` rather than a bespoke wrapper.
>
> ai-microcore

Does **Opus 5** read better? 👍 👎

OFFENSIVE SECURITY

◆ Exploitation & C2

xalgorix

Feedback ⓘ v4.6.205 NOTES ↗ RANK

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

> xalgorix made scan configuration changes apply without restarting the engine and made build_plan rebuilds preserve prior progress, while also reworking its skill catalog — filtering source-review skills out of black-box scans, adding an opt-in adaptive phase playbook delivery mode, and slimming the bundled catalog from 865 to 330 skills with new aliases and categories.

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

01 Source-aware skill filtering for black-box scans

Not checked yet

85

BREAKING

`search_skills`, `list_skills`, and `read_skill` now automatically filter source-review skills (SAST, secret scanning, dependency scanning, CI/CD, supply-chain) out of black-box scans, surfacing them only when source code is attached. As a result, black-box scans no longer see these skills in `search_skills` or `list_skills`, and `read_skill` declines to load them and explains why.

— Names affected functions and skill categories, mechanism clear v4.6.204

02 Adaptive phase playbook delivery

Not checked yet NEW

85

An opt-in `XALGORIX_PHASE_PLAYBOOKS` setting (off by default) enables adaptive phase playbook delivery, sending a phase's playbook whenever the engine detects active tasks or matching vulnerability-class testing.

Opt in to adaptive phase playbook delivery so the agent receives playbooks for injection, SSRF, and access-control phases even when early recon tasks are still open.

```
$ XALGORIX_PHASE_PLAYBOOKS=true xalgorix scan <target>
```

— Exact env var plus a runnable example command v4.6.204

03 Skill catalog slimmed to web/API focus

Not checked yet BREAKING

85

The bundled skill catalog was slimmed from 865 to 330 skills (14 MB to 5.1 MB) by removing 29 non-web categories, so `search_skills` results focus on web

applications, APIs, identity, recon, cloud exposure, and AI features. If a scan previously loaded any of the 29 removed non-web skill categories by name, `read_skill` now reports them as not found.

— Concrete size/count numbers and explicit breaking behavior v4.6.204

---

04 **Hot-reload specialist and checklist settings** Not checked yet

IMPROVED

80

Specialist and checklist settings now take effect for new scans without an engine restart. Set `XALGORIX_DISABLE_AUTO_DELEGATE` to disable the specialist wave or `XALGORIX_CHECKLIST` to change the scan checklist from the Environment settings page.

— Names exact env vars and settings location, no runnable command v4.6.205

---

05 **build_plan rebuild reliability** Not checked yet IMPROVED

75

A `build_plan` rebuild now preserves plan progress across rebuilds, keeping completed and skipped task statuses for tasks that share the same id, vulnerability class, and endpoint instead of resetting all tasks to pending. It also no longer shrinks the endpoint surface by replacing known endpoints with only the agent's inventory note paths — the inventory note is now merged into the known surface.

— Clear before/after behavior but no direct command to invoke v4.6.205

---

06 **New skill aliases and category placement** Not checked yet NEW

65

The SSRF skill is now accessible directly via alias `ssrf` and the SAML skill via alias `saml_attacks`; the nmap recon skill is now available under the `reconnaissance` category.

— Names exact aliases and category but minimal mechanism v4.6.204

└──▷ **BREAKING ON UPGRADE**

! If a scan previously loaded any of the 29 removed non-web skill categories by name, `read_skill` now reports them as not found.

! Black-box scans no longer see source-review skills (SAST, secret scanning, dependency scanning, CI/CD, supply-chain) in `search_skills` or `list_skills`; `read_skill` declines to load them and explains why.

WAS THIS USEFUL? 👍 👎

BUILD

◆ AI Coding Agents

Warp

Feedback ⓘ 2026.11.19 (v0.2026.11.18.08.17) NOTES ↗ RANK

Warp is an AI terminal that runs commands and helps developers build software.

> Warp added persistent environment variables for cloud agent runs and extended its Factory configuration system with deferred repository checkouts and cross-repo skill loading.

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

01 Environment variable persistence for cloud agent runs

Not checked yet NEW 70

Developers can persist environment variables across cloud agent setup commands by writing `KEY=VALUE` lines to the `$WARP_VARS_FILE` file, letting secrets or config values set in one setup command flow through the rest of the run without hard-coding them.

Pass a secret or config value from a cloud agent setup command into the rest of the run without hard-coding it.

```
$ echo 'MY_API_KEY=abc123' >> $WARP_VARS_FILE
```

— Names exact env var and shows a runnable command 2026.11.19 (v0.2026.11.18.08.17)

THINNER COVERAGE BELOW

02 Factory configuration: deferred checkouts and skill loading

Not checked yet NEW 35

Factories can be configured to defer repository checkouts until an agent actually needs them using factory files, and can load skills from other repositories in a Factory by adding skill reference files.

— Describes two Factory additions without naming exact config keys

2026.11.19 (v0.2026.11.18.08.17)

└──▷ ALSO FROM THESE RELEASES

WAS THIS USEFUL? 👍 👎

wrongstack

Feedback ⓘ v1.7.18 NOTES ↗ RANK

WrongStack is an AI coding agent that reads code, edits files, runs commands, and reasons through bugs across CLI, TUI, and browser interfaces.

> WrongStack added fine-grained ACP session mode control and MCP system-prompt instructions, relocated Code Intelligence into its own package, and restored desktop installers for all platforms while pinning quick sessions' tool surface and shell.

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

01 **ACP session mode control** Not checked yet NEW 88

Control the ACP session mode via `session/set_mode`, `wstack acp serve --yolo`, or `wstack acp serve --mode` to switch between `default`, `plan`, `auto-edit`, and `yolo` modes, with permission prompts that now clarify what always covers.

Start an ACP server in yolo mode so the agent applies edits without prompting — useful in a trusted CI environment.

```
$ wstack acp serve --yolo
```

Start an ACP server in plan-only mode so the agent proposes changes but never edits files autonomously.

```
$ wstack acp serve --mode plan
```

— Names exact commands, flags, and all four modes. v1.7.16

THINNER COVERAGE BELOW

02 **MCP server instructions in system prompt** Not checked yet NEW 45

Pass MCP server instructions into the system prompt over ACP and via `--mcp-config`.

— Names flag but lacks example or mechanism detail. v1.7.16

03 **Code Intelligence relocated with per-server tool groups**

Not checked yet IMPROVED 43

Code Intelligence now lives in the `@wrongstack/codebase` package, and each MCP server gets its own Quick tool group.

— Names the package but no usage path given. v1.7.16

04 **Tool surface and shell pinning in quick sessions** Not checked yet

NEW

28

Quick sessions now pin their tool surface and shell for the duration of the conversation.

— Describes behavior but no config key or command given. v1.7.18

05 **Desktop installers restored for all platforms** Not checked yet

IMPROVED

23

Desktop installers are available again for every platform, including all session modes and MCP server instructions introduced in v1.7.16.

— States availability but no install steps or platforms listed. v1.7.18

WAS THIS USEFUL? 👍 👎

corecoder

Feedback ⓘ v0.16.0 NOTES ↗ RANK

Corecoder is a minimal AI coding agent (~1,000 lines of Python) inspired by Claude Code. Works with any LLM.

> CoreCoder's only shipped change this window is bounded, best-effort cooperative cancellation for MCP requests over stdio and Streamable HTTP transports, with several edge cases hardened to avoid duplicate side effects or orphaned cancels.

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

01 **Cooperative cancellation for MCP stdio/HTTP requests**

Not checked yet IMPROVED 64

CoreCoder adds bounded, best-effort cooperative cancellation for stdio and Streamable HTTP MCP requests. `initialize` requests are never cancelled and the original tool call is never replayed after a timeout, preventing duplicate side effects. Late stdio replies that arrive after the waiter has gone away are dropped to keep the stdio transport clean across manual reconnects, and stdio cancellation stays bound to the original process across those reconnects so a reconnect cannot orphan an in-flight cancel. HTTP cancellation is skipped when a connection fails before the request is sent, avoiding spurious cancel traffic.

— Clear mechanism and edge cases but no config/flag to invoke it v0.16.0

WAS THIS USEFUL? 👍 👎

Yuval Sarel cones

Feedback ⓘ v0.4.28 NOTES ↗ RANK

Cones is a terminal dashboard for managing coding-agent sessions across projects.

> Cones reworks its session-list grouping for worktrees, making nested grouping the default and remapping older config values accordingly.

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

01 Nested worktree grouping in session list

Not checked yet BREAKING

87

Setting `worktrees: nested` in config — now the default — groups worktree sessions inside their repository in the session list, giving worktrees with two or more sessions a bold `⑂ name` heading that folds like a tag, while a lone worktree session sits among the repository's own sessions with its worktree name shown dim. Setting `worktrees: folder` instead keeps every worktree as its own top-level folder. The previous `repo` and `flat` config values are now read as `nested`, so configs relying on the old `repo` or `flat` behaviour will silently switch to the new nested layout. Sessions moved to a tag, and launches listed under their launcher, no longer count toward a worktree's heading in the session list.

Opt into the new nested worktree grouping explicitly, or switch back to the flat-folder view, by setting `worktrees` in your cones config.

yaml

```
worktrees: nested
```

— Names exact config key, values, file and default-remap behaviour. v0.4.28

└──▷ **BREAKING ON UPGRADE**

! The `repo` and `flat` `worktrees` config values are now read as `nested`; any config relying on `repo` or `flat` behaviour will silently switch to the `nested` layout.

WAS THIS USEFUL? 👍 👎

forgedock

Feedback ⓘ v1.63.0 NOTES ↗ RANK

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

> ForgeDock shipped stacked-lanes orchestration for parallel planning and building of dependent issues, alongside a new GitHub Actions lane provider and a round of lane-infrastructure hardening: CI budget gating, computeshare reservations, bounded cs-cache volumes, and outcome-based verification.

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

01 **Stacked-lanes orchestration for dependent issues** Not checked yet

NEW

80

Stacked-lanes phase 1 lets forgedock plan dependent issues at `t=0` by setting `orchestration.stacking: plan` in `forge.yaml`, reducing orchestration latency instead of waiting on the dependency. Phase 2 builds the dependent issue on the dependency's HEAD once it reaches PR-ready, adding stacked PR support.

Enable stacked-lanes planning so dependent issues start planning immediately at `t=0` rather than waiting, speeding up large batches.

yaml

```yaml
orchestration:
  stacking: plan
```

— Names config key and two-phase mechanism with timing. v1.63.0

02 **CI budget gate for lane workflow changes** Not checked yet NEW

60

A new CI budget gate blocks lanes from adding workflows or CI jobs unless they declare a trigger scope and cost.

— Describes gating condition but not the declaration syntax. v1.62.0

THINNER COVERAGE BELOW

03 **GitHub Actions lane provider** Not checked yet NEW 55

ForgeDock adds a GitHub Actions lane provider so that public repos now run lanes on hosted runners by default.

— Names provider and default behaviour change but no config detail. v1.62.0

04 **Computeshare priority reservation model** Not checked yet NEW

55

A computeshare priority reservation model lets one host run a reserved workload and lend its remaining capacity to lanes.

— Explains mechanism, no config key or command given. v1.62.0

05 **Bounded cs-cache volumes** Not checked yet IMPROVED 50

`cs-cache` volumes now have bounds, including a cache sweep, a size cap, and idle eviction.

— Names the surface and limits but no commands to tune it. v1.62.0

06 **Shellcheck lane-tools recipe** Not checked yet NEW 50

Adds a `shellcheck` lane-tools recipe; remote lanes park `env.tool_missing` when `shellcheck` is unavailable or validation requires it.

— Names the env key but mechanism is only briefly stated. v1.62.0

07 **Agent container namespacing by agent ID** Not checked yet IMPROVED 45

Agent containers, volumes, and reapers are now namespaced by agent ID to isolate pool resources per agent.

— States mechanism but no config surface or scope detail. v1.63.0

08 **Clawd splash screen on install** Not checked yet NEW 40

Adds a Clawd splash screen shown on `npx forgedock` install.

— Trivial cosmetic change with exact command named. v1.62.0

09 **Outcome-based verification messaging** Not checked yet IMPROVED 25

Verification now shows that the outcome improved during verification, rather than only confirming that a change exists.

— Vague prose with no mechanism or surface named. v1.62.0

WAS THIS USEFUL?  

yetone magpie

Feedback ⓘ v0.1.4704 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.

> Magpie's latest releases add fine-grained reasoning-effort control for subagents, per-model-family rate-limit tracking for Claude accounts, native web-search passthrough for partner presets, more transparent MCP hub access filtering, and a shift to magpie-native metrics for Discover rankings.

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

01 Reasoning effort control for subagents

Not checked yet NEW

90

A subagent's reasoning effort can now be set via the `effort` field in Claude Code agent frontmatter (accepted values: `low`, `medium`, `high`, `xhigh`, `max`, or a number) and via `model_reasoning_effort` in Codex role files, letting an agent run above its default instead of spending more tokens than needed. Codex subagent effort is validated against the model's supported levels before saving, so an unsupported level is refused before reaching the agent.

— Exact field names and accepted values given, directly usable. v0.1.4682

02 Native web-search passthrough via Searches preset field

Not checked yet NEW

85

Adds `Searches bool` to `PresetDef` in `internal/provider/presets.go` so a partner preset can declare it handles web search natively, causing the agent's `web_search_20250305` / `web_search` tool to be forwarded as-is rather than routed through magpie's own searcher. Also adds a `PresetSearches(p Provider) bool` function in the same file to let callers check at runtime whether a key+URL provider's preset declares native web-search support.

— Exact field, file path and function named; internal API not user command. v0.1.4697

03 Per-model-family rate-limit tracking for Claude accounts

Not checked yet NEW

80

Adds per-model-family rate-limit tracking to the `claude-limits.json` state file via a new `models` map field, so a Claude account hitting its Fable limit is bypassed only for sessions on that model family and remains available for Opus and other families.

— Names exact state file and field but no CLI/UI path shown. v0.1.4700

04 Hub access filtering in initialize response

Not checked yet NEW

70

The hub's `initialize` response now tells a caller which MCP servers were withheld, grouping them by reason — `program`, `here`, `browser`, or `memory` — and separately counts servers the calling key may not use at all via `access.Key.MCP`, without naming those servers.

— Names exact reasons and field but no example call shown. v0.1.4704

THINNER COVERAGE BELOW

05 **Discover rankings based on magpie usage and likes** Not checked yet

IMPROVED

40

Discover's weekly top-plugin rankings now use in-magpie usage count and in-magpie likes instead of npm downloads or GitHub star counts, so rankings reflect the actual magpie community rather than any npm user or inherited repository stars.

— Describes the change but no named metric, config or command. v0.1.4701

NEW INTEGRATION TRAE CN target support

v0.1.4698 CODE ↗ STABLE

└──▷ GET THIS VERSION ▾

Docker

```
$ docker pull ghcr.io/yetone/magpie:0.1.4698
```

git

```
$ git clone --branch v0.1.4698 https://github.com/yetone/magpie.git
# already have the repo? check out this version:
$ git checkout v0.1.4698
```

published under the project's GitHub account · checked 23 Nov Wrong command?

WAS THIS USEFUL? 👍 👎

agentbro

Feedback ⓘ v3.3.0 NOTES ↗ RANK

Agentbro is a macOS Dynamic Island for AI coding agents: Claude Code, Codex, Gemini CLI, Cursor, hooks, skills, and local agent management.

> AgentBro's v3.3.0 release adds portable Skill Pack export/import with hash-based diffing and lockfile recovery, brings WSL, DeepSeek Harness, QwenWork CN, and WorkBuddy connections into the Windows Island, and adds Homebrew installation plus SSH-host Agent management and MCP Inspector protocol improvements.

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

01 Skill Pack export, import, and Git update review

Not checked yet

NEW

85

AgentBro adds export of Skill Packs as versioned JSON manifests and import from a local file or a public GitHub file link, showing `SHA-256` content differences and missing-member review before import; `skills-lock.json` can be inspected from `Settings > Projects` to check installed Skills and recover missing entries from a pinned GitHub revision or an installed npm dependency. It also lets you review which files will be added, changed, or removed before applying a Git Skill update, and import Skills directly from public Markdown or ZIP download links.

Recover a Skill Pack on a new machine by importing the versioned manifest exported from your previous host.

> 📍 In AgentBro, go to Skill Packs, choose Import, and open the exported JSON manifest file (or paste a public GitHub file link). Review missing members and SHA-256 differences, then confirm to restore.

— Names exact file, UI path, hash mechanism, and import formats v3.3.0

02 WSL, DeepSeek Harness, QwenWork CN, and WorkBuddy connections

Not checked yet NEW

80

Install Claude Code or Codex CLI Hooks for a registered WSL distribution so WSL sessions appear in the Windows Island with their distribution name and a link back to the Linux project directory; connect `DeepSeek Harness` on Windows via its `AgentBro` Hook to report live sessions and eligible approvals in the Island; connect `QwenWork CN` in Agent Management to show task activity in the Island, with confirmations handled inside `QwenWork CN` itself; and connect `WorkBuddy` and `WorkBuddy AI` on Windows using separate desktop profiles, with `WorkBuddy AI` confirmation notices linking back to the desktop app.

Bring WSL-based Claude Code or Codex CLI sessions into the Windows Island so you can approve and track them alongside native sessions.

> 📍 In AgentBro, open Island settings, select your registered WSL distribution, and install the Claude Code or Codex CLI Hook for it. WSL sessions will appear in the Island labelled with the distribution name.

– Enumerates every named integration and its Hook path v3.3.0

---

03 **Homebrew install for macOS** Not checked yet NEW 65

AgentBro can now be installed on macOS via Homebrew with `brew tap shirenchuang/tap && brew install --cask agentbro`.

Install AgentBro on macOS using Homebrew for managed, repeatable installs.

```
$ brew tap shirenchuang/tap && brew install --cask agentbro
```

– Exact runnable command given v3.3.0

THINNER COVERAGE BELOW

04 **Agent management on SSH hosts** Not checked yet IMPROVED 55

Manage custom Agents, their Hooks, native plugin settings, and Codex MCP servers on a selected SSH host; the Agent sidebar can be searched by name and shows installation status from current detection.

– Names SSH host scope and Codex MCP but no steps v3.3.0

---

05 **DeepSeek Harness reconnect after Hook update** Not checked yet 55

BREAKING

If `DeepSeek Harness` is already connected, you must quit it, remove and reinstall its Hook in AgentBro, then reopen Harness.

– Exact migration steps given for existing connections v3.3.0

---

06 **MCP Inspector protocol and stream handling** Not checked yet

IMPROVED 50

MCP Inspector now handles negotiated capabilities, pagination, complete stdio and HTTP streams, and cancellation of the original SSH operation.

– Lists mechanisms but no usage steps v3.3.0

└──▷ BREAKING ON UPGRADE

! If DeepSeek Harness is already connected, you must quit it, remove and reinstall its Hook in AgentBro, then reopen Harness.

WAS THIS USEFUL? 👍 👎

Alibaba OpenCodeReview

Feedback ⓘ v1.14.2 NOTES ↗

RANK

> OpenCodeReview added Turkish-language support across its README, site documentation, and VS Code localization package.

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

01 **Turkish localization support** Not checked yet NEW 25

OpenCodeReview now supports Turkish in the README, site documentation, and VS Code localization package.

— Names surfaces but no mechanism or detail beyond language addition v1.14.2

WAS THIS USEFUL? 👍 👎

prflow

Feedback ⓘ v2.65.0 NOTES ↗ RANK

prflow is a tool that turns feature requests into review-ready pull requests with implementation, tests, review loop, and docs.

> prflow's v2.65.0 release reworks the `/prflow:spec` workflow with optional claim-by-claim verification, fewer clarifying questions, and approval-gated filing, while Implement gains automatic run criteria for uncovered behaviors and `/prflow:docs-verify` drops its `--report-only` flag.

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

01 **`--report-only` flag removed from `/prflow:docs-verify`**

Not checked yet BREAKING 75

The `--report-only` flag is removed from `/prflow:docs-verify`; passing it now causes the command to refuse the run. Drop the flag — the command now reports only by default.

— Exact flag and command named with clear migration instruction v2.65.0

02 **Claim-by-claim draft verification in /prflow:spec** Not checked yet

NEW 60

`/prflow:spec` now offers to check its draft claim by claim against your code before filing; each run asks once whether to run the check, which adds time and cost to that run.

— Names the command and mechanism but no example shown v2.65.0

THINNER COVERAGE BELOW

03 **Fewer clarifying questions in /prflow:spec** Not checked yet

IMPROVED 55

`/prflow:spec` now asks fewer clarifying questions, usually settling choices in one round, and lists the decisions it made itself so you can overrule them before filing.

— Describes behavior change with before/after but no concrete syntax v2.65.0

04 **Approval-gated filing in /prflow:spec** Not checked yet BREAKING 50

`/prflow:spec` now files only after you approve the draft, removing the previous non-interactive filing route.

— Clear before/after but no migration detail beyond the removal v2.65.0

05 **Implement adds run criteria for uncovered outcomes**

Not checked yet · IMPROVED · 50

Implement adds a run criterion for an uncovered Desired Behavior outcome and keeps going; the workpad quotes the sentence the criterion came from.

— Explains mechanism but no concrete command or flag to act on v2.65.0

└──▷ **BREAKING ON UPGRADE**

! The `--report-only` flag is removed from `/prflow:docs-verify`; passing it now causes the command to refuse the run. Drop the flag — the command now reports only by default.

WAS THIS USEFUL? 👍 👎

jcode

Feedback ⓘ v1.1.1 NOTES ↗ RANK

Jcode is a high performance coding agent harness written in Rust, built for heavy multi-agent dev workflows

> jcode's latest release focuses on streamlining onboarding with new TUI login and subscription commands, while also boosting token plan inference credits and improving build distribution for Linux.

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

01 **TUI login and subscription commands** Not checked yet NEW

72

Users can access one-step browser login and contextual subscription nudges via the `/subscribe` and `/cloud` commands in the TUI. The `/login` starter also offers direct login via top provider shortcuts for OpenAI, Anthropic, Gemini, API key, and Auto Import.

— Names exact commands and provider shortcuts, no deeper mechanism given v1.1.1

THINNER COVERAGE BELOW

02 **Real semver releases for Linux x86_64 self-dev builds**

Not checked yet IMPROVED

40

Linux x86_64 self-dev builds are now published as real semver releases with paginated platform fallback.

— Describes change but lacks concrete flags or commands v1.1.1

03 **Doubled token plan inference credits** Not checked yet IMPROVED

37

jcode now doubles token plan inference credits from 2× to 4× for all plans.

— Concrete numbers but no mechanism or user action v1.1.1

WAS THIS USEFUL? 👍 👎

pstack-claude

Feedback ⓘ v0.9.151 NOTES ↗ RANK

pstack-claude is a port of Poteto's pstack agent framework to Claude Code, Codex, Copilot, Pi, and other AI coding assistants.

> pstack-claude ships a standalone `open-pstack` installer distributed via npm and Homebrew, adds an interactive `hc` terminal mode for Pi agents, and fixes `reflect` skill routing across Pi, Codex, and OpenCode, alongside documented REST fallback reads for CI status when GitHub GraphQL is blocked in cloud sessions.

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

01 **open-pstack installer via npm and Homebrew** Not checked yet NEW

95

Install pstack without cloning the repository using `npx open-pstack` from npm or `brew install michael-denyer/tap/open-pstack` from Homebrew; the tool lists agent harnesses on `PATH`, lets you pick one, and runs each harness's README recipe. Unattended runs are controlled with `--only`, `--yes`, `--dry-run`, and `--list` flags, and the installer can set up Muse, Grok Build, MiniMax Code, Antigravity, and the shared `~/.agents/skills` directory directly from the plugin copy staged at `~/.local/share/open-pstack/pstack`, removing the need for a repository clone.

— Runnable install commands and flags given in full v0.9.149

02 **Interactive hc terminal mode for Pi agents** Not checked yet NEW

85

Run Pi agents in an interactive `hc` terminal instead of a hidden RPC process by adding `pi agent host: hc` to the sheet, letting users watch and type into each child agent in real time. Child agents in `hc` mode now receive the parent's environment through a 0600 file instead of argv, and `stop_agent` sends SIGTERM to the terminal's process group then SIGKILL after `killGraceMs`.

— Exact sheet config key and signal sequence given v0.9.147

03 **reflect command skill routing across Pi, Codex, OpenCode**

Not checked yet IMPROVED

80

The `reflect` command now writes edited pstack skills to `.pi/skills/<skill>/` or the Pi agent directory's `skills/` folder on Pi, replacing the package copy rather than the unused `.claude/skills/` path. On Codex it writes edited skills to `.agents/skills/<skill>/` or `${CODEX_HOME:-~/.codex}/skills/<skill>/`, beside the plugin's `pstack:<skill>`. On OpenCode it now reads full session

transcripts — including compacted turns — via `opencode session list --format json` and `opencode session export`.

— Every path and command named, but routing fix not demonstrated v0.9.147

04 **REST fallback for CI status reads in cloud sessions**

Not checked yet NEW 75

`cloud-session-pr-reads.md` documents REST fallback reads for use when GitHub GraphQL is blocked in Claude Code cloud sessions, covering an Actions run's jobs, one job's log, `gh run view`, and `gh run list`, and explaining how a check run's `status` and `conclusion` fields give its result. When `watch-pr` exits with code 7 and `graphql-refused` in a cloud session, the babysit playbook now points the agent at this doc, which lists the REST reads available and their limits.

— Named doc and gh commands but no end-user walkthrough v0.9.151 v0.9.147

05 **Pi extension shares model sheet with Claude Code** Not checked yet

NEW 60

The Pi extension now reads `.claude/pstack-models.md` and replaces the user's sheet in that project, letting teams that run both Pi and Claude Code maintain one shared model sheet per project.

— Names the shared file but no further detail on sync behavior v0.9.149

THINNER COVERAGE BELOW

06 **npm package provenance via trusted publishing** Not checked yet

NEW 40

The npm package is now published with provenance through trusted publishing by pushing a release tag.

— One-line mechanism with no further operational detail v0.9.149

WAS THIS USEFUL? 👍 👎

keploy

Feedback ⓘ v3.7.16 NOTES ↗ RANK

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

> Keploy v3.7.16 reduces test-capture noise by filtering out static assets and binary downloads during recording, and improves MongoDB test reporting with clearer operation names.

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

01 **Static asset and binary filtering during recording**

Not checked yet IMPROVED 50

The `record` command now ignores static assets and binary downloads by default, reducing noise in captured test cases.

— Names the command but not which asset types or how to override v3.7.16

02 **MongoDB operation names in test summaries** Not checked yet

IMPROVED 35

Report summaries now derive the MongoDB operation name from the request message, giving clearer test-run output for Mongo traffic.

— Describes the change but no example output or mechanism detail v3.7.16

WAS THIS USEFUL? 👍 👎

Letta Code

Feedback ⓘ v0.34.69 NOTES ↗ RANK

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

> Letta Code's task CRUD tools now return full task snapshots instead of partial data, giving agents richer context on task state.

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

01 **Full task snapshots in CRUD tools** Not checked yet IMPROVED

28

Task CRUD tools now return full task snapshots instead of partial data, giving agents richer context on task state.

— Describes the change but no mechanism, endpoint or example given v0.34.69

WAS THIS USEFUL? 👍 👎

agentty

Feedback ⓘ v0.9.62 NOTES ↗ RANK

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

> Agentty v0.9.62 adds an interactive clarifying-questions card for ambiguous requests and substantially hardens sandboxing — isolating MCP servers, ACP agents and retrieval helpers, refusing to start when a sandbox can't be built, protecting `~/.agentty`, and extending protections to hosts without user namespaces with new diagnostics and kernel audit logging.

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

01 **Refuse to start unsandboxed on sandbox build failure**

Not checked yet 80

BREAKING

Servers or sandboxes that cannot be built are now refused rather than started unsandboxed, with the reason given; passing `--sandbox on` now refuses to start if the configured sandbox posture cannot be built on the current host.

— Names exact flag and resulting behavior change v0.9.62

02 **Protection of ~/.agentty from agent file tools** Not checked yet

NEW 80

agentty now protects `~/.agentty` from agent file tools under all Trust handoff settings, including `allow`, and through symlinks, with sandboxed commands seeing it read-only even when the workspace is `$HOME`. Writing to another agent's server list — `.mcp.json`, `~/.claude.json`, `.codex/`, `.gemini/`, `.junie/` — is now treated as a trust handoff.

— Names every protected path and trust setting, but no manual command v0.9.62

03 **Sandbox pane diagnostics and kernel audit logging** Not checked yet

IMPROVED 75

The sandbox pane now lists blocked file paths and, on hosts without user namespaces, names exactly which protections are degraded (for example `other processes: visible`, 'protected files: tool gate only'). Sandboxing now also works on such hosts (for example Ubuntu 24.04 with AppArmor defaults) using Landlock alone, keeping secrets masked, read-only paths like `.vscode/tasks.json` and `.git/hooks` read-only, and routing network through agentty's proxy; on Linux 6.15+, the kernel's audit log now records each sandbox denial.

— Names exact host/kernel versions and paths; pane location not precisely navigated

v0.9.62

04 **Clarifying-questions card for ambiguous requests** Not checked yet

NEW 70

A new clarifying-questions tool presents up to four multiple-choice questions in a card when a request is ambiguous, with keyboard navigation (arrows, Enter, digit keys, Space for multi-select) and a review page before anything is sent. It can be disabled via `Ctrl+K → Settings → Ask clarifying questions`, which stops the tool from being offered to the model entirely.

— Exact settings path given; sending/trigger mechanism not detailed v0.9.62

---

05 **Sandboxing for MCP servers, ACP agents and retrieval helpers**

Not checked yet NEW 65

MCP servers, ACP agents, and retrieval helpers now each run in an isolated sandbox: MCP servers get the workspace and a proxied network, ACP agents additionally get their own sign-in directory, and retrieval helpers get a read-only workspace with no network.

— Clear per-surface scope but no manual trigger or flag given v0.9.62

WAS THIS USEFUL? 👍 👎

ogcode

Feedback ⓘ v0.60.0 NOTES ↗ RANK

ogcode executes code tasks via a local-first AI agent that manages tools, memory, permissions, and human oversight to turn intent into completed work.

> ogcode adds per-turn memory so the agent retains context across turns, backed by a new database table and a configurable context window.

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

01 **Per-turn memory via turn summaries** Not checked yet NEW

83

Each completed turn now gets a background-synthesized digest stored in a new `turn_summary` table (migration 082), giving the agent memory of what it just did across turns. The number of recent turn digests injected as context into the next turn is controlled by the `OGCODE_TURN_SUMMARY_WINDOW` environment variable, which defaults to two summaries.

Widen the agent's cross-turn memory window to five summaries when working on long, multi-turn tasks where two summaries lose too much context.

```
$ OGCODE_TURN_SUMMARY_WINDOW=5 ogcode
```

— Names table, migration number, and env var with default. v0.60.0

WAS THIS USEFUL? 👍 👎

flow-next

Feedback ⓘ flow-next-v9.20.0 NOTES ↗ RANK

flow-next is a workflow plugin that takes a bug, idea or ticket to a verified pull request with specs, cross-model review, live QA, and receipts

> flow-next's v9.20.0 release reorganizes CLI usage output behind a new `--orchestration` flag, adds dependency data to `flowctl specs --json` and a `--forge-only` snapshot flag, and tightens default behavior in memory, task-listing and strategy-drift recipes.

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

01 Orchestration recipes moved behind --orchestration flag

Not checked yet

91

BREAKING

`flowctl usage --orchestration` now carries the multi-CLI bridge and model-steering recipes that plain `flowctl usage` used to print; running `flowctl usage` alone no longer includes them. Moving the recipes shrinks the CLI guide from 17.8 KB to 4.2 KB, reducing agent context-window consumption. Existing `CLAUDE.md` or `AGENTS.md` blocks still work, but re-running `/flow-next:setup` updates the block to reference `flowctl usage --orchestration` directly.

Retrieve bridge and model-steering recipes when an agent is about to hand work to another CLI or pick a model, without loading the full CLI guide.

```
$ flowctl usage --orchestration
```

— Full mechanism, sizes and migration path given. flow-next-v9.20.0

02 --forge-only flag for pilot snapshot

Not checked yet NEW

78

`flowctl pilot snapshot --forge-only` resolves which forge a local checkout belongs to without triggering a full PR and spec listing, useful in `resolve-pr` and `land` workflows.

Resolve which forge a local checkout belongs to — useful in `resolve-pr` and `land` workflows — without triggering a full PR and spec listing.

```
$ flowctl pilot snapshot --forge-only
```

— Exact command and use case named. flow-next-v9.20.0

03 Dependency data in flowctl specs --json

Not checked yet · NEW

70

`flowctl specs --json` now returns each open spec's `depends_on_epics`, surfacing dependency relationships across all open specs in a single call instead of querying each spec individually.

Inspect dependency relationships across all open specs in one call, rather than querying each spec individually.

```
$ flowctl specs --json
```

— Named field and exact command but limited mechanism detail. flow-next-v9.20.0

---

04 Default limits and null semantics in memory module

Not checked yet · BREAKING

68

Skill calls to `memory search` with no limit now automatically pass `--limit 10`. In a `memory apply` plan, `set: {key: null}` now removes that key from the entry instead of writing a literal `null`.

— Two named memory-module defaults changed, no commands to run. flow-next-v9.20.0

THINNER COVERAGE BELOW

---

05 Strategy drift scan reads limited done specs

Not checked yet

40

BREAKING

The strategy drift scan now reads every open spec plus the 30 most recent done specs instead of all done specs.

— Gives a concrete limit but no actionable surface. flow-next-v9.20.0

---

06 Task listing recipe scoped to open work

Not checked yet · BREAKING

30

The 'what tasks are there' recipe now lists open work only, omitting done tasks and tasks under done specs.

— Brief behavior note with no mechanism or surface named. flow-next-v9.20.0

└──▷ BREAKING ON UPGRADE

! Plain `flowctl usage` no longer prints the bridge and model-steering recipes; run `flowctl usage --orchestration` to access them. Existing `CLAUDE.md` or `AGENTS.md` blocks still work, but re-running `/flow-next:setup` updates the block to name `flowctl usage --orchestration` directly.

- ! Skill calls to `memory search` with no limit now pass `--limit 10` automatically.
- ! The 'what tasks are there' recipe now lists open work only, omitting done tasks and tasks under done specs.
- ! The strategy drift scan now reads every open spec plus the 30 most recent done specs instead of all done specs.
- ! In a `memory apply` plan, `set: {key: null}` now removes that key from the entry instead of writing `null`.

WAS THIS USEFUL? 👍 👎

◆ Coding Agent Add-ons

Diagram Design

Feedback ⓘ changes since 2026-10-25 CODE ↗

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

Diagram Design adds box plot and arc diagram variants, a geometry verifier, PlantUML import, and six new diagram types including SBGN and event modeling

Adds a box plot variant to scatter diagrams for five-number summaries and an arc diagram variant to dependency diagrams for skip-span imports. Not checked yet

Adds a geometry verifier inside the skill package and a self-check that flags boxed text overflowing the width budget. Not checked yet

Adds import support for PlantUML sequence and class diagrams, and Mermaid `quadrantChart` blocks, redrawn as Quadrant diagrams. Not checked yet

Adds the SBGN process description diagram type, requirement traceability, event modeling, steps (an ordered stage rail), and unit grid as new diagram types. Not checked yet

Adds an offline option that uses system fonts so generated output works without loading Google Fonts. Not checked yet

---

WAS THIS USEFUL? 👍 👎

◆ Speech & Voice AI

Daily Pipecat

Feedback ⓘ v1.17.0 NOTES ↗

Pipecat is a framework for realtime voice and multimodal agents, orchestrating speech-to-text, model inference, and text-to-speech as a streaming pipeline.

Pipecat adds native SIP transport, LLM extensions middleware, Rime Prism speech-to-speech service, AWS Transcribe language ID, Deepgram Flux turn-endpoint control, and user waiting silence detection.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v1.17.0 https://github.com/pipecat-ai/pipecat.git
# already have the repo? check out this version:
$ git checkout v1.17.0
```

NEW INTEGRATION Add a native SIP transport with `pipecat-ai[sip]`: the SIPTransport class registers a SIP account or dials through a registration-less trunk, handles dial-in/dial-out with `on_dialin_ready`, `on_dialin_connected`, `on_dialout_answered`, start_dialout(), send_dtmf(), and sip_refer(), and supports DTMF in both directions, hold/resume, and VP8 video. Not checked yet

Configure SIP media-NAT traversal and debugging via SIPConnection(extra_params=..., jitter_buffer_mode=..., jitter_buffer_ms=..., native_log_level=..., sip_trace=...), or via environment variables `SIP_EXTRA_PARAMS`, `SIP_STUN_SERVER`, `SIP_JITTER_BUFFER`, `SIP_NATIVE_LOG_LEVEL`, and `SIP_TRACE=1`. Not checked yet

Run the development runner in SIP mode with `-t sip` to start an HTTP server with `POST /start` and `POST /daily-sip-trunk-webhook`; use `-t sip -d` to run one bot immediately with no server; use `SIP_NET_INTERFACE` to restrict the stack to one local interface. Not checked yet

Provision Daily SIP credentials from code with new `DailyRESTHelper` methods: create_sip_client(), list_sip_clients(), get_sip_client(), delete_sip_client(), and rotate_sip_client_password(), wrapping Daily's `/sip-clients` REST API. Not checked yet

Perform an attended SIP transfer with SIPTransport.sip_attended_transfer({"toEndPoint": "sip:..."}): the transport dials a consult leg, waits for answer, then sends a REFER with a Replaces header so the two peers connect directly. Not checked yet

+7 more

NEW INTEGRATION Rime Prism speech-to-speech integration

WAS THIS USEFUL?  

◆ Model Training & Fine-Tuning

fizgig

Feedback ⓘ v9.8.5 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's v9.8.5 release adds an AMD GPU performance toggle, speeds up queued Studio jobs that only differ by LoRA, and warns users when a second browser tab has taken over setup.

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

01 **FIZGIG_MIOPEN environment variable for AMD GPUs** Not checked yet

NEW

80

Setting the `FIZGIG_MIOPEN=0` environment variable disables MIOpen convolutions on AMD GPUs, falling back to PyTorch's own kernels instead; each stage's log reports whether MIOpen is active.

— Names exact env var and its effect, fully actionable v9.8.5

THINNER COVERAGE BELOW

02 **LoRA swapping for queued Studio jobs** Not checked yet IMPROVED

47

Studio now swaps the primary LoRA on the already-loaded model for queued jobs whose only difference is the LoRA, matching the behaviour Render gained in 9.8.4, and shows a Loading LoRA status while doing so.

— Describes mechanism but no config surface to act on v9.8.5

03 **Second-tab setup conflict notice** Not checked yet NEW

40

Studio now detects when a second browser tab has loaded its own setup over the first, and displays a notice in the first tab so users know their next Render will reload its own setup.

— Clear behaviour description but minor, no config v9.8.5

WAS THIS USEFUL? 👍 👎

◆ Image & Video Generation

Runway

Feedback ⓘ updated · Nov 23, 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 tool for building, branding and hosting custom creative apps powered by its models.

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

01 **App Studio for building custom creative apps** Not checked yet

NEW 40

A new interface lets users describe a creative app in plain language, and Runway builds, brands, and hosts it using Runway's own models combined with the user's own files.

— Describes capability in prose with no named flags, endpoints, or limits

snapshot-20261123

WAS THIS USEFUL? 👍 👎

AGENT

◆ MCP Servers & Tooling

Prefect FastMCP

Feedback ⓘ v4.2.5 NOTES ↗ RANK

FastMCP is a Python framework for building MCP servers and clients, handling transport, schema generation, authentication, and server composition.

> FastMCP v4.2.5 adds live subscription streams for push notifications, an OpenTelemetry-backed UsageMiddleware, an experimental Events extension with its own CLI, and tightens OAuth/OIDC security defaults including refresh-token theft detection and stricter redirect URI matching.

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

01 Live subscription streams via `subscriptions/listen`

Not checked yet 95

BREAKING

On 2026-07-28 connections FastMCP now serves `subscriptions/listen`, letting clients hold an open stream and receive tool, prompt, and resource change events pushed by FastMCP.notify_tools_changed() and its siblings; servers also now advertise `listChanged` and `resources.subscribe` and accept up to 1024 open streams per process by default. Disable this server-wide with `FASTMCP_ENABLE_SUBSCRIPTIONS_LISTEN=false` in the environment or `fastmcp.settings.enable_subscriptions_listen = False` in code before the server is created, which is recommended on serverless platforms or with small worker pools.

Opt out of live subscription streams on a serverless host where long-lived open requests are billed or time-limited.

```
$ FASTMCP_ENABLE_SUBSCRIPTIONS_LISTEN=false fastmcp run
my_server.py
```

— Names protocol version, env var, code setting, stream limit, exact command v4.2.5

02 Header and cookie handling for wrapped OpenAPI backends

Not checked yet 95

BREAKING

The new `forward_client_headers` option on `FastMCP.from_openapi`, `FastMCP.from_fastapi`, and `OpenAPIProvider` controls which client headers reach a wrapped OpenAPI backend, accepting `False` to forward none or a list of names to forward only those. As of this release, OpenAPI tools no longer forward MCP transport headers such as `mcp-protocol-version` to the wrapped API by default, and OpenAPI

cookie parameter values are now percent-encoded outside the characters cookies allow, so a backend reading a cookie containing `;`, spaces, or non-ASCII text must decode it.

Prevent an OpenAPI-backed server from leaking MCP transport headers to the upstream API, while still forwarding an allow-listed `x-request-id` header.

```python
from fastmcp import FastMCP

mcp = FastMCP.from_openapi(
    openapi_url='https://api.example.com/openapi.json',
    forward_client_headers=['x-request-id'],
)

if __name__ == '__main__':
    mcp.run()
```

— Named flag, three entry points, runnable example, exact header v4.2.5

---

03 **OAuth/OIDC security hardening** Not checked yet BREAKING

90

If neither `valid_scopes` nor `required_scopes` is set, CIMD clients can no longer request any scope, so `valid_scopes` must be set to the scopes you accept. Redirect URI matching is stricter: a URI must now match an exact allowed pattern's query component and may not carry a fragment, and an exact pattern at a host's root (e.g. `https://app.example/`) now matches only that exact path, requiring `/*` appended to allow any path. Presenting a refresh token more than 60 seconds after it was rotated now ends the whole grant as suspected theft, so clients sharing one stored refresh token between processes must share the rotated one; revoking a refresh token now also revokes that grant's access tokens.

— Enumerates every scope, redirect, and token behavior change with fixes v4.2.5

---

04 **AzureProvider multi-tenant issuer pinning** Not checked yet

75

BREAKING

Pin the token issuer in `AzureProvider` for multi-tenant authorities (`organizations`, `consumers`, or `common`) by passing `token_issuer`; without it, users from any tenant are now accepted.

— Exact parameter and tenant values, plus before/after behavior v4.2.5

---

05 **Experimental Events extension and CLI** Not checked yet NEW

70

Access the experimental Events extension (SEP-3415) via `fastmcp.experimental.events`, which supports poll and push delivery, along with a new `fastmcp experimental events list|watch` CLI.

— Names module, spec, and exact CLI subcommand v4.2.5

06 **UsageMiddleware for OpenTelemetry metrics** Not checked yet NEW 65

New `UsageMiddleware` records per-user, per-tool usage as the OpenTelemetry `mcp.server.operation.duration` histogram for `tools/call` and `prompts/get` calls; it should be added before any error-handling middleware.

— Names the metric and operations but no setup example v4.2.5

THINNER COVERAGE BELOW

07 **Dynamic client registration TTL** Not checked yet NEW 50

New `client_registration_ttl_seconds` setting on `OAuthProxy` and OIDCProxy lets unused dynamic client registrations expire from persistent storage.

— Names the setting but gives no usage example v4.2.5

└──▷ **BREAKING ON UPGRADE**

! If you set neither `valid_scopes` nor `required_scopes`, CIMD clients can no longer request any scope; set `valid_scopes` to the scopes you accept.

! A redirect URI must now match an exact allowed pattern's query component and may not carry a fragment.

! An exact redirect URI pattern at a host's root (such as `https://app.example/`) now matches only that exact path; append `/*` to allow any path.

! Presenting a refresh token more than 60 seconds after it was rotated ends the whole grant as a suspected theft; clients that copy one stored refresh token between processes must share the rotated one.

! Revoking a refresh token now also revokes that grant's access tokens.

! On 2026-07-28 connections, servers now advertise `listChanged` and `resources.subscribe` and accept up to 1024 open `subscriptions/listen` streams per process by default; servers on serverless platforms or with small worker pools should set `FASTMCP_ENABLE_SUBSCRIPTIONS_LISTEN=false`.

! OpenAPI tools no longer forward MCP transport headers such as `mcp-protocol-version` to the wrapped API.

! OpenAPI cookie parameter values are now percent-encoded outside the characters cookies allow; a backend that reads a cookie containing `;`, spaces, or non-ASCII text must decode it.

WAS THIS USEFUL? 👍 👎

api-mcp-server

Feedback ⓘ v4.15.1 NOTES ↗ RANK

MCP server that exposes Hostinger's platform API for hosting deployment, CDN control, domain management, and WordPress operations.

> api-mcp-server adds three new ecommerce webhook operations for retrieving, deleting, and listing webhooks.

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

01 **Ecommerce webhook management operations** Not checked yet NEW

65

Three new operations manage ecommerce webhooks: `ecommerce_webhooks_get` retrieves a single webhook, `ecommerce_webhooks_delete` removes one, and `ecommerce_webhooks_list` lists all configured webhooks.

— Names exact operations but no usage example or parameters v4.15.1

WAS THIS USEFUL? 👍 👎

memwhale

Feedback ⓘ v0.30.0 NOTES ↗ RANK

Memwhale is a persistent, local memory for developers and their coding agents. Records commands, output, errors, and the fixes that worked into SQLite and serves them over MCP.

> Memwhale introduces a knowledge memory system that imports assistant preferences from Claude Code, Codex, ChatGPT, and Claude, lets you review them before use, and exports approved statements into agent instruction files or Markdown.

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

01 **Knowledge import, review, and export workflow** Not checked yet

NEW 93

Pull preferences and standing instructions from Claude Code, Codex, ChatGPT, and Claude into a local review queue with `mw knowledge import`, then inspect proposed statements with `mw knowledge pending` and act on each by id or `all` using `approve`, `keep`, or `reject`. Write approved knowledge into an instruction file for Claude Code or Codex with `mw knowledge export --into <file>`, or omit `--into` to print a Markdown block for pasting into ChatGPT or Claude; add `--project <folder>` to scope the export to a single project so project-specific statements stay out of the global pool. The same import, review, and export workflow is also reachable in the desktop app via the new Knowledge page and its Import my conversations button.

Bring in everything you have typed to Claude Code and Codex, then review what was found before anything is used.

```
$ mw knowledge import
mw knowledge pending
```

Approve a proposed statement by id, then write all approved knowledge into the Codex instruction file so the agent sees it on the next run.

```
$ mw knowledge pending
# approve statement #2 shown above
mw knowledge export --into ~/.codex/AGENTS.md
```

Export only the knowledge that belongs to a specific project folder, keeping project-specific decisions out of other agents' context.

```
$ mw knowledge export --project ~/src/myproject --into
~/src/myproject/.claude/AGENTS.md
```

— Names every command, flag, and UI entry point with runnable examples. v0.30.0

WAS THIS USEFUL? 👍 👎

tokensave

Feedback ⓘ v7.17.0 NOTES ↗ RANK

Tokensave is the most comprehensive code intelligence MCP server for AI coding agents. 40+ tools, 30+ languages, 9 agent integrations.

> Tokensave v7.17.0 adds Open Knowledge Format bundle import, export, and validation, an index-annotated grep mode, confirmed/ambiguous/unresolved caller labeling, broader Swift dead-code detection, opt-in Caveman compatibility, and an explicit Cargo build-runner option for local dev builds.

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

01 OKF bundle import, export, and validation

Not checked yet NEW

95

Import Open Knowledge Format bundles into the project with `tokensave okf import <dir>`, list them with `okf list`, and remove with `okf remove`; `--strict` rejects non-conformant bundles. Validate OKF v0.2 bundles with `tokensave okf validate <dir>` (exits 1 on violations, `--strict` also fails on warnings, `--json` for machine output) and export the code index as a bundle with `tokensave okf export --output <dir>`. Imported concepts become searchable through the new `tokensave_knowledge` MCP tool in the `memory` area of `tokensave_more`.

Import a team-shared OKF knowledge bundle into the project state so agents can search concepts via `tokensave_knowledge`.

```
$ tokensave okf import ./bundles/threat-model --strict
```

— Full command set, flags, and exit codes specified. v7.17.0

02 Confirmed/ambiguous/unresolved labels on callers

Not checked yet

IMPROVED

80

`tokensave_callers` now labels every same-named call reference in a `references` field as `confirmed`, `other`, `ambiguous`, or `unresolved`; uncertain sites carry `to_read: true`. Results obey the `limit` parameter and report returned and total counts when truncated.

— Exact field and value names given, but no runnable example. v7.17.0

03 Opt-in Caveman compatibility mode

Not checked yet NEW

70

Enable cooperation with Caveman by setting `caveman_compatibility = true` in `~/.tokensave/config.toml`, or via `TOKENSAVE_CAVEMAN_COMPATIBILITY=1`.

Enable Caveman interop for a Claude Code workflow without touching per-project config.

```toml
caveman_compatibility = true
```

— Exact config key, file path, and env var given. v7.17.0

04 **Explicit Cargo build runner for dev builds** Not checked yet NEW

70

Select Cargo explicitly for local builds and tests with `--cargo` or `TOKENSAVE_BUILD_RUNNER=cargo` when using the new optional Blaze runner at `scripts/dev-build <cargo subcommand> [args]`.

— Exact flag, env var, and script path given. v7.17.0

05 **Swift out-of-body calls tracked in dead code analysis**

Not checked yet IMPROVED

65

`tokensave_dead_code` now records Swift calls made outside function bodies — property initializers, computed properties, top-level `main.swift` code, `deinit`, and `subscript` — requiring a reindex of Swift projects to pick this up.

— Names exact Swift constructs covered and the reindex requirement. v7.17.0

THINNER COVERAGE BELOW

06 **Index-annotated grep** Not checked yet NEW

50

`tokensave grep` annotates ripgrep hits with caller/definition facts from the code index, letting agents triage which references are confirmed calls versus stale without opening each file.

Annotate rg hits with caller/definition facts from the index to triage which references are confirmed calls vs. stale without opening each file.

```
$ tokensave grep rg --type rust 'authenticate'
```

— Command shown but underlying mechanism only briefly described. v7.17.0

WAS THIS USEFUL? 👍 👎

elementor-mcp

Feedback ⓘ v3.22.0 NOTES ↗ RANK

Elementor-mcp wordpress MCP server with 234 Free tools and 547 total with Pro for building pages, managing content, and operating your site. Availability depends on installed integrations and enabled tools.

> elementor-mcp's v3.22.0 release adds theme and page-builder integrations for Uncode and Bricks, new schema-markup and redirect management tools, and a partial-edit mechanism for large database values.

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

01 VacationRental structured data generation

Not checked yet NEW

95

Generate structured data for vacation rentals with `generate-schema-markup` using the new `VacationRental` type, which validates required fields (identifier, 8+ images, coordinates to 5 decimal places, occupancy) and supports `head` placement and `remove` on any post type.

Add vacation rental structured data to a property post type page, with validation of required fields before writing.

```
$ generate-schema-markup --post-id 99 --type VacationRental --
placement head
```

— Runnable command and validation rules both given v3.22.0

02 Partial edits for large database values

Not checked yet NEW

95

Edit part of a large database text value without resending the whole value by passing `edits` (`{ column: [ { find, replace, expect } ] }`) to `update-rows`, with `if_md5` to guard against concurrent changes.

Patch a specific phrase in a large post-meta value without reading and re-sending the full content — useful for large Elementor data blobs.

```
$ update-rows --table postmeta --where
"meta_key='_elementor_data' AND post_id=42" --edits
'{"meta_value": [{"find": "Old Headline", "replace": "New
Headline", "expect": 1}]}' --if_md5 '<md5-from-prior-read>'
```

— Exact parameter names and runnable example provided v3.22.0

03 **Rank Math redirect and robots management** Not checked yet NEW

85

Manage Rank Math redirects with the new `rankmath-read` and `rankmath-write` tools — read and write 301, 302, 307, 410, and 451 redirects with `exact`, `contains`, `starts-with`, `ends-with`, or `regex` sources, and set per-post-type robots defaults; requires Rank Math's Redirections module.

— Names tools, codes, and match types, no example call v3.22.0

04 **Uncode theme and Content Blocks management** Not checked yet NEW

80

Manage Uncode theme pages and Content Blocks with the new `uncode-read` and `uncode-write` tools, which list Wireframes, prepare placeholder images and forms, and read or change theme options. `uncode-write` ships disabled by default and requires Uncode and Uncode Core 2.12.8 or later.

— Names both tools and version requirements, no example call v3.22.0

05 **Disabled widget filtering across widget tools** Not checked yet

IMPROVED

75

`list-widgets` and `get-widget-schema` now mark disabled widgets, and `add-free-widget`, `add-pro-widget`, and the atomic add tools refuse disabled widgets with an actionable error.

— Names all five affected tools, no usage example v3.22.0

06 **Bricks templates in Templates library** Not checked yet NEW

65

Build pages from ten Bricks designs via the Templates library using a new Builder switch to filter between Elementor and Bricks templates; requires Bricks 2.4 or later.

— UI path given but no example command v3.22.0

THINNER COVERAGE BELOW

07 **Per-page JSON-LD listing tool** Not checked yet NEW

55

List per-page JSON-LD with the new read-only `list-page-schema` tool, which reports types, rating markup, and source.

— Names tool and outputs, no invocation example v3.22.0

WAS THIS USEFUL? 👍 👎

◆ AI Agent Frameworks

cotal

Feedback ⓘ v0.175.3 NOTES ↗ RANK

Cotal is the open standard for agent coordination

> Cotal's headline addition this window is native Agent Client Protocol (ACP) support — letting clients like T3 Code and Zed spawn, drive and interrupt agents via `cotal acp` — alongside broad reliability work: symlink-aware backup verification, self-update hardening, dry-run service previews, offline workflow simulation, and dozens of sharper error messages across join, auth, extension, endpoint and status commands.

└──▷ **WHAT SHIPPED · 22 FEATURES** most completely described first ? what's the number?

01 Agent Client Protocol (ACP) support

Not checked yet NEW

95

Connect T3 Code, Zed and other Agent Client Protocol clients to Cotal agents using the new `@cotal-ai/acp` extension and its `cotal acp` command, which serves ACP v1 on standard input and output. Stop an in-progress agent turn from an ACP client using the manager's new `interrupt` command (contract revision 53), which writes the seat connector's declared interrupt key — Escape for Claude Code and Pi — into the agent's terminal.

Expose a Cotal agent to an ACP-compatible client such as T3 Code or Zed so it can spawn and drive agents over standard input/output.

```
$ cotal acp
```

— names the extension, command, protocol version and interrupt mechanism v0.175.1

02 Catalog-read exit codes in `cotal models --harness`

Not checked yet IMPROVED

80

The `cotal models` reference now distinguishes a named catalog read from an all-catalog read: naming the one unreadable catalog with `--harness` fails with exit 5, while an all-catalog read with unreadable rows stays partial with exit 4.

Audit a specific model catalog and get a hard failure (exit 5) when it cannot be read, so CI pipelines can distinguish a single broken catalog from a partial all-catalog read.

```
$ cotal models --harness <catalog-name>
```

— exact exit codes and a runnable command are given v0.175.3

03 **CLI error messages and usage guidance improvements**

Not checked yet IMPROVED 80

`cotal chat --stdio` now accepts requests at the documented line limit when they end with CRLF. The CLI names misspelled nested subcommands and suggests the closest match, in both text and JSON usage errors. When `cotal join` cannot find a credential file, it now suggests a provisioned replacement that grants read access to the landing channel and explains how to use its new lifecycle UID. `endpoint show <name>` now refuses with `TARGET_AMBIGUOUS` naming each space and the retry for it when several registered spaces match the project folder, instead of a generic message with no names and no retry. Extension add now installs packages whose entry point is a directory, resolving it to the directory's index file instead of failing the import and rolling the install back. `cotal doctor auth` now shows the damaged registration file and its recovery steps even when `--space` names the registration. A managed spawn preview on an open or static space now refuses a `--on` value that cannot be a manager instance ID, instead of previewing it and failing only at launch. JSON watch and history follow streams now report an invalid broker URL or empty space selector with an error record and an end record. The `--server` flag is now documented as overriding the recorded broker in ACP. The preview and confirmation commands suggested by `cotal history clear` now keep your requested output format, so scripts can read the JSON receipt from the first purge. Startup now names the blocked directory component when `endpoint init` cannot scaffold, and asks for a reference JSON path when `endpoint add --file` is empty.

— names every flag, command and error code touched across two releases v0.175.3 v0.175.2

04 **Channel name and grammar handling in `cotal join`** Not checked yet

IMPROVED 78

`join` now accepts displayed channel names such as `#general` and refuses unsupported expiry flags before prompting sign-in to a user-auth space, while `#all` opens separately from the aggregate console feed and chat on channels starting with a hyphen reopens using the suggested command. It also validates the landing `--channel` grammar before dialing, rejecting invalid names with `USAGE_INVALID` instead of reporting missing mesh provisioning.

Join a channel using its display name (with the `#` prefix) so you do not need to look up the internal channel identifier.

```
$ cotal join #general
```

— names flags, error code and a runnable join example v0.174.4 v0.174.3

05 **Dry-run JSON preview for `cotal service start`** Not checked yet

NEW

75

Preview a service start with `cotal service start <name> --foreground --dry-run --json` to get one JSON preview object without starting the service.

Inspect what a service start would do in a CI pipeline before actually launching it.

```
$ cotal service start my-service --foreground --dry-run --json
```

— exact runnable command with all flags named v0.174.3

06 **Workspace and auth API additions for session tracking**

Not checked yet NEW

73

Filter channel history by publisher using the new `publisher` filter on `channelHistoryPage` in `@cotal-ai/core`, and list manager instances with the new optional `AuthProvider.listManagerInstances` implemented in `@cotal-ai/auth`. Track agent state across sessions with new `@cotal-ai/workspace` additions: `ownManagerInstances`, `userSignInAuth`, `signInChange`, an observer `lifecycleUid` option, and a `gone` list on the `ps` reply for registrations whose host is gone.

— names every API surface added but no usage example given v0.175.1

07 **Symlink-aware backup verify and inspect** Not checked yet IMPROVED

70

`cotal backup verify` and `cotal backup inspect` now report a missing manifest at the real backup directory when opened through a symbolic link, resolving the link instead of silently succeeding.

Verify a backup opened via a symlink — cotal now resolves the symlink and reports a missing manifest at the real directory rather than silently succeeding.

```
$ cotal backup verify
```

Inspect a backup archive accessed through a symlink to confirm the manifest is present at the underlying path.

```
$ cotal backup inspect
```

— names both commands with runnable examples v0.175.2

08 **Hyphen-prefixed channel names in `cotal auth mint`**

Not checked yet IMPROVED 70

The `cotal auth mint` correction now works for channel names that start with a hyphen, with the suggested `--allow-subscribe` or `--allow-publish` flag using `=` so the channel is read as its value.

Mint a credential that allows subscribing to a channel whose name starts with a hyphen, using `=` to prevent the name from being parsed as a flag.

```
$ cotal auth mint --allow-subscribe=<-channel-name>
```

— exact flags and a runnable example are included v0.175.3

09 **Passive console mode via `--no-input`** Not checked yet NEW

70

Run `cotal console --no-input` to open a passive line stream that shows output but accepts no console commands, even on an interactive terminal.

Tail a live session's output in a monitoring script without accidentally sending console commands.

```
$ cotal console --no-input
```

— exact runnable command with the flag named v0.174.3

10 **Earlier pre-submission control-call refusal** Not checked yet

IMPROVED 68

Catch pre-submission control-call failures earlier: a control call whose chosen run was replaced now reports `preSubmitRefusal` with `selectedSeatChanged` before submission, so `cotal attach` exits as `TARGET_NOT_FOUND` and `cotal team agent remove` reports refused instead of an unknown outcome.

— names exact error codes and the two affected commands v0.175.1

11 **Offline workflow simulation on open meshes** Not checked yet NEW

65

Workflow commands now offer offline simulation on open meshes, keep the selected route in checkpoint and receipt follow-ups, and ship a working simulation example. A workflow simulation can name a results file via `--handler` so a missing scripted result produces a clear usage error rather than a silent failure.

Run a workflow simulation and name a results file so a missing scripted result produces a clear usage error rather than a silent failure.

```
$ cotal workflow simulate --handler <results-file>
```

— describes behaviour and includes a runnable simulate command v0.174.3

12 **Utility commands: `personas rm` and `who --waiting`**

Not checked yet NEW 65

Remove a persona catalog entry without touching any same-named project file using `cotal personas rm`. Filter waiting participants with `cotal who --waiting` and get an explicit 'no participants match the filter' message when nothing is found.

— names both runnable commands and their exact behaviour v0.174.3

13 **Owner-scoped manager rights without admin token** Not checked yet

IMPROVED 63

Control every agent under your own owner account without the `admin` scope: a signed-in person's manager view now carries `inspect`, `ps`, and owner-mode `input`, `interrupt`, `despawn` and `attach` rights derived from the ledger row, not a scope token.

— explains the permission mechanism but gives no example to run v0.175.1

THINNER COVERAGE BELOW

14 **`cotal clean history --attempt` flag enforcement** Not checked yet

IMPROVED 55

Enforce `--attempt` flag rejection on `cotal clean history` so no retained messages are deleted when the flag is supplied without a valid target.

— names the exact flag and command, no runnable example given v0.174.4

15 **Manifest and log rotation handling improvements** Not checked yet

IMPROVED 55

Improves manifest handling: log rotation now accepts a path supplied via `--log-file`, and the CLI rejects extra topology paths and checkpoint-resume consent on fresh manifest starts.

— names the flag but behaviour description stays terse v0.174.4

16 **Extension and mesh management refinements** Not checked yet

IMPROVED 55

Allow removal of a scoped extension whose package name starts with an underscore, using the same name shown in the extension inventory. Let `meshes add` verify

unchanged unchecked registrations, and have `meshes rm` direct prepared spaces to `space forget`.

– names the commands involved but mechanism stays light v0.174.4

17 **Status and ps diagnostics improvements** Not checked yet IMPROVED

55

`cotal status` now recommends restoring an exited manager without stopping its broker, and `status` and `ps` name an occupied port or invalid channel seed before suggesting a foreground stack restart.

– names the commands and scenarios but no exact flags v0.174.3

18 **Single-task `once: true` helper in agent briefings**

Not checked yet NEW

50

Agent briefings now offer a single-task helper via `once: true` without listing which harnesses support it.

Use `once: true` in an agent briefing to dispatch a single task and have the manager automatically refuse connectors that cannot accept it.

```yaml
once: true
```

– names the key and shows usage but omits supported harnesses v0.174.4

19 **Linux broker-port TIME-WAIT recovery** Not checked yet IMPROVED

50

On Linux, restarting a controller stack now waits up to 65 seconds for TIME-WAIT to clear so a broker port released just after one minute can be reused without another command.

– names the exact timing but no command surface to invoke v0.174.4

20 **Self-update reliability improvements** Not checked yet IMPROVED

42

Self-update now uses a compatible installed Node.js to run npm when the Node.js running Cotal is too old for npm; waits for installed first-party connectors and extensions to reach the registry and keeps the existing installation if a required version is still missing; and finds npm in relative PATH entries from the folder where the command was run.

– describes mechanism across two releases but no command to run v0.175.3 v0.175.2

21 **Dashboard endpoints reject unsupported methods** Not checked yet

IMPROVED

40

Dashboard endpoints now return `405` for unsupported HTTP methods on dashboard read endpoints.

— names the status code but not the specific endpoint paths v0.174.3

22 **Startup validation for exchange ports and signing keys**

Not checked yet NEW

25

Validate public exchange ports and first-time identity provider signing keys before startup.

— a bare one-line description with no named surface v0.174.4

WAS THIS USEFUL? 👍 👎

◆ Agent Memory

claude-mem

Feedback ⓘ v16.1.0 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 v16.1.0 adds a standalone Cloud Sync option for cross-device memory syncing, offered as a paid add-on alongside existing memory providers.

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

01 **Cloud Sync option in installer** Not checked yet NEW

72

Selecting Cloud Sync during `npx claude-mem install` lets users sync memory across devices independently of their existing memory provider. It is offered as a $12/month plan with a 30-day trial or a $99/year plan.

Connect multiple machines to the same memory store without switching your existing local memory provider.

```
$ npx claude-mem install
```

— Names exact command and pricing, but no mechanism detail on sync itself. v16.1.0

WAS THIS USEFUL? 👍 👎

◆ Applied AI Agents

OpenClaw

Feedback ⓘ v2026.11.7 NOTES ↗ CODE ↗ RANK

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

> OpenClaw's v2026.11.7 release adds browser pairing via short code, saved-key live transcription, and dynamic ChatGPT model discovery, alongside gateway reliability work and three breaking plugin SDK changes.

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

01 Gateway method dispatch replaces mutable chat controllers

Not checked yet

85

BREAKING

Mutable `chatAbortControllers` and `chatQueuedTurns` access is removed; plugins must replace this with `dispatchGatewayMethod` from `openclaw/plugin-sdk/gateway-method-runtime` and declare `contracts.gatewayMethodDispatch: ['authenticated-request']` in the plugin manifest.

— Names old API, replacement function, and required manifest key. v2026.11.7

02 Session controller admission for setActiveEmbeddedRun

Not checked yet

75

BREAKING

Session-keyed setActiveEmbeddedRun(...) registrations must now move into the host-provided `runAttempt` callback; unadmitted registration reports `SESSION_CONTROLLER_ADMISSION_REQUIRED`.

— Exact symbol and error code given, migration path clear. v2026.11.7

03 AgentTurnStartOwner deadline field rename

Not checked yet

60

BREAKING

AgentTurnStartOwner.observe() consumers must update from `expiresAtMs` to `startDeadlineAtMs`; there is no compatibility alias, with migration guidance documented in `docs/plugins/sdk-migration.md`.

— Names old and new fields and points to migration doc. v2026.11.7

THINNER COVERAGE BELOW

04 Live transcription with saved provider API keys

Not checked yet

55

IMPROVED

Talk's live transcription can now use a saved Deepgram, ElevenLabs, or Mistral API key instead of requiring re-entry.

— Names the three providers but not the config location for the saved key. v2026.11.7

05 **Dynamic ChatGPT model catalog discovery** Not checked yet IMPROVED 45

Newly available ChatGPT subscription models can now be discovered without waiting for a new OpenClaw release, and sign-in discovery is now independent of npm.

— Explains the change but gives no command or setting to trigger discovery. v2026.11.7

06 **Browser pairing via short code** Not checked yet NEW 40

A new browser can be paired with the Gateway using a short code that is approved on the Gateway host, with clearer on-screen instructions shown while the approval is pending.

— Describes the flow but no exact UI path or command given. v2026.11.7

07 **Gateway reliability and resilience improvements** Not checked yet IMPROVED 30

Interrupted replies and child subagent results can now be recovered across Gateway restarts without losing pending work; chats, readiness checks, and unrelated sessions stay responsive during slow filesystem operations, large recovery packs, and page extraction; and Git and package updates now complete through schema changes, busy databases, and retained recovery state instead of stalling around a new release.

— Three thin reliability notes with no named flags, metrics, or limits. v2026.11.7

└──▷ **BREAKING ON UPGRADE**

! Session-keyed setActiveEmbeddedRun(...) registrations must move into the host-provided `runAttempt` callback; unadmitted registration now reports `SESSION_CONTROLLER_ADMISSION_REQUIRED`.

! Mutable `chatAbortControllers` and `chatQueuedTurns` access is removed; replace with `dispatchGatewayMethod` from `openclaw/plugin-sdk/gateway-method-runtime` and declare `contracts.gatewayMethodDispatch: ['authenticated-request']` in the plugin manifest.

! AgentTurnStartOwner.observe() consumers must update from `expiresAtMs` to `startDeadlineAtMs`; there is no compatibility alias. See `docs/plugins/sdk-migration.md` for migration guidance.

WAS THIS USEFUL? 👍 👎

◆ Agent Sandboxes

containarium

Feedback ⓘ v0.141.0 NOTES ↗ RANK

Containarium is an open-source agent runtime — SSH-native isolation, eBPF egress policy, Kubernetes + LXC backends, GPU passthrough, MCP-native CLI

> Containarium expanded its run audit trail to cover more call types and added exclusive run support to prevent concurrent execution conflicts.

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

01 **Audit trail covers Read, Grep, Glob calls** Not checked yet

IMPROVED 40

containarium now records Read , Grep , and Glob calls in the run audit, expanding audit trail coverage beyond shell execution.

— Names the exact calls but no mechanism detail. v0.141.0

02 **Exclusive runs with typed busy refusal** Not checked yet NEW

40

containarium now supports exclusive runs with a typed busy refusal, preventing concurrent execution conflicts on the same box.

— Describes behavior but no API or flag named. v0.141.0

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ Local LLM Runtimes

mlxtop

Feedback ⓘ v3.4.1 NOTES ↗ RANK

The mlxtop terminal UI monitors local LLMs on Apple Silicon, showing loaded models, memory use, GPU activity, and generation speed.

> mlxtop adds an opt-in environment variable to allow authenticated access to runtimes bound to the local machine's LAN or Thunderbolt addresses, not just loopback.

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

01 **Authenticated access to LAN/Thunderbolt-bound runtimes**

Not checked yet NEW 80

Setting `MLXTOP_ALLOW_REMOTE_AUTH=1` enables authenticated access to runtimes bound to this machine's own LAN or Thunderbolt addresses; without this variable set, credentials are still sent only to loopback.

Allow mlxtop to send credentials to a runtime bound to this Mac's LAN address, as in a cluster where each node serves its peers.

```
$ MLXTOP_ALLOW_REMOTE_AUTH=1 mlxtop
```

— Names exact env var and default behavior, with runnable example v3.4.1

WAS THIS USEFUL? 👍 👎

KoboldCpp

Feedback ⓘ v1.125.1 NOTES ↗ CODE ↗ RANK

KoboldCpp runs GGUF language models locally as a single-file application with a built-in web UI and API.

> KoboldCpp v1.125.1 adds native speech generation and transcription (TTS via `--wavtokenizer`, Qwen3ASR/SenseVoice transcription, word-level timestamp alignment), new integrations (BeaverNet worker hosting, MCP web search, `sdcpp` API compatibility), and new model/generation support, while tightening API-key enforcement on generation endpoints and dropping audio from LTX video generation.

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

01 Speech generation and transcription support

Not checked yet NEW

90

KoboldCpp can load upstream llama.cpp TTS models such as Kokoro via the `--wavtokenizer` slot for speech generation, transcribe audio with Qwen3ASR and SenseVoice models via the new `--transcribemodel` flag (renamed from `--whispermodel`), generate word-level timestamps by combining a transcriber model with new MMS or Qwen3 forced aligner models, and automatically transcribe `reference_text` reference audio for TTS voice clone models when a transcriber model is loaded.

Load a Kokoro TTS model alongside a text model to enable speech generation, placing the wavtokenizer mmproj in the dedicated slot.

```
$ koboldcpp --model model.gguf --wavtokenizer kokoro-mmproj.gguf
```

Load a transcription model for speech-to-text so that TTS voice clone reference audio is automatically transcribed.

```
$ koboldcpp --model model.gguf --transcribemodel qwen3asr.gguf
```

— Multiple named flags with runnable examples, but no limits or benchmarks given.

v1.125.1

02 New model support and generation features

Not checked yet NEW

85

Adds support for Inkling 2 Mini and Xing 4.0 29BA4B text generation models (using `--no-op-offload` on CUDA to mitigate known output corruption with large prompts on

Inkling 2), MiniMax H3 Image generation via an appropriate `.kcppt` template, interleaved reference image prompts on compatible models using `--sdclip2` for the mmproj with references cited inline as `<image:1>`, and `F8_E4M3` as a new KV cache quantization type.

Cap the CUDA partial-offload corruption risk on Inkling 2 Mini with large prompts by disabling op offload.

```
$ koboldcpp --model Inkling-2-Mini-IQ4_NL.gguf --no-op-offload
```

— Names specific models, flags, and includes a runnable mitigation command. v1.125.1

---

03 **Password now secures all generation endpoints** Not checked yet

60

BREAKING

When `--password` is set, all generation endpoints are now secured by the API key, including noscript mode — workflows that relied on unauthenticated access to any endpoint while `--password` was set will break.

— Clear before/after behaviour and named flag, no migration steps given. v1.125.1

THINNER COVERAGE BELOW

04 **BeaverNet, MCP search, gzip, and sdcpp API support**

Not checked yet NEW

55

KoboldCpp can host a BeaverNet volunteer crowdsourced LLM worker from within the app (similar to AI Horde), expose its web search capability as an MCP tool via the existing MCP server bridge, accept gzip-compressed HTTP requests, and expose the sd.cpp native `sdcpp` API as a compatibility layer for tools already written against that API.

— Named integrations and APIs but no setup commands given. v1.125.1

---

05 **Single flag for request timeouts** Not checked yet IMPROVED

55

All request timeout values are now read from a single `--reqtimeout` flag.

— Exact flag named but no default value or prior behaviour described. v1.125.1

---

06 **Agent delete tool and image attachment command** Not checked yet

NEW

55

KoboldCpp's agent gains a dedicated delete tool that operates automatically within the working directory without invoking `shell`, and the `/attach` command to attach images to the next agent message from disk.

— Named command and tool behaviour but no example shown. v1.125.1

07 **Expanded save slot capacity** Not checked yet IMPROVED 50

Increases the number and size of server-side save slots set via `--savedatafile`, and Kobold Lite now supports up to 20 browser save slots while displaying the used context token budget.

— Named flag and concrete slot count, but no prior limit stated. v1.125.1

08 **Audio dropped from LTX video generation** Not checked yet BREAKING 25

Audio support has been dropped from LTX video generation.

— Bare statement with no replacement or rationale given. v1.125.1

└──▷ **BREAKING ON UPGRADE**

! The `--whispermodel` flag is renamed to `--transcribemodel`; existing launch scripts using `--whispermodel` will need to be updated.

! When `--password` is set, all generation endpoints are now secured by the API key, including noscript mode — workflows that relied on unauthenticated access to any endpoint while `--password` was set will break.

! Audio support has been dropped from LTX video generation.

WAS THIS USEFUL?  

◆ ML Platforms & SDKs

Tracer Cloud OpenSRE

Feedback ⓘ v0.1.2026.11.23 NOTES ↗

RANK

OpenSRE is an open-source toolkit for building AI SRE agents.

> OpenSRE's only change this window was per-server MCP management commands added to the CLI.

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

01 **Per-server MCP management commands** Not checked yet NEW

40

The CLI can now verify, list, and remove individual MCP servers one at a time, rather than managing them only in bulk.

— Names CLI capability but no exact command syntax given. v0.1.2026.11.23

WAS THIS USEFUL? 👍 👎

siliconscope

Feedback ⓘ v4.9.2 NOTES ↗ RANK

Siliconscope sudoless Apple Silicon system monitor (native SwiftUI GUI) with ANE / Media Engine / memory-bandwidth tracking

> SiliconScope v4.9.2 adds memory bandwidth tracking for M5 Ultra and exposes per-die CPU temperature sensors on M3 Ultra systems.

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

01 **M3 Ultra dual-die CPU temperature sensors** Not checked yet NEW

65

The Sensors card now lists both dies' CPU sensors on M3 Ultra, named Die 1 CPU 1–8 and Die 2 CPU 1–8, so loads on the second die are reflected in the overall CPU temperature reading.

— Names exact sensor labels and UI location, no command v4.9.2

THINNER COVERAGE BELOW

02 **M5 Ultra memory bandwidth support** Not checked yet NEW

55

SiliconScope adds support for reading memory-controller throughput on the M5 Ultra chip. A chip report can be captured via `siliconscope --chip-report` to verify correct bandwidth readings.

Capture an M5 Ultra bandwidth report to verify correct memory-controller throughput readings.

```
$ siliconscope --chip-report
```

— Names chip and command but no mechanism detail v4.9.2

WAS THIS USEFUL? 👍 👎

◆ Inference Serving & Optimization

Niko1221 Strata

Feedback ⓘ v0.2.20 NOTES ↗ RANK

Strata is a local inference engine with one-click install for Windows and Linux, providing an OpenAI/Anthropic-compatible API on localhost for models like Qwen and DeepSeek.

> Strata v0.2.20 introduces 4-tier GPU+RAM+CPU+SSD Mixture-of-Experts execution down to 4 GiB GPUs, multi-GPU layer splitting, direct lossless GGUF quant support for Inkling 2 Mini, and broad hardware acceleration gains across NVIDIA CUDA, Apple Metal, AMD HIP, and Intel Arc SYCL backends.

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

01 4-tier GPU+RAM+CPU+SSD MoE execution

Not checked yet NEW

95

Strata can now run Mixture-of-Experts models across GPU, RAM, CPU, and SSD tiers for Qwen3, HY-4, and Inkling 2 Mini, down to 4 GiB GPUs. Use `--kv-offload auto` for automatic host KV offloading on 32K+ context, `--ssd-experts --ram-experts-gb 0` for GPU+SSD-only mode, or CPU+SSD-only mode on the smallest GPUs. HY-4 Flash hybrid RAM+VRAM throughput now reaches 128.25 tok/s on RTX 5090 + 128 GB DDR5, up from 30 tok/s (4.27×).

— Names exact flags and a concrete before/after throughput number v0.2.20

02 NVIDIA CUDA MoE decode acceleration

Not checked yet IMPROVED

75

Strata now accelerates MoE decode on NVIDIA CUDA (`sm_60` – `sm_121` , including Tesla V100 `sm_70`) with all 13 Unsloth expert pairs fused in `EXP_MODE=8` , delivering 1.04×–1.67× faster MoE decode and up to 2.58× faster down projection.

— Specific architectures and speedup figures but no user-facing flag v0.2.20

03 Multi-GPU MoE layer splitting

Not checked yet NEW

70

Strata can scale MoE inference across 2–4 GPUs on Qwen3.8-Flash-Next, HY-4 Flash, and Inkling 2 Mini using `--layer-split auto` .

— Clear flag given but no mechanism or benchmark detail v0.2.20

04 **Inkling 2 Mini lossless GGUF quant support** Not checked yet NEW

70

Inkling 2 Mini can now run directly from lossless Unsloth GGUF quants via `GGUF_RAW` execution, including `UD-Q4_K_XL`, `Q8_0`, `UD-IQ4_XS`, `UD-IQ3_XXS`, `UD-Q6_K_XL`, `UD-Q5_K_XL`, `UD-Q3_K_XL`, and `UD-Q2_K_XL`.

— Names execution mode and all eight quant formats v0.2.20

05 **Intel Arc SYCL acceleration** Not checked yet IMPROVED 70

Strata now accelerates Intel Arc SYCL (Alchemist and Battlemage) with hoisted multi-token `Split<TY>` / `KqSplit<TY>` and SG16 XMX SLM decoders, reaching 1.37×–2.73× faster across `IQ3_XXS`, `Q8_0`, `IQ4_XS`, `Q4_K`, and `NVFP4`, plus 4.11× faster Intel Arc Pro B70 SYCL prefill.

— Detailed mechanism, formats, and quantified speedups v0.2.20

06 **Apple Silicon Metal acceleration** Not checked yet IMPROVED

65

Strata now accelerates Apple Silicon Metal with native `IQ3_XXS`, `Q2_K`, `Q3_K`, `Q8_0`, `NVFP4`, and halved-load `W8<IQ4_XS/IQ4_NL>` formats across decode and ragged MoE prefill.

— Names formats but gives no setup steps or benchmark v0.2.20

07 **AMD HIP GPU acceleration** Not checked yet IMPROVED 65

Strata now accelerates AMD HIP (RDNA2/RDNA3/RDNA3.5 Strix Halo/RDNA4) with unified Wave32 WMMA prefill and 6-op `v_perm_b32` LUT16 Wave32 DP4A decode.

— Names mechanism and GPU generations but no benchmark v0.2.20

WAS THIS USEFUL? 👍 👎

LMCache

Feedback ⓘ v0.5.8 NOTES ↗ RANK

LMCache reuses and shares language-model key-value caches across requests and serving instances to reduce repeated prefill work.

> LMCache's v0.5.8 release adds peer-to-peer RDMA transfer with a shadow-L1 index over borrowed peer memory, new L2 storage backends for Valkey, HF3FS and Device-DAX, VMM KV-cache support for ROCm and vLLM, and coordinator API-key authentication, while retiring the bundled NIXL package and CUDA 12.9 support.

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

01 New L2 storage backends: Valkey RDMA, HF3FS, Device-DAX

Not checked yet NEW 73

Adds a Valkey RDMA L2 adapter and blob-TCP transport for Valkey-backed L2 storage, HF3FS `shard_dirs` to distribute stores across hash-based subdirectories, and mapping of shared Device-DAX memory regions.

— Names backends and a config key but lacks usage example v0.5.8

02 VMM KV-cache support across ROCm and vLLM Not checked yet NEW

73

Enables VMM KV-cache IPC on ROCm via `rocm.bindings` with an opt-in host-side event wait, exposes vLLM multiprocess `kv_cache_access` with auto-detected VMM KV allocation, and adds VMM region sharing via file descriptors plus sleep mode with VMM-shared vLLM KV caches.

— Rich mechanism detail but no exact flag to enable the option v0.5.8

03 Coordinator security and control API Not checked yet NEW

65

Protects gRPC and ZMQ endpoints with API-key authentication, binds the coordinator to loopback only by default, and adds a coordinator status query plus chunk-level pin/unpin controls.

— Names auth and controls but no exact endpoint or flag given v0.5.8

04 **Object storage allocation controls** Not checked yet IMPROVED 65

Shared-object grouping is now opt-in by default, overridable with `--separate-object-groups`, and LMCache now warns when lazy L1 allocation disables the requested `--shm-name`.

— Names exact flags with clear behavioural change v0.5.8

05 **NIXL packaged separately from core library** Not checked yet 60

BREAKING

NIXL is no longer bundled with the main `lmcache` package; install it separately as `lmcache-nixl`.

— Clear migration step but no rationale given v0.5.8

06 **Peer-to-peer RDMA transfer and shadow-L1 memory sharing**

Not checked yet NEW 60

Supports peer-to-peer RDMA writes, peer store RPCs, chunk-level transfer engines, and RDMA worker placement across fabric interfaces, and introduces a shadow-L1 index over borrowed peer memory with a tensorless `BorrowedMemoryObj`, exposing tagged object addressing through cache controls.

— Names mechanisms but no config surface to invoke them Names a class but gives no usage path

v0.5.8

THINNER COVERAGE BELOW

07 **Token-dropping transfer improvements** Not checked yet IMPROVED 45

Introduces resident-length-aware allocation and safer prefix publication for token-dropping requests, and moves token-dropping transfers to an LMCache-driven SDK path.

— Describes change but no concrete API or flag named v0.5.8

08 **CUDA 12.9 support retired** Not checked yet BREAKING 40

CUDA 12.9 images and wheels are retired.

— States retirement without migration guidance v0.5.8

09 **SGLang connector health-check interface** Not checked yet NEW

35

Adds a health-check interface to the SGLang connector.

— Bare mention with no mechanism or usage detail v0.5.8

10 **Native MUSA CacheGen codec** Not checked yet NEW

20

Adds a native MUSA CacheGen codec.

— Only named in summary, no further detail given v0.5.8

└──▷ **BREAKING ON UPGRADE**

! NIXL is no longer bundled with the main `lmcache` package; install it separately as `lmcache-nixl`.

! CUDA 12.9 images and wheels are retired.

WAS THIS USEFUL? 👍 👎

◆ AI / LLM Gateways

cc-router

Feedback ⓘ v7.1.0 NOTES ↗ RANK

Cc-router is a 本地运行的大模型聚合网关，GUI桌面端app，零代码部署，把Coding Plan、大模型 API 额度聚合成一个虚拟 Plan，一键接入 Claude Code、Claude Desktop App、OpenClaw、OpenCode 等工具。Bundle your scattered Token Plan, Coding Plan, and LLM API quotas into a single virtual Plan, and plug it into Claude Code, Claude Desktop App, OpenClaw, OpenCode, and more.

> cc-router's v7.1.0 release introduces Sparrow Bakery, a plugin hub spanning six plugin types for extending the gateway, alongside desktop notifications, a new built-in theme, and one-click subscription reconnection.

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

01 **Sparrow Bakery plugin hub with six plugin types** Not checked yet

NEW 95

A new sidebar entry lets users browse, search, and filter a community plugin catalog, installing plugins via a GitHub repository URL pinned to a specific commit or by loading from a local folder. Provider plugins extend `cc-router` beyond built-in upstreams with pre-install domain and diff previews, declarative `JSON` balance queries, and existing subscriptions continuing on saved snapshots after uninstall. Pricing plugins compute estimated costs in real time from the current price table, surfacing a Costs card in Statistics and per-request estimates in Request logs details. Rules plugins (disabled by default after install) reroute requests to another virtual model or reject them before routing, and can append to the system prompt, clamp sampling/thinking parameters, remove tool definitions, or redact text, with a dry-run preview against recent request logs (`/v1/systemone` exempt). Notifier plugins (disabled by default after install) push selected events to `ntfy` , Bark , Feishu , `DingTalk` , Telegram , and Slack without sending API keys, prompts, or headers. UI theme plugins offer a pre-install Try on preview, and receipt theme plugins support `PNG` , `PDF` , and `HTML` export.

Route push alerts for subscription quota exhaustion to a Telegram channel without exposing API keys or prompts.

> 📍 In the app, open Plugins (Sparrow Bakery), install a notifier plugin for Telegram, enable the plugin, select the Subscription quota reached event, enter your bot token (stored masked), and click Save & send test .

Block requests that exceed a token budget before they are sent upstream, using a rules plugin dry-run to verify against recent logs before going live.

> 📍 In the app, open Plugins (Sparrow Bakery), install a rules plugin, go to its configuration form, set the before-send parameter to clamp max_tokens, click Dry

run to preview against recent request logs, then enable the plugin.

— Names every plugin type, endpoint, and service with install mechanics v7.1.0

02 **Desktop notifications for subscription and proxy events**

Not checked yet NEW 80

Enable system desktop notifications under Settings → General → Notifications, with configurable event selection, a Daily summary time (default 21:00), 10-minute deduplication, and click-to-navigate to the matching page.

Turn on desktop alerts for subscription auth failures and proxy stops so your team is notified the moment an upstream goes down.

📍 In the app, go to Settings → General → Notifications, enable System desktop notifications, confirm that Subscription auth failed and Proxy stopped unexpectedly are checked, and click Send test notification to verify delivery.

— Exact settings path, default value and dedup window given v7.1.0

THINNER COVERAGE BELOW

03 **New Sparrow built-in theme** Not checked yet NEW 45

A new built-in Sparrow theme — warm-cream background with ginger-orange accent — can be selected from Settings → General → Appearance or the sidebar.

— Names theme and exact UI path but no further mechanism v7.1.0

04 **One-click subscription connection rebuild** Not checked yet

IMPROVED 45

When a provider description has updated, a subscription's connection can be rebuilt with one click using a field-by-field diff banner shown on the subscription edit page.

— Describes trigger and UI location but little mechanism detail v7.1.0

05 **Receipt footer code default changed to barcode** Not checked yet

45

BREAKING

The receipt Footer code default changes from QR code to Barcode; existing setups that relied on the QR code default will now produce barcodes unless explicitly reconfigured.

— Names exact setting and before/after but no migration steps v7.1.0

└──▷ BREAKING ON UPGRADE

! Receipt Footer code default changes from QR code to Barcode; existing setups that relied on the QR code default will now produce barcodes unless explicitly reconfigured.

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

matrixone

Feedback ⓘ v4.2.9 NOTES ↗ RANK

Matrixone is an ai-native HTAP database with Git-for-Data and built-in vector search, serving as the data and memory backbone for intelligent agents and applications.

> MatrixOne v4.2.9 improves backup efficiency and query memory handling, while introducing a breaking change to ENUM storage that requires a full-cluster shutdown before upgrading.

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

01 **ENUM format change requires full-cluster shutdown** Not checked yet

60

BREAKING

Once new-format ENUM definitions are written by v4.2.9, that data must not be reopened with v4.2.8 or its hotfixes; a full-cluster shutdown is required before upgrading to avoid incompatibility.

– Clear migration requirement with exact version compatibility noted v4.2.9

THINNER COVERAGE BELOW

02 **Byte-bounded spill runs for wide-row queries** Not checked yet

IMPROVED 40

Spill runs for disk-spilling queries are now bounded by bytes rather than row count, improving memory behavior for wide rows on compute nodes.

– Mechanism named but no tunable parameter or limit given v4.2.9

03 **Immutable object reuse in incremental backups** Not checked yet

IMPROVED 38

Incremental backups now reuse immutable objects from earlier backups instead of re-transferring them, reducing redundant data transfer and storage usage.

– Describes benefit but no config key or command to invoke it v4.2.9

└──▷ BREAKING ON UPGRADE

! Once new-format ENUM definitions are written by v4.2.9, that data must not be reopened with v4.2.8 or its hotfixes. A full-cluster shutdown is required before upgrading.

WAS THIS USEFUL? 👍 👎

Milvus

Feedback ⓘ client/v3.0.1 NOTES ↗ RANK

Milvus is a high-performance, cloud-native vector database built for scalable vector ANN search

> Milvus's Go SDK v3.0.1 adds Row-Level Security policy management and enforcement across read, write, and bulk-import paths, partial field replacement on upserts, lightweight snapshot summaries, nullable/nested array schema support, and new Timestamptz/Geometry field types — alongside breaking changes to template parameter handling, AutoID upserts, and the Go toolchain requirement.

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

01 Row-Level Security policy management and enforcement

Not checked yet NEW 90

Adds RLS management APIs `CreateRowPolicy`, `UpdateRowPolicy`, `DropRowPolicy`, and `ListRowPolicies` (configured via `NewCreateRowPolicyOption` / `NewUpdateRowPolicyOption` with `WithUsingExpr`, `WithCheckExpr`, and `WithDescription`), plus `SetRLSPrincipalTags`, `GetRLSPrincipalTags`, `DeleteRLSPrincipalTags`, and `ListRLSPrincipals` (with `WithCountOnly` for count-only responses) for tagging and listing RLS principals. RLS context can be applied to `Search`, `Query`, iterators, `Insert`, `Upsert`, `Delete`, and the REST bulk importer via `WithRLSPrincipal` or `WithSkipRLS`, and the same options are now available on `BulkImportOption` for bulk import jobs.

Create a row policy that restricts read access to rows owned by the current user, then verify it was registered.

go

```
// Create a policy allowing reads only where owner matches the
principal
opt := milvusclient.NewCreateRowPolicyOption("my_collection",
"owner_read_policy").
    WithUsingExpr("owner == @principal").
    WithDescription("Restrict reads to row owner")
_ = client.CreateRowPolicy(ctx, opt)

// List all policies on the collection
policies, _ := client.ListRowPolicies(ctx,
milvusclient.NewListRowPoliciesOption("my_collection"))
fmt.Println(policies)
```

— Full API surface named across operations with a runnable example client/v3.0.1

02 **Nullable and nested array schema support** Not checked yet NEW

85

Adds null elements inside `Array` and vector-array columns via `NewElementNullableColumnXxxArray` constructors with `ElementNullableColumn` accessors (`ElementNullable` , `ElementLen` , `ElementValidData` , `ElementIsNull`) and `WithRowValidData` / `WithElementSparseMode` options, plus nested `Array<Array<T>>` struct sub-fields via `ColumnXxxArrayArray` columns (e.g. `ColumnInt64ArrayArray`) and `NewElementNullableColumnXxxArrayArray` constructors with `NestedArrayColumn` accessors (`LeafType` , `LeafNullable` , `LeafValidData` , `LeafIsNull`). A sub-field can be added to an existing struct at runtime by passing a `struct[sub_field]` path to `AddCollectionField` (via `NewAddCollectionFieldOption`), which calls `AlterCollectionSchema` , with leaf nullability set via `entity.Field.WithElementNullable` .

— Every constructor and accessor named but no usage example client/v3.0.1

03 **Partial field replacement via WithPathReplace** Not checked yet

NEW

80

Column-based and row-based upserts support partial field replacement using WithPathReplace(fieldName, path), which can target a scalar Array element, a complete StructArray element, a struct child, or a JSON field value — avoiding rewriting the whole field.

Partially update a single element inside a JSON field without overwriting the whole document, using path replacement on a column-based upsert.

go

```go
// Replace just the 'status' key inside the 'metadata' JSON field
upsertOpt :=
milvusclient.NewColumnBasedUpsertOption("my_collection").
    WithPathReplace("metadata", "status")
_, _ = client.Upsert(ctx, upsertOpt, idCol, statusCol)
```

— Exact method signature and runnable example given client/v3.0.1

04 **Lightweight snapshot summaries via ListSnapshotSummaries**

Not checked yet NEW

80

`Client.ListSnapshotSummaries` retrieves lightweight server-side snapshot summaries ordered by create time, avoiding per-name `DescribeSnapshot` round-trips. The underlying `ListSnapshots` protobuf now carries optional

`snapshotSummaries`; servers that return only names are reported as unsupported via `ErrServiceUnimplemented`.

Fetch snapshot summaries in server creation order without issuing a separate DescribeSnapshot call per snapshot.

```go
summaries, err := client.ListSnapshotSummaries(ctx,
milvusclient.NewListSnapshotSummariesOption("my_collection"))
if err != nil {
    log.Fatal(err)
}
for _, s := range summaries {
    fmt.Println(s)
}
```

— Named method, protobuf field, and error type with example `client/v3.0.1`

05 **Timestamptz and Geometry field types** Not checked yet NEW

78

Row-based inserts now accept `FieldTypeTimestamptz` and `FieldTypeGeometry` values, mapped to `NewColumnTimestamptz` (accepting `time.Time`, `*time.Time`, or RFC3339Nano ISO strings) and `NewColumnGeometryWKT`. Query results parse RFC3339Nano ISO timestamp strings back into `time.Time` or `*time.Time` struct fields via updated `row.CoerceValue` and `row.SetField`.

— Constructors and types named but no usage example `client/v3.0.1`

06 **WithTemplateParam no longer copies slice values** Not checked yet

75

BREAKING

Slice-valued `WithTemplateParam` values (`[]int64`, `[]string`, `[]float64`, `[]bool`) are no longer copied — mutating the slice while a Search/Query call is in flight, or after Request() returns, is now undefined behaviour; callers must pass a copy if they intend to keep writing to the slice.

— Exact flag and failure mode named, mitigation stated `client/v3.0.1`

THINNER COVERAGE BELOW

07 **Full AutoID Upsert lookup semantics changed** Not checked yet

55

BREAKING

Full AutoID Upsert now always retains the AutoID primary key as its lookup key and requires Query availability and a loaded target collection.

— Behaviour change described but no migration example given client/v3.0.1

08 **Row reflection helpers moved to rowutil package** Not checked yet

IMPROVED

45

Row-based reflection helpers (`row.AnyToColumns` , `row.SetField` , struct tag constants) are now consolidated in the new `client/internal/rowutil` package.

— Internal refactor named but no behavioural detail client/v3.0.1

09 **Go toolchain requirement bumped to 1.25.0** Not checked yet

45

BREAKING

The module's `go` directive is bumped to Go 1.25.0 (from 1.24.9); projects with an older `go` directive in their `go.mod` will need to update.

— States exact version change, no further migration detail client/v3.0.1

└──▷ **BREAKING ON UPGRADE**

! The module's `go` directive is bumped to Go 1.25.0 (from 1.24.9); projects with an older `go` directive in their `go.mod` will need to update.

! Slice-valued `WithTemplateParam` values (`[]int64` , `[]string` , `[]float64` , `[]bool`) are no longer copied — mutating the slice while a Search/Query call is in flight, or after Request() returns, is now undefined behaviour; callers must pass a copy if they intend to keep writing to the slice.

! Full AutoID Upsert now always retains the AutoID primary key as its lookup key and requires Query availability and a loaded target collection.

! `ListSnapshots` protobuf messages now carry optional `snapshotSummaries` ; servers that return only names are reported as unsupported by `ListSnapshotSummaries` .

WAS THIS USEFUL?  

dosco GraphJin

Feedback ⓘ v3.21.24 NOTES ↗ RANK

GraphJin is a governed graph for AI agents — GraphQL + MCP over your databases, files, APIs, and code.

> GraphJin now supports updating multiple rows by primary key in a single mutation, cutting down on per-record round trips.

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

01 **Bulk row update by primary key** Not checked yet NEW 45

GraphJin can now update a list of rows in one mutation, matched by primary key, without issuing separate queries per record.

— Describes behaviour but no mutation syntax or example given v3.21.24

WAS THIS USEFUL? 👍 👎

ai-microcore

Feedback ⓘ v6.10.0 NOTES ↗ RANK

Ai-microcore is a handy lib for smooth interaction with large language models (LLMs) and crafting AI apps.

> ai-microcore v6.10.0 adds provider-agnostic `tool_calls` and `finish_reason` fields on LLM responses across OpenAI, Anthropic, and Gemini, and changes how existing tool-call helpers behave when a native tool call is present.

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

01 Provider-agnostic tool_calls and finish_reason on LLMResponse

Not checked yet NEW 90

ai-microcore now exposes `response.tool_calls` (a list of `ToolCall` objects with `name`, `arguments`, `id`, and `args`) and `response.finish_reason` (one of `stop`, `length`, `tool_calls`, `content_filter`, or `None`) uniformly on llm() and allm() responses across OpenAI, Anthropic, and Gemini backends, whether streamed or not. Tool calls from streamed responses are now assembled from the stream and returned on the response object instead of being lost.

Call a tool-enabled model and inspect which tool was invoked and why the model stopped generating.

python

```python
from microcore import llm

response = llm("What is the weather in Kyiv?", tools=tools)
print(response.tool_calls)          #
[ToolCall(name='get_weather', arguments='{"city": "Kyiv"}',
id='call_1')]
print(response.tool_calls[0].args)  # {'city': 'Kyiv'}
print(response.finish_reason)       # 'tool_calls'
```

— Names exact fields, value sets, backends, and includes runnable example v6.10.0

02 Native tool-call handling in is_tool_call/as_tool_call/mcp.exec

Not checked yet 80

BREAKING

is_tool_call(), as_tool_call(), and mcp.exec() now read native tool calls in addition to tool calls written in the response text, so the same code handles both `@ai_func`/MCP tool calls and native tool-call responses. Previously, for a response carrying a native tool call, is_tool_call() returned `False`, as_tool_call() returned `None`, and

mcp.exec(response) raised `WrongMcpUsage`; they now report and run the call. If a response has both a native tool call and one written in its text, the native call now takes precedence, whereas text-embedded calls could previously win. A tool-call-only response is now an empty string instead of the string `'None'` previously returned by OpenAI (sync) and Gemini, and with hidden output configured, stream callbacks no longer receive an empty string after the end marker.

— Names functions and exact before/after behavior, no runnable migration command v6.10.0

└──▷ **BREAKING ON UPGRADE**

! For a response with native tool calls, is_tool_call() previously returned `False`, as_tool_call() returned `None`, and mcp.exec(response) raised `WrongMcpUsage`; they now report and run the call.

! If a response has both a native tool call and one written in its text, the native one is now used (previously text-embedded calls could take precedence).

! A response that only calls tools is now an empty string; OpenAI (sync) and Gemini previously returned the string `'None'`.

! With hidden output configured, stream callbacks no longer receive an empty string after the end marker.

WAS THIS USEFUL? 👍 👎

FalkorDB

Feedback ⓘ v6.2.0 NOTES ↗

FalkorDB is a super fast Graph Database uses GraphBLAS under the hood for its sparse adjacency matrix graph representation. Our goal is to provide the best Knowledge Graph for LLM (GraphRAG).

FalkorDB v6.2.0 adds schema-hash signalling in query replies, dual AGPLv3/SSPLv1 licensing, an 11x faster SAVE, and server-wide responsiveness improvements.

└──▷ GET THIS VERSION ▾

Docker

```
$ docker pull ghcr.io/falkordb/falkordb:6.2.0
```

git

```
$ git clone --branch v6.2.0 https://github.com/FalkorDB/FalkorDB.git
# already have the repo? check out this version:
$ git checkout v6.2.0
```

published under the project's GitHub account · checked 23 Nov Wrong command?

Query graphs with schema-change awareness by passing `SCHEMA_HASH <h>` to `GRAPH.QUERY` or `GRAPH.RO_QUERY` with `--compact`; the reply returns the current schema hash and full label, relationship-type, and property names when the hash differs, so clients can detect a graph dropped and recreated under the same key. Not checked yet

NEW SUPPORTED PLATFORM Use FalkorDB under AGPLv3 as an alternative to SSPLv1; both license texts (`LICENSE-SSPL`, `LICENSE-AGPL`) ship in every Docker image and release asset. Not checked yet

`SAVE` and `BGSAVE` are now 11x faster on large graphs — a 3.07M-node, 117M-edge graph saves in 8.2 s instead of 90.5 s, and peak extra memory drops from 4.5 GB to 1.8 GB. Not checked yet

`MATCH` after `UNWIND` or `WITH` now finds its start node per row instead of walking every edge, reducing a 382,000-edge delete from 747 s to 0.37 s. Not checked yet

Write queries and constraint creation that wait for a long-running read no longer freeze the server; `PING` and other commands are answered while they wait. Not checked yet

+1 more

NEW SUPPORTED PLATFORM Valkey 9.2.0 support

└──▷ **BREAKING ON UPGRADE**

! The `latest` and bare version tags (e.g. `6.2.0`) now run Valkey 9.2.0 instead of Redis. Valkey cannot read save files written by Redis 7.4 or later; upgrading a Redis-based volume without migration will cause the container to stop on startup. Switch to `-redis` tags (`6.2.0-redis`, `latest-redis`) to stay on Redis, or follow the migration steps using `FALKORDB_REDIS_MIGRATION=1`.

WAS THIS USEFUL? 👍 👎

infino

Feedback ⓘ v0.11.36 NOTES ↗ RANK

Infino is a search library built on Parquet. Fast, efficient, and scalable.

> Infino shipped multi-column BM25 ranking and a query optimization that speeds up null-check counts.

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

01 **Multi-column BM25 search with `bm25_search_fields`**

Not checked yet NEW 65

A new `bm25_search_fields` parameter lets a BM25 search call rank a single query across several text columns at once, rather than requiring separate searches per column.

– Names the exact parameter but no example usage shown. v0.11.36

THINNER COVERAGE BELOW

02 **Range index used for COUNT IS NOT NULL queries** Not checked yet

IMPROVED 55

SQL COUNT col IS NOT NULL queries now read from the range index instead of scanning rows, speeding up these aggregate queries.

– Explains mechanism change but gives no benchmark numbers. v0.11.36

WAS THIS USEFUL? 👍 👎

aegis-stack

Feedback ⓘ v0.21.0 NOTES ↗ RANK

Aegis-stack is a production-ready FastAPI platform with modular components and a built-in control plane.

> v0.21.0 adds on-device TTS/STT via Kokoro and NVIDIA Parakeet, a new runner component for coding-agent sessions, approval-gated change and memory workflows, and standalone MCP token management (with a breaking change for pre-existing tokens), alongside agent inheritance, code-mode resource limits, database activity retention controls, a live call stage, and AI cost breakdowns.

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

01 On-device TTS/STT via Kokoro and Parakeet

Not checked yet NEW

95

Adds on-CPU text-to-speech via Kokoro (`TTS_PROVIDER=kokoro`) and speech-to-text via NVIDIA Parakeet (`STT_PROVIDER=parakeet`) with no API key required, installed through `ai[voice]` (`uv add kokoro-onnx` and `uv add 'onnx-asr[cpu,hub]'` `av`). A Kokoro-FastAPI speech server can be tracked via `inference[kokoro_fastapi]` or run as a Compose service via `inference[docker,kokoro_fastapi]` , configured at `SPEECH_BASE_URL` and managed with `my-app inference speech` , `speak` , `speech-model` , or `GET` `/api/v1/inference/speech` .

— Names env vars, install commands, CLI subcommands and endpoint. v0.21.0

02 Runner component for coding-agent sessions

Not checked yet NEW

95

Adds runner activity posting via `POST /api/v1/runner/runs/activity` signed with the runner's token, `runner hooks` to print Claude Code hook config for a session, and runs.ask(..., branch=, session_id=runs.new_session()) to target a run on a specific branch and session, with `--timeout` (default 15 minutes), `--resume-command` , and overrides via `RUNNER_COMMAND` and `RUNNER_RESUME_COMMAND` . Runs can also be titled and sourced for display via runs.ask(title=..., source=...).

— Endpoint, CLI subcommand, method signature, env vars and defaults all named. v0.21.0

03 Approval-gated change and memory workflows

Not checked yet NEW

90

Pending approval cards can be revised via `change_queue.revise` , `POST` `/changes/{id}/revise` , or the agent's `revise` tool, with editability controlled by `ChangeExecutor.editable` and narrowed by `edits` and `choices` ; scanned pictures are shown via `scan` (`/partials/changes/{id}/scan`), follow-up logic runs after approval via `after_commit` , and cards are held for items about to be

removed via `expire_naming`. Memory changes are queued as approval cards via `save_memory`, `replace_memory`, `update_memory`, and `forget_memory`, with nothing committed until the user approves.

— Enumerates every named method, endpoint and field for the workflow. v0.21.0

04 **MCP token management without the auth service** Not checked yet

80

BREAKING

Create, list, and revoke MCP tokens without the auth service via the `mcp-tokens` CLI or the token API; token calls act on behalf of `MCP_OWNER_USER_ID`. Once the auth service is added to the stack, any MCP token created before that point is refused.

— CLI and env var named, plus the exact breaking condition. v0.21.0

05 **Local template path for aegis init** Not checked yet NEW

70

`AEGIS_TEMPLATE_PATH` points `aegis init` at a local checkout instead of cloning from GitHub, letting developers render templates offline or test local changes.

Point `aegis init` at a local repo clone to render templates offline or test local template changes.

```
$ AEGIS_TEMPLATE_PATH=/path/to/aegis-stack uvx aegis-stack init my-app
```

— Single config key with a runnable example command. v0.21.0

06 **Resource limits for code-mode execution** Not checked yet NEW

70

`AI_CODE_MODE_MAX_SECONDS` (default 10 s) and `AI_CODE_MODE_MAX_MEMORY_BYTES` (default 64 MiB) cap how long and how much memory a code-mode script execution can consume.

— Two named config keys with defaults but no file path. v0.21.0

07 **Retention control for slow-query and lock-wait records**

Not checked yet NEW

70

`DATABASE_ACTIVITY_RETENTION_SECONDS` (default one week) controls how long slow-query and lock-wait records are kept; records are viewable in Overseer > Database > Slow & locked.

— Config key with default plus exact UI navigation path. v0.21.0

08 **Per-model and per-call AI cost breakdowns** Not checked yet NEW

65

View per-model and per-call AI cost breakdowns in Overseer AI > Costs, or print them with `ai costs`.

— UI path plus a runnable CLI command. v0.21.0

THINNER COVERAGE BELOW

09 **Single-level agent inheritance via extends** Not checked yet NEW

45

The `extends` field names an agent's parent, enabling one-level agent inheritance.

— Names the field but no mechanism beyond one sentence. v0.21.0

10 **Full-screen live call stage in chat UI** Not checked yet NEW

45

A full-screen live call stage is available at `<chat mount>/stage` alongside the chat interface.

— UI path given but no further mechanism described. v0.21.0

└──▷ **ALSO FROM THESE RELEASES**

└──▷ **BREAKING ON UPGRADE**

! An MCP token created before the auth service was added is refused once auth is added to the stack.

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

LangChain LangSmith

Feedback ⓘ updated · Nov 23, 2026

NOTES ↗ RANK

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

> LangSmith's latest release focuses on LLM Gateway observability and Bedrock authentication, alongside thread-level annotation automation, sandbox diagnostics, and several smaller dataset and permissions fixes.

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

01 **Gateway metadata surfaced in traces** Not checked yet NEW

85

LLM Gateway responses now include an `X-LangSmith-Gateway-Metadata` response header describing gateway routing, limits, guard activity, and upstream errors, which clients can attach to traces as `ls_gateway_info` metadata. Trace details now show a dedicated Gateway section with LLM Gateway outcomes, policy limits, selected models, and fallback attempts, and runs blocked by the gateway display a Blocked badge in the trace tree.

— Names exact header, metadata key, and UI elements snapshot-20261123

02 **IAM role authentication and model routing for Bedrock**

Not checked yet IMPROVED

75

The LLM Gateway now authenticates to Amazon Bedrock with an assumed IAM role or AWS access keys instead of requiring a static Bedrock API key, with existing Bedrock API keys still taking precedence when multiple credential types are configured. Requests are routed to the API format supported by each model family: Anthropic models use Messages, OpenAI models use Responses, and other Bedrock models use Chat Completions.

— Explains mechanism and routing but no exact config steps snapshot-20261123

03 **CSV export for colon-containing dataset keys** Not checked yet

IMPROVED

65

Datasets whose input, output, or metadata keys contain a colon (including `source_run_feedback:<key>` metadata keys) can now be exported to CSV.

— Names exact key pattern and export format snapshot-20261123

04 **Thread-level automation with SmithDB queries** Not checked yet

60

BREAKING

Thread webhook and annotation-queue automations now require SmithDB queries and show thread filters instead of run filters; automations grouped by thread can add one annotation queue item per matched conversation so reviewers can inspect the whole thread instead of only one run. Automations previously using run filters will need to be updated to the new SmithDB-query-based thread filters.

— Describes behavior change and migration need, no exact query syntax snapshot-20261123

---

05 **Sandbox resource history and WebSocket fixes** Not checked yet

IMPROVED

60

Sandboxes now retain recent CPU, memory, and network usage history to help understand resource consumption and diagnose performance bottlenecks. WebSocket connections through a sandbox service URL now reach the service with their query string intact, allowing dev servers that authenticate WebSocket connections via a query parameter (such as Vite's HMR token) to connect.

— Names CPU/memory/network metrics and the Vite HMR query-string fix snapshot-20261123

---

06 **Turns view deprecated in thread detail** Not checked yet DEPRECATED

60

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; users should use the Messages or Details view instead.

— Gives exact removal dates, version, and replacement views snapshot-20261123

THINNER COVERAGE BELOW

---

07 **Tag-on-create permission for resources** Not checked yet NEW

45

Users with tag-on-create permission for datasets, projects, or prompts can now see and apply tags when creating those resources.

— Names the permission and affected resources, no further mechanism snapshot-20261123

---

08 **API Keys page full-list filtering** Not checked yet IMPROVED

45

The API Keys page now filters keys across the full paginated list by key, description, and workspace.

— Names the exact filter fields and UI location snapshot-20261123

---

09 **New GPT-5.6 models in LangSmith Chat** Not checked yet NEW

35

LangSmith Chat now offers GPT-5.6 Sol, GPT-5.6 Terra, and GPT-5.6 Luna alongside existing OpenAI models.

— Names models but no further detail snapshot-20261123

---

10 **IPv6 support for LangSmith services** Not checked yet IMPROVED

25

LangSmith services now accept both IPv4 and IPv6 connections.

— Bare statement with no further mechanism snapshot-20261123

└──▷ **BREAKING ON UPGRADE**

! 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.

! Thread annotation-queue and webhook automations now require SmithDB queries; automations using run filters will need to be updated.

WAS THIS USEFUL? 👍 👎

Superlog Labs Superlog Feedback ⓘ

changes since 2026-10-24 CODE ↗

Superlog monitors AI agent telemetry with OpenTelemetry and groups signals into incidents for observability.

Superlog adds org-level monthly spend limits, ClickHouse insert pacing, and MCP request forwarding for migrated projects

└──▷ **TRY IT**

Prevent slow-replica accumulation in a ClickHouse cluster behind a TCP load balancer by opening a fresh connection per insert.

```
$ INGEST_CLICKHOUSE_KEEP_ALIVE=false node dist/ingest-consumer.js
```

Slow insert pacing so busy tables build larger batches rather than creating excessive parts, reducing ClickHouse merge pressure.

```
$ INGEST_CLICKHOUSE_MIN_INSERT_INTERVAL_MS=500 node dist/ingest-consumer.js
```

Control ClickHouse insert pacing by setting `INGEST_CLICKHOUSE_MIN_INSERT_INTERVAL_MS` to enforce a minimum gap between inserts into the same table, growing batch sizes instead of producing more parts when inserts speed up. Not checked yet

Disable ClickHouse keep-alive connections by setting `INGEST_CLICKHOUSE_KEEP_ALIVE=false` to open one connection per insert, spreading load evenly across replicas instead of accumulating on the slowest one. Not checked yet

Orgs can now set a monthly spend limit that freezes all billable features once the estimated bill reaches the ceiling, without affecting usage within included allowances. Not checked yet

MCP requests for projects migrated to a workspace are now forwarded to `${workspace_url}/api/mcp` with the same JSON-RPC body, so MCP clients require no reconfiguration after a project cut-over. Not checked yet

New `project_migrations` table (fields: `project_id`, `workspace_url`, `cut_over_at`, `created_at`) tracks cut-over state; `@superlog/db` exports `isProjectCutOver` and `cutOverWorkspaceUrl` for per-project lookup cached 30 seconds per process. Not checked yet

+2 more

WAS THIS USEFUL? 👍 👎

Braintrust

Feedback ⓘ updated · Nov 23, 2026 NOTES ↗

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

Braintrust adds a Decisions API for LLM judges, pattern rules and scheduled digests for Loop, a grouped thread view, and an Inference activity page for Gateway token usage.

Enable the Decisions API for LLM-as-a-judge scorers so the model selects a choice directly and records confidence and per-choice probability — models like GPT-6 Luna use it when turned on, and custom models with the `decisions` flavor always use it. Not checked yet

Create pattern rules to narrow Loop's scheduled Patterns runs to specific traces with a SQL filter and define problems for Loop to watch for. Not checked yet

Send recurring project-activity summaries to your team with scheduled digests, choosing compact or detailed writing, a schedule, and destinations. Not checked yet

Explore Braintrust Gateway token usage on the new Settings > Inference activity page, filtering by project, feature, and whether requests use built-in models or your own provider keys. Not checked yet

View multi-trace sessions as a single ordered conversation in the new thread view, which groups traces by `metadata.conversation_id` including traces that log the ID only on child spans. Not checked yet

+9 more

NEW INTEGRATION OpenCode 2 tracing

NEW INTEGRATION Google Antigravity SDK tracing

NEW INTEGRATION aiobotocore async Bedrock Runtime tracing

NEW INTEGRATION LiveKit Agents tracing

NEW INTEGRATION GPT-Live / Pipecat full-duplex voice tracing

└──▷ BREAKING ON UPGRADE

! Terraform provider v0.0.9 now requires Terraform or OpenTofu 1.11.1 or later.

! `braintrust_ai_secret` now requires `secret_version` alongside `secret`, and `braintrust_env_var` requires `value_version` alongside `value`; changing only the value no longer updates it, and references to `secret` or `value` return `null`.

! The `braintrust_org_automation` Terraform resource is removed; delete its blocks from your configuration and run `terraform state rm` on each address before running `terraform plan`.

- ! LiveKit Agents now uses `livekit.llm_response` for LLM calls it traces directly, replacing `LLM request` and `Realtime request`; saved queries, filters, and scorers using the old span names must be updated.
- ! For text-to-speech calls traced directly by LiveKit, `time_to_first_audio` on the `tts` span has moved to `time_to_first_token` on the `livekit.tts` child span.
- ! OpenAI spans no longer attach audio by default; set `BRAINTRUST_CAPTURE_USER_AUDIO_ATTACHMENTS` (input audio) or `BRAINTRUST_CAPTURE_AGENT_AUDIO_ATTACHMENTS` (generated audio) to `true` to restore audio attachments on transcription, translation, speech, and chat spans.

WAS THIS USEFUL? 👍 👎

GOVERN

◆ Model & Agent Protection

Minimal Software Research minimal

Feedback ⓘ

release-7ff9a855 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 rolled out tab-completion for box names across every `min box` verb, cleaner piped log output, new sealed-box loadout enforcement, and retired the legacy stop/run-path commands in favor of their replacements.

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

01 **Removed bare `min stop` and `min task run --path`** Not checked yet

90

BREAKING

The bare `min stop` and `min task run --path` commands are removed and now register as unknown commands; use `min host stop` and `-C <dir>` instead.

— Exact before/after commands with a clear migration path. release-97724242

release-7ff9a855

02 **Tab completion for box names in `min box` commands**

Not checked yet IMPROVED

70

`min box` now tab-completes box names in every verb that accepts one, covering commands such as `min box show`, `min box audit --parent`, `min box show --network`, and `min box show -o json`.

Review the full audit trail for a box, including parent history, to verify no entries have been tampered with.

```
$ min box audit mybox --parent
```

Inspect a box's declared network effect to verify egress rules before running an agent workload.

```
$ min box show mybox --network
```

Export box details as JSON to feed into downstream tooling or audit pipelines.

```
$ min box show mybox -o json
```

Review the full audit trail for a box, merging its parent chain, to verify no egress decisions were tampered with.

```
$ min box audit mybox --parent
```

Inspect a box's declared network effect in machine-readable form to check egress posture in CI.

```
$ min box show mybox --network
```

— Names exact commands but not the completion mechanism or shells supported.

release-97724242 release-7ff9a855

03 **Loadout enforcement for sealed boxes (T126)** Not checked yet NEW

60

`min box` now refuses a named loadout on a sealed box, a loadout that sets `loadout_mode`, and a contradicted constraint, enforced via rule `T126`.

— Lists refusal conditions but gives no example invocation. release-7ff9a855

THINNER COVERAGE BELOW

04 **Strips terminal control codes from `min box logs`** Not checked yet

IMPROVED

50

`min box logs` now strips terminal control codes from its output when stdout isn't a terminal, keeping piped or redirected logs free of escape sequences.

— Clear behavior change but no flag or example to invoke it. release-7ff9a855

05 **Named QUIC block helper in `min finalize-install`** Not checked yet

IMPROVED

30

`min finalize-install` now names the QUIC block helper used during installation.

— Bare mention of a naming change with no further behavior detail. release-7ff9a855

└──▷ ALSO FROM THESE RELEASES

Store a secret token for a box so agents can access it without it appearing in the project file.

```
$ min secret set MY_API_TOKEN
```

└──▷ **BREAKING ON UPGRADE**

! The bare `min stop` and `min task run --path` commands are removed. Use `min host stop` and `-C <dir>` instead.

! The old session verbs `min stop` and `min task run --path` are now unknown commands; use `min host stop` and `-C <dir>` instead.

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Open-Weight Models

Alibaba Qwen Models

Feedback ⓘ updated · Nov 23, 2026

NOTES ↗ RANK

Qwen is Alibaba's language and multimodal model family, including coding, vision, image, speech, and reasoning models.

> Qwen added qwen-audio-3.1-live, a real-time voice conversation model that can keep talking with a user while a background task runs and then speak the result once ready.

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

01 **Real-time voice conversation with qwen-audio-3.1-live**

Not checked yet NEW 45

The new `qwen-audio-3.1-live` model adds real-time voice conversation capability, letting the voice model continue talking with a user while a background process handles a task, then speak the result once it's ready.

— Names the model but no API or usage details. snapshot-20261123

WAS THIS USEFUL? 👍 👎

Google Gemma 3 4B

Feedback ⓘ price cut · Nov 22, 2026

MODEL ↗

Gemma 3 4B is a multimodal model from Google.

Price cut on OpenRouter (google/gemma-3-4b-it): $0.05 → $0.04 in, $0.10 → $0.05 out per million tokens (-50%), first seen 22 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

Tencent Hy3

Feedback ⓘ price increase · Nov 22, 2026 MODEL ↗

Hy3 is a chat model from Tencent.

Price increase on OpenRouter (tencent/hy3): $0.132 → $0.14 in, $0.528 → $0.58 out per million tokens (+10%), first seen 22 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

Thinking Machines Inkling 2 Mini

Feedback ⓘ

price cut · Nov 22, 2026 MODEL ↗

Inkling 2 Mini is a multimodal model from Thinking Machines.

Price cut on OpenRouter (thinkingmachines/inkling-2-mini): $0.10 → $0.075 in, $0.40 → $0.30 out per million tokens (-25%), first seen 22 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

gryphe MythoMax 13B Feedback ⓘ

price increase · Nov 22, 2026 MODEL ↗

MythoMax 13B is a chat model from gryphe.

Price increase on OpenRouter (gryphe/mythomax-l2-13b): $0.08 → $0.30 in, $0.11 → $0.50 out per million tokens (+355%), first seen 22 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

ConvAI Innovations Laya Feedback ⓘ v0.5.2 NOTES ↗

RANK

Laya is a non-autoregressive System 1 decision engine. Typed choice, score and yes/no decisions over any text in a single forward pass, in 100+ languages, with a router that picks the right checkpoint per request.

> Laya v0.5.2 adds a targeted teacher-escalation path for uncertain learner samples.

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

01 **Escalator routes uncertain samples to vLLM teacher**

Not checked yet NEW 65

New `laya.Escalator` and `laya.vllm_teacher` route uncertain learner samples to a vLLM-backed teacher model for targeted remediation.

— Names exact modules but no usage or config detail v0.5.2

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · November 23, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue