THE AI TOOLCHAIN

NO. 096

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED NOVEMBER 20, 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 305 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 64 of 65 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.

GOVERN

Stop retrieval from leaking rows the caller isn't allowed to see

vim ? help

Cortex Search now evaluates row access and masking policies per caller, so a RAG answer can no longer surface records the asking user would be denied in SQL — previously the common fix was maintaining a second, pre-filtered index. gograph applies the same instinct to coding agents, with opt-in read/write/session rules enforced at the hook layer for Claude Code, Codex and Antigravity.

Snowflake Cortex AI · gograph

BUILD

Spend your context window on the task instead of the tool list

Collapsing the toolset into a single `run_python` entry point backed by a native C++ `_mx` SDK drops the schema the model must read on every turn from 103k to ~2k tokens. For long 3ds Max sessions that is the difference between the agent remembering what it was asked to do and compacting the task away.

3dsmax-mcp

EVALUATE

Score live agent conversations and judge runs without hand-labelling

Braintrust's Decisions API gives LLM-as-judge scorers a structured verdict surface, with a thread view for multi-trace sessions and native tracing for LiveKit Agents and Pipecat voice pipelines — voice agents have been close to unobservable in eval tooling. Weave's Signals tab scores agent conversations automatically, and ADK's multi-run eval metrics award partial credit on tool matching, so a run that picked the right tool with one wrong argument no longer reads as total failure.

Braintrust · CoreWeave · Google Agent Development Kit

Does **Opus 5** read better? 👍 👎

BUILD

◆ AI Coding Agents

yetone magpie

Feedback ⓘ v0.1.4594 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 biggest window adds project-level library export/apply and full Pi package management (CLI subcommands, manifest recognition, project write-through), alongside gateway-side upgrades — per-key routing overrides, automatic sub2api billing-rate sync, embedding/rerank model discovery, and new Responses API compact endpoints — plus smaller fixes to Claude account trust prompts, skill removal, and DeepSeek Harness hook support.

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

01 **Embedding and rerank model discovery across providers**

Not checked yet NEW 95

`GET /v1/models?kind=embedding,rerank` lists embedding and rerank models for any added provider, with each model exposed as `provider/model` with its `kind` and route (`supported_endpoint_types`). magpie also reads Together AI's `/v1/models` array and routes each model by its `type` field, placing embedding models on `/v1/embeddings` and rerank models on `/v1/rerank` automatically.

Discover all embedding and rerank models available across your configured providers, including newly supported Together AI retrieval models.

```
$ curl 'http://localhost:9090/v1/models?kind=embedding,rerank' -H
'Authorization: Bearer <key>'
```

— Runnable endpoint example included v0.1.4591

02 **CLI subcommands for project library management** Not checked yet

NEW 95

`magpie library project export` and `apply` move a project's skills, servers, subagents, hooks, plugins, Claude Code plugins, Pi packages, and instructions to another folder without keys or sign-in credentials; `magpie library project pi` gains `add` , `on` , `off` , `rm` / `remove` , `adopt` , and `list` subcommands that read and write the project's `.pi/settings.json` `packages` array, with `magpie library project pi adopt <project> <source>` preserving the original entry in `orig` so it can be toggled off and on without losing user data; and `library project hook add <dir> <hook> agents=<agent,...>` assigns individual library hooks to named agents'

project hooks files while `library project hook rm <dir> <hook>` removes magpie-managed entries without touching user-written ones, backed by a new `ProjectHooks` function in `internal/library/project_plugins.go`.

Turn a Pi package on for a project, writing its source into `.pi/settings.json`.

```
$ magpie library project pi add /path/to/project npm:pi-demo
```

Disable a Pi package without removing it — useful when you want to keep the entry but stop Pi from loading it.

```
$ magpie library project pi off /path/to/project npm:pi-demo
```

Inspect all Pi package states for a project in machine-readable form, e.g. to pipe into a script.

```
$ magpie library project pi list --json /path/to/project
```

Add the `bark` library hook to Claude and Codex for a specific project directory, so both agents run it at turn-end without wrapping it in a plugin.

```
$ magpie library project hook add . bark agents=claude,codex
```

Remove magpie's `bark` hook from a project while leaving any hooks the user wrote manually in their agent config files.

```
$ magpie library project hook rm . bark
```

— Exact CLI commands with full scope of carried items Full subcommand set with runnable examples Exact add/remove commands with runnable examples

v0.1.4581 v0.1.4571 v0.1.4567

---

03 **Skill removal without backup and symlinked skill handling**

Not checked yet IMPROVED 90

Remove a skill from the library without leaving a copy in magpie's backups by choosing the no-backup path (`d` then `D` in the TUI, or `RemoveSkillsHow` with `keep=false`); the skill's folder under `~/.magpie/backups` is deleted entirely rather than archived. The remove-skill confirmation sheet shows a `Keep a copy in magpie's backups` checkbox whose state persists in `localStorage` key `magpie.libSkillBackup`

(`'on'` / `'off'`), carrying the last choice forward. Skills linked from an external folder (e.g. `~/.agents/skills/`) now have only their symlink removed on deletion, leaving the original folder untouched and showing no backup-or-delete checkbox.

— Exact TUI keys and config keys named v0.1.4585

04 **Responses API compact endpoints** Not checked yet NEW 90

New endpoints `POST /v1/responses/compact` and `POST /responses/compact` are now routed to upstream providers whose own API is the Responses API, returning the provider's `response.compaction` object untouched.

Compact a conversation via a Responses-API-backed provider, getting back the sealed compaction object to reuse in a later `/responses` call.

```
$ curl -X POST http://localhost:<port>/v1/responses/compact \
  -H 'Authorization: Bearer <key>' \
  -H 'Content-Type: application/json' \
  -d '{"model":"<responses-api-
model>","instructions":"summarise","input":
[{"type":"message","role":"user","content":
[{"type":"input_text","text":"Compact this conversation"}]}]}'
```

— Runnable curl example included v0.1.4575

05 **Per-key routing strategy overrides** Not checked yet NEW 85

Each gateway key can set its own routing strategy for the routing groups it accesses via the Routing field, with values `smart` , `rotate` , `drain` , `free` , `fit` , `acme/sticky` , or `follow` to revert to the group's own routing. The strategy can be updated at runtime with the `routing-key` operation and the `Change{Key: id, Routing: ...}` struct, which Authenticate then reflects on the key's Routing field.

— Named fields and operation but no runnable example v0.1.4589

06 **Project resource source field in list output** Not checked yet IMPROVED 85

The `--json` flag on project list commands now includes an `items` array where each entry carries a `source` field — `'magpie'` for resources magpie placed and `'project'` for read-only resources the project's own files already provided. Project list output also shows the project's own skills, servers, hooks, and subagents in a separate read-only section, distinct from what magpie placed.

Distinguish which resources magpie manages vs. which came from the project's own config files, for auditing or scripting.

```
$ magpie library project list --json | jq '[.items[] |
select(.source == "project")]'
```

— Runnable jq filter example provided v0.1.4568

07 **Plugin manifest support across platforms** Not checked yet NEW

80

Pi packages that declare only extensions in `package.json`'s `pi.extensions` are now recognised as plugins, with their extension paths tracked in a new `extensions` field, and when given to a project for Pi their folder is written as a single entry in `.pi/settings.json` that stays one entry on reapply, is cleanly removed on withdrawal, and leaves any existing declaration for the same package untouched; Claude Code plugins now surface their skills, hooks, and subagents by reading `plugins/installed_plugins.json` and `enabledPlugins` from `settings.json` and `settings.local.json` under the Claude Code config folder, with a plugin installed from a git download retaining a stable name across updates; and Codex plugins with a `.codex-plugin/plugin.json` manifest are now detected and imported, merging skills and MCP servers from the Codex manifest (Claude Code fields taking precedence where both define the same key) while hooks defined in the Codex manifest are routed exclusively to Codex.

— Named config file and field but no direct CLI command Named files but no command surface for the reader Named manifest path and precedence rule, no CLI command

v0.1.4581 v0.1.4572 v0.1.4570

08 **Grok Build own-context compaction default** Not checked yet

IMPROVED

80

Grok Build now uses its own compaction threshold (85% of the context window) by default via GrokOwnCompact() when no size is set in `CompactAt` or `AgentCompacts`, instead of inheriting magpie's global default of 272K tokens that caused premature compaction on large-window models. The `ForAgent` method in `internal/settings/settings.go` now sets `FullContext = true` for Grok Build when no per-agent compact size is configured, allowing the model to run to its full window before compacting.

— Precise before/after numbers but no direct user command v0.1.4579

09 **Automatic relay billing-rate sync from sub2api** Not checked yet

NEW

80

Reads each relay key's group billing rate automatically from `GET /v1/sub2api/billing` in the background via `KeepKeyRatesRead`, refreshing every few hours so relay price changes require no manual update. Adds `TopUpRate` to the provider configuration to record what a relay dollar costs, letting each key bill at its

group's `group_rate_multiplier` times that rate. Keys with a manually set `KeyPriceRates` value keep that rate; keys whose background read has never succeeded fall back to the provider's `PriceRate` rather than assuming 1×.

— Endpoint and config fields named, no direct CLI example v0.1.4577

---

10 **DeepSeek Harness (dsh) hooks support** Not checked yet NEW

75

Adds `dsh` to the list of agents magpie can write hooks for, storing rules in `$DSH_HOME/magpie-hooks.json` in Claude Code's hooks shape and auto-installing the `@deepseek-ai/dsh-hooks-claude-code` plugin per profile before writing the config row. A `magpie-hooks` plugin row is added to each DeepSeek Harness profile's patch list, referencing `$DSH_HOME/magpie-hooks.json` via the `configPath` key under the plugin config.

— Named env var and config path, no direct CLI command shown v0.1.4569

---

11 **Pre-trust caller folder for Claude accounts** Not checked yet NEW

60

Adds `trustCaller` logic so a saved Claude account's run in the caller's working folder starts with that folder pre-recorded as trusted in the account's own config file, preventing Claude Code's trust prompt from stalling each account in turn at startup.

— No command or config key the reader can invoke directly v0.1.4594

THINNER COVERAGE BELOW

12 **Unmanaged agent hooks visibility and settings file access**

Not checked yet NEW

40

magpie now lists, imports, and removes agent hooks it did not write, and shows each agent's settings file location from the agent row's menu in the Agents view.

— Only a brief summary, no mechanism detail given v0.1.4588

---

13 **Auto-add externally selected model to served list** Not checked yet

IMPROVED

40

Selecting a model that is not currently in a provider's served list now adds it to that list automatically, so agents see it alongside the dashed models.

— No named flag or command, just behavior description v0.1.4582

WAS THIS USEFUL? 👍 👎

Block Goose

Feedback ⓘ v1.60.2 NOTES ↗ CODE ↗ RANK

Goose is an open-source AI agent with a desktop app and CLI for coding and other automated tasks.

> Goose shipped native desktop apps for macOS, Linux, and Windows, added Agent Client Protocol support and an oi-kit agent framework, expanded local model inference, grew its model catalog past 85 providers, and added Gemini 4 Neon with full context and thinking-level support.

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

01 **Gemini 4 Neon model added** Not checked yet NEW 80

Adds `gemini-4-neon` to the model catalog with a 1,048,576-token context, 65,536 max output tokens, and thinking levels low/medium/high/xhigh, matching the existing Gemini 4 hydrogen entry.

— Names model id and exact context/output/thinking specs v1.60.2

THINNER COVERAGE BELOW

02 **Built-in local model inference** Not checked yet NEW 45

Goose can run local models directly inside the app without Ollama or a separate inference server, giving users control over inference and visibility into how models run.

— Describes behavior but no setup steps or flags product docs

03 **oi-kit agent framework** Not checked yet NEW 45

Agents can be built on the Open Intelligence Kit (`oi-kit`), which runs across native apps, servers, and the browser and can resume from saved state.

— Names oi-kit and resume mechanism, no usage example product docs

04 **85+ model providers supported** Not checked yet IMPROVED 45

Goose now supports over 85 model providers, including Anthropic, OpenAI, Gemini, OpenRouter, and Amazon Bedrock.

— Names providers and count but no config steps product docs

05 **Native desktop app for macOS, Linux, Windows** Not checked yet

NEW

40

Goose adds a native desktop app for macOS, Linux, and Windows alongside the existing CLI.

— Names platforms but no install or UI detail product docs

06 **Agent Client Protocol support** Not checked yet NEW

40

Goose can be used as an agent backend in other applications through the Agent Client Protocol (ACP).

— Names the protocol but no integration steps product docs

WAS THIS USEFUL?  

OpenAI Codex CLI

Feedback ⓘ rust-v0.173.0-alpha.1 NOTES ↗

CODE ↗ RANK

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

> Across three releases, Codex CLI added remote app-server support for `codex exec --remote`, a default-on TUI workspace panel with document and image previews and subagent transcripts, task-group navigation and remote-control pairing in the TUI, custom MCP status lines, and several API refinements including removal of the `thread/sample` prompt limit and thread-ID filtering on `thread/list`.

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

02 **Custom MCP status lines** Not checked yet NEW 80

Codex CLI exposes custom status lines, first surfaced in `/experimental`, then formalized as author- and user-selectable custom MCP status line providers accessible via the `/statusline` command in the TUI when `tui_extensions_statusline` is enabled in the app server's configuration.

– Named command and config flag but no example rust-v0.171.0 rust-v0.173.0-alpha.1

06 **Remote control pairing commands** Not checked yet NEW 65

The TUI gains `/rc` and `/remote` commands to manage remote control pairing and display device pairing codes.

– Named commands but no walkthrough given rust-v0.171.0

07 **Workspace panel enabled by default** Not checked yet IMPROVED 60

The TUI workspace panel is now enabled by default, adding searchable document previews, image previews, and subagent transcripts alongside the main conversation.

– Describes new panel contents but no navigation path rust-v0.171.0

THINNER COVERAGE BELOW

WAS THIS USEFUL? 👍 👎

OpenQodex

Feedback ⓘ openqodex@0.22.0 NOTES ↗ RANK

OpenQodex is an open source AI code review for coding agents. A separate reviewer checks every change your agent writes before you push, in Claude Code, Codex, Cursor and Cline.

> OpenQodex's code graph gained deeper route, type and package analysis in this release, including gin service route reading, cross-language gap reporting, and shape-based TypeScript interface resolution.

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

01 Gin service route reading in graph routes

Not checked yet NEW

95

The `graph routes` command now reads gin services, including engines, groups with their prefixes, `Use` middleware, and handlers; inline route handlers like app.get("/x", (c) => ...) are treated as first-class definitions, and `--exclude-tests` omits routes whose application is built in a test file.

— Names exact command, flag and inline-handler example. openqodex@0.22.0

02 Expanded TypeScript reference finding in graph references

Not checked yet IMPROVED

95

`openqodex graph references` on a TypeScript type now finds uses in interface properties, type aliases, type arguments (new Map<string, Repo>(), `extends Holder<Repo>`), type parameter constraints and defaults, and `implements` / `extends` clauses; files that import a type but whose usage is not read now set the floor and name the file.

— Names exact command with multiple concrete code examples. openqodex@0.22.0

03 Cross-language gap reporting in code graph answers

Not checked yet NEW

85

Code graph answers now surface unread-language gaps — in repositories containing Rust, C, C++, C#, Kotlin, Swift, PHP, or Scala files — with reasons like 2 Rust files not read ; each answer gains an `unknown.groups` field breaking gaps down by cause with a count and up to two examples, plus a `listIncomplete` flag and a certain (list incomplete) marker in CLI output.

— Names new field and flag but no command to invoke. openqodex@0.22.0

04 Shape-based TypeScript interface resolution

Not checked yet NEW

85

TypeScript interface calls in the code graph now also reach classes that match the interface by structural shape (without `implements`), recorded with evidence `by-`

`shape`; resolution is capped at 20 classes per call, and a `fan-out-capped` gap marks when the list is incomplete.

— Names evidence tag and cap, but no runnable example. openqodex@0.22.0

05 **Per-line import counting in graph packages** Not checked yet

IMPROVED 75

`openqodex graph packages` now counts every import line separately, so a file that imports one package twice (e.g. `import type { Turn }` and `import { TurnSchema }`) records two import sites rather than one.

— Names command and exact before/after counting behavior. openqodex@0.22.0

WAS THIS USEFUL? 👍 👎

Letta Code

Feedback ⓘ v0.34.67 NOTES ↗ RANK

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

> Letta Code added support for configuring cloud repositories so agents can load skills directly from them.

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

01 **Cloud repository skill loading** Not checked yet NEW 30

Agents can now configure cloud repositories and load their skills from them.

— No mechanism, config key or command named. v0.34.67

WAS THIS USEFUL? 👍 👎

Factory

Feedback ⓘ

CLI v1.16.0, Desktop v1.16.0: Clickable login and MCP links, Script step titles, lower u

BLOG ↗ RANK

Factory's Droids are AI software-development agents with a CLI, web app and API.

> Factory's v1.16.0 release for the CLI and Desktop apps focuses on usability polish—clickable login and MCP links, labeled script steps, and an org-level metric prefix—while tightening usage limits and locking down legacy workspace and sync settings.

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

01 **Exportable metric name prefix for organizations** Not checked yet

NEW

45

Organizations can now set a name prefix for metrics Droid exports to their collector via the new exported metric name prefix setting.

– Names the setting but not where to configure it

CLI v1.16.0, Desktop v1.16.0: Clickable…

02 **Clickable login and MCP authorization links** Not checked yet

IMPROVED

35

The login screen now shows a clickable link that carries your device code, removing the need to copy it manually, and MCP authorization links are now clickable in MCP views and command output.

– Clear UX change across two surfaces, no deeper mechanism given

CLI v1.16.0, Desktop v1.16.0: Clickable…

03 **Lower usage limits on paid plans** Not checked yet BREAKING

35

Most paid plans now have lower five-hour and weekly usage limits, though monthly limits remain unchanged.

– States exact scope but no numbers or plan names

CLI v1.16.0, Desktop v1.16.0: Clickable…

04 **Script step titles in call output** Not checked yet IMPROVED

30

Script steps now show each step's title alongside a bar in its call output, making multi-step runs easier to follow.

– Describes UI change but no config or command given

CLI v1.16.0, Desktop v1.16.0: Clickable…

05 **Session contents sync locked outside Enterprise** Not checked yet

30

BREAKING

Plans outside Enterprise can no longer turn session contents sync off; the setting is now locked.

– Clear before/after but no setting path named CLI v1.16.0, Desktop v1.16.0: Clickable…

06 **Legacy Cloud workspace sessions now read-only** Not checked yet

DEPRECATED

30

Legacy Cloud workspace sessions are now read-only history, with no restart or download actions available.

– States behavior change but no migration path given

CLI v1.16.0, Desktop v1.16.0: Clickable…

└──▷ **BREAKING ON UPGRADE**

! Most paid plans now have lower five-hour and weekly usage limits (monthly limits unchanged).

! Plans outside Enterprise can no longer turn session contents sync off; the setting is now locked.

! Legacy Cloud workspace sessions are now read-only history, with no restart or download actions available.

WAS THIS USEFUL? 👍 👎

flow-next

Feedback ⓘ flow-next-v9.17.0 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 now lets any model family act as a reviewer via a pluggable command backend, automatically re-routes reviewed specs back to plan review when cited files drift, and lets teams share review criteria from a central guidelines repo.

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

01 **Command review backend for any model family** Not checked yet NEW

95

Any model family can serve as the code reviewer: set `review.backend` to `command:<name>` in config and point `FLOW_REVIEW_COMMAND_<NAME>` (its argv given as a JSON array) at a wrapper script, selectable via `review.backend`, `FLOW_REVIEW_BACKEND`, `--review`, per-spec and per-task pins, or fan-out draws. Review arguments are passed only through the environment variable and are never written to tracked config or the branch under review — a `review.commands` key in `.flow/config.json` is never read, and a review that finds one stops and names the variable to set instead. Review receipts now record `mode: 'command'`, the label, and the model the wrapper reported (or `not reported` if none).

Route all plan, implementation and completion reviews through a local Grok wrapper so you can use a model family not available via Copilot or Cursor subscriptions.

```
$ export FLOW_REVIEW_COMMAND_GROK='["opencode", "run", "--model", "grok-4"]'
/flow-next:flow --review=command:grok <spec-id>
```

Route all review gates through Grok by declaring the backend in config and pointing the environment variable at your local wrapper script.

```
$ export FLOW_REVIEW_COMMAND_GROK=~/bin/grok-review-wrapper
```

Pin the Grok backend in flow-next config so every review gate uses it without a per-run flag.

```yaml
review.backend: command:grok
```

— Names config key, env var, flags and a runnable wrapper command `flow-next-v9.17.0`

02 Stale-spec drift returns specs to plan review

Not checked yet

90

BREAKING

A reviewed spec whose cited files change before work starts is automatically routed back to plan review, including under `flow --auto` and goal mode, with the changed paths reported as `plan_review_drift` in `flowctl show <spec> --json`. After upgrading from 9.16.0, a SHIP spec already plan-reviewed by a CLI backend (`codex`, `copilot`, `cursor` or `claude`) is sent back to plan review immediately if any cited file changed and no task has started.

Inspect which cited files triggered a drift back to plan review after upgrading, so you know exactly what changed before restarting the build.

```
$ flowctl show <spec-id> --json | jq '.plan_review_drift'
```

— Gives exact JSON field and inspection command, notes upgrade impact `flow-next-v9.17.0`

THINNER COVERAGE BELOW

03 Shared review criteria across a guidelines repo

Not checked yet

NEW

55

Teams can reach shared review criteria from one guidelines repository via two documented methods in `.flow/criteria.md`: a symlink to a shared criteria file, or a criterion that names a shared guidelines file by path.

— Names the config file and two methods but no example `flow-next-v9.17.0`

└──▷ BREAKING ON UPGRADE

! A SHIP spec that a CLI backend (`codex`, `copilot`, `cursor` or `claude`) plan-reviewed on 9.16.0 already records the commit its review saw; if any cited file has changed and no task has started, flow sends it back to plan review immediately after upgrade.

! A command reviewer's argv now comes only from `FLOW_REVIEW_COMMAND_<LABEL>` on the reviewing machine; a `review.commands` key in `.flow/config.json` is never read, and a review that finds one stops and names the variable to set instead.

WAS THIS USEFUL? 👍 👎

Anthropic Claude Code

Feedback ⓘ v2.1.329 NOTES ↗

CODE ↗ RANK

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

> Claude Code's single release in this window focused on observability and trust: new diagnostic surfaces in `/mcp`, `/status`, and `/tasks`, OpenTelemetry support inside Code Mode, and clearer visibility into hook provenance and failures.

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

01 MCP server error detail in /mcp

Not checked yet NEW 62

`/mcp` shows the raw JSON-RPC error message a remote HTTP MCP server returned when it refused a connection, surfaced in a new Server message row on that server's page, to help diagnose connection failures.

Diagnose why a remote HTTP MCP server is refusing connections by reading the raw JSON-RPC error message surfaced on its page.

```
$ /mcp
```

— Names command and exact UI row, mechanism is brief v2.1.329

02 Scheduled-task visibility in /tasks

Not checked yet NEW 60

`/tasks` now surfaces scheduled tasks and loop wakeups during a long agentic run, letting users check what is queued and cancel a runaway task without leaving the session.

During a long agentic run, check what scheduled tasks and loop wakeups are queued — and cancel a runaway one — without leaving the session.

```
$ /tasks
```

— Names command and exact use case, but no deeper mechanism given v2.1.329

03 Marketplace auto-update status in /status

Not checked yet NEW 60

`/status` adds a Marketplace auto-update row that identifies managed marketplaces receiving no automatic updates and states the reason why.

Quickly spot which managed marketplaces are not receiving automatic updates and understand why before troubleshooting stale tooling.

```
$ /status
```

— Names command and row but little further mechanism v2.1.329

THINNER COVERAGE BELOW

04 **OpenTelemetry support in Code Mode** Not checked yet NEW

55

Code Mode can now run while OpenTelemetry events and metrics are active; each tool call emits its own events, though tracing remains off in Code Mode.

— Explains behaviour and a before/after limitation but no command shown v2.1.329

05 **Hook provenance in trust dialog** Not checked yet IMPROVED

45

The trust dialog now shows where a folder's hooks are declared, letting users audit hook provenance before granting trust.

— Describes the change but gives no concrete navigation or example v2.1.329

06 **Hook load-failure warning on non-interactive hosts**

Not checked yet NEW

45

Non-interactive hosts such as Claude Desktop now show a warning in the conversation when hooks in managed settings fail to load.

— States the behaviour but lacks concrete reproduction steps v2.1.329

WAS THIS USEFUL? 👍 👎

keploy

Feedback ⓘ v3.7.11 NOTES ↗ RANK

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

> Keploy's CLI now automatically names a new app based on its git repository and directory path.

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

01 **Automatic app naming from git repo path** Not checked yet IMPROVED

35

Keploy's CLI now names a new app after its git repository and directory path automatically when creating an app, removing the need to manually specify an app name.

— Describes behaviour but no flag or command shown v3.7.11

WAS THIS USEFUL? 👍 👎

mjolnir

Feedback ⓘ v2.72.0 NOTES ↗ RANK

Mjolnir is a manage Codex, Claude Code, Muse Code, Kimi Code, Grok Build, and DeepSeek Harness with durable sessions, isolated environments, quotas, and remote control. A Rust terminal and web control plane for Agent Client Protocol (ACP) coding agents.

> Mjolnir v2.72.0 adds DeepSeek Harness (DSH) history indexing into its shared project memory, improves checkpoint traceability in Observe, and tightens relay budget enforcement with clearer error messages.

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

01 **Forward-compatible config key reading** Not checked yet IMPROVED

60

Clients can now read configuration keys added after they were built, allowing newer `config.toml` settings to be consumed by older client builds without error.

– Names the exact config file and explains compatibility mechanism v2.72.0

THINNER COVERAGE BELOW

02 **DSH history indexing via SessionWiki** Not checked yet NEW

50

Mjolnir now indexes DeepSeek Harness (DSH) histories using SessionWiki native adapters, making DSH session history available through the shared project memory system.

– Names the mechanism (SessionWiki adapters) but no config or command v2.72.0

03 **Relay budget enforcement on prompts** Not checked yet IMPROVED

40

Prompts are now held to the worker's relay budget, and refused limits are named explicitly in error output.

– Describes behaviour change but no specific limit values or flags v2.72.0

04 **Checkpoint location reporting in Observe** Not checked yet IMPROVED

35

The Observe view now reports where each checkpoint finds a session's checkout, making it easier to trace session state across targets.

– Names the Observe view but no concrete fields or UI steps v2.72.0

WAS THIS USEFUL?  

Charm Crush

Feedback ⓘ v0.102.0 NOTES ↗ CODE ↗ RANK

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

> Crush v0.102.0 adds configurable system prompts across its agent pipeline, real-time MCP Channels for pushing server events into sessions, and auto-detected Nerd Font icon rendering with manual override.

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

01 MCP Channels for real-time server events

Not checked yet NEW

88

MCP Channels let MCP servers push events into a session in real time. Enable them per-invocation with `--channels server:name` on the CLI, or permanently per-server via `--channel true` in `mcp add`, stored in `crushrc`.

Permanently attach a webhook MCP server as a Channel so it can push events into every session.

```
$ mcp add webhook \
    --type http \
    --url https://example.com/mcp \
    --channel true
```

— Named flags, config file, and a runnable command example v0.102.0

02 Custom system prompts per agent stage

Not checked yet NEW

80

Set custom system prompts for the main agent, subagents, planning, compaction, session titles, and `crush init` using `option prompt <name> <path>` in config, or override per-invocation with `--prompt name=path`.

— Names config key and flag but no runnable example given v0.102.0

03 Nerd Font icon rendering control

Not checked yet IMPROVED

75

Nerd Font icon rendering is auto-detected by default and can be forced with `option ui nerd-fonts true` or `option ui nerd-fonts false` in config, which is required when running Crush over SSH since auto-detection cannot reach the host terminal.

Enable Nerd Font icons when running Crush over SSH, where auto-detection cannot reach the host terminal.

```text
option ui nerd-fonts true
```

— Exact config key given with SSH use case and example v0.102.0

WAS THIS USEFUL? 👍 👎

Amazon Kiro Feedback ⓘ

Crew: Conductor Mode for Everyone and Replies From Anywhere (2026-11-19) BLOG ↗

RANK

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

> Kiro Crew's 0.13 release makes conductor mode standard for all builds, adds multi-channel session notifications, and ships a batch of new agent, prompt, and safety controls.

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

01 Sandboxed agent read-only path controls Not checked yet NEW

85

List read-only folders for the sandboxed agent using `agent.sandbox_readonly_paths`, and keep chosen sub-folders writable with `agent.sandbox_readonly_exceptions`, for example restricting `/etc/secrets` and `/infra` while leaving `/infra/scratch` writable.

Prevent the sandboxed agent from writing to secrets and infra folders while still allowing writes inside a designated scratch sub-folder.

```
$ kirocrew config set agent.sandbox_readonly_paths
'["/etc/secrets", "/infra"]'
kirocrew config set agent.sandbox_readonly_exceptions
'["/infra/scratch"]'
```

— Runnable commands with exact config keys and example paths.

Crew: Conductor Mode for Everyone and R…

02 Slash commands from saved prompts Not checked yet NEW

85

Create saved prompts as slash commands by adding `slash_command: true` to a prompt file's front matter in `~/.kiro/prompts`, then invoke it as `/<file name>`.

Turn a frequently reused prompt into a slash command so the agent can invoke it by name in any chat.

```yaml
---
title: "Summarise PR"
slash_command: true
---
Summarise the open pull request and list action items.
```

— Exact file path, front-matter field, and invocation syntax given.

Crew: Conductor Mode for Everyone and R…

03 **Batch write guard for unseen reads** Not checked yet BREAKING

80

A new batch write guard, on by default, refuses a write sent beside a read when it quotes an id or value the agent has not seen yet; set `hooks.batch_write_guard` to `false` to disable it.

— Explains exact trigger mechanism and config key to disable.

Crew: Conductor Mode for Everyone and R…

04 **Custom credit-usage alert thresholds** Not checked yet NEW

75

Send credit-usage alerts at custom thresholds by setting `telemetry.plan_alert_pcts` to a list of percentages, e.g. `kirocrew config set telemetry.plan_alert_pcts '[50, 80, 95]'`.

Alert yourself when your plan has used 50%, 80%, and 95% of its credits so you can act before hitting the limit.

```
$ kirocrew config set telemetry.plan_alert_pcts '[50, 80, 95]'
```

— Runnable command with exact config key shown. Crew: Conductor Mode for Everyone and R…

05 **Quick reply hover card for waiting sessions** Not checked yet NEW

75

Answer waiting sessions from a hover card by resting the pointer on a session or pressing the right arrow key to see its recap, approvals, questions, and a quick reply box; toggle the card off via Quick Reply Card on Hover under Settings → Chat → Sessions.

— Describes interaction mechanism and toggle location.

Crew: Conductor Mode for Everyone and R…

06 **Telemetry enabled by default with opt-out controls**

Not checked yet

BREAKING

70

Usage data collection is now on by default in public builds; it can be disabled via `telemetry.enabled: false`, the `KIROCREW_TELEMETRY=0` environment variable, or `DO_NOT_TRACK=1`.

— Exact config key and two env vars given, no further mechanism.

Crew: Conductor Mode for Everyone and R…

07 **New session keyboard shortcuts** Not checked yet NEW 65

New shortcuts let you reopen the last closed session with Cmd+Shift+T (Ctrl+Shift+T on Windows/Linux, Alt+Shift+T in a browser tab), refuse a waiting tool call with Option+Shift+R (Alt+Shift+R), and upload a file with Cmd+Shift+U (Ctrl+Shift+U).

— Exact key combos for three distinct actions. Crew: Conductor Mode for Everyone and R…

08 **Queued external config edits on Linux** Not checked yet BREAKING 65

On Linux, external edits to `config.json`, `config.local.json`, `denied_commands.json`, or `computer_use.json` that Kiro Crew does not recognize now wait for you to apply or discard them before taking effect.

— Names exact files affected but UI step not detailed.

Crew: Conductor Mode for Everyone and R…

09 **Notification delivery to Slack, Discord, Telegram, Webex, Feishu**

Not checked yet NEW 60

Session notifications can now be delivered as DMs to Slack, Discord, Telegram, Webex, or Feishu by configuring Deliver to under Settings → Notifications → Sources for each source row.

— Names five channels and UI path but no API detail.

Crew: Conductor Mode for Everyone and R…

10 **Chat forking with /fork** Not checked yet NEW 60

Fork a chat into a copy of the conversation and continue from a new prompt using `/fork <prompt>`.

— Exact command given but no further behavior detail.

Crew: Conductor Mode for Everyone and R…

THINNER COVERAGE BELOW

11 **Conductor mode on by default for all builds** Not checked yet 55

BREAKING

Conductor mode (formerly Ultra Mode) is now on in every standard build with no Feature Preview switch required; start new chats in conductor mode under Settings → Chat → Sessions. It has left Settings → Developer → Feature Previews.

— UI path given but no mechanism of conductor mode itself.

Crew: Conductor Mode for Everyone and R…

12 **Settings navigation reorganized** Not checked yet BREAKING

55

The Releases page moved; release notes now live in Settings → About. MCP Servers moved to their own tab under Customize, beside Skills; Settings → Connections now keeps only the provider cards.

— Exact new locations given but no functional change explained.

Crew: Conductor Mode for Everyone and R…

NEW INTEGRATION Amazon VPC connectivity for Kiro cloud sessions

└──▷ **BREAKING ON UPGRADE**

! Ultra Mode is renamed to conductor mode; it has left Settings → Developer → Feature Previews and needs nothing turned on.

! The Releases page moved; release notes now live in Settings → About.

! MCP Servers moved to their own tab under Customize, beside Skills; Settings → Connections now keeps only the provider cards.

! Usage data (`telemetry.enabled`) is on by default in public builds.

! The batch write guard (`hooks.batch_write_guard`) is on by default; it refuses a write sent beside a read when it quotes an id or value the agent has not seen yet — set it to `false` to turn it off.

! On Linux, external edits to `config.json` , `config.local.json` , `denied_commands.json` , or `computer_use.json` that Kiro Crew does not recognize now wait for you to apply or discard them before taking effect.

General: Connect Cloud Sessions to Your VPC (2026-11-19) BLOG ↗ LATEST

STABLE

Kiro adds Amazon VPC connectivity for cloud sessions in preview on Enterprise plans, letting teams reach private resources without making them public.

Connect Kiro cloud sessions to private Amazon VPC resources by creating a VPC connection in the Kiro console and setting it as the profile default in Controls, so teams can reach internal resources without making them public. Not checked yet

VPC connections are available in preview on Enterprise plans for IAM Identity Center profiles in the `us-east-1` region; the profile must use Controls, either from creation

> or after migration. Not checked yet

WAS THIS USEFUL? 👍 👎

◆ Model Training & Fine-Tuning

NVIDIA Megatron Bridge

Feedback ⓘ v0.7.1 NOTES ↗

RANK

Megatron Bridge connects Hugging Face model workflows with Megatron Core for distributed model training and checkpoint conversion.

> Megatron Bridge adds a dataloader prefetch control for full-iteration CUDA graph support and corrects GB200 fine-tuning recipes for Nemotron 3 Super.

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

01 Dataloader prefetch_factor for CUDA graph support

Not checked yet

NEW

85

The `prefetch_factor` option in the dataloader config sets the number of batches workers buffer ahead of time, enabling full-iteration CUDA graphs that require a whole step of microbatches to be loaded before the graph replays. Example: `prefetch_factor: 4` under `dataloader:` in YAML config, increasing beyond PyTorch's default of 2 batches.

Increase the dataloader prefetch depth beyond PyTorch's default of 2 batches to satisfy full-iteration CUDA graph requirements, where all microbatches in a step must be resident before graph replay.

```yaml
dataloader:
  prefetch_factor: 4
```

— Names exact config key, file location, and default behavior v0.7.1

THINNER COVERAGE BELOW

02 GB200 fine-tuning recipes for Nemotron 3 Super

Not checked yet

IMPROVED

45

Dedicated GB200 SFT and PEFT fine-tuning recipes for Nemotron 3 Super now ship with GB200-tuned defaults, replacing previously published commands that incorrectly used H100 HybridEP topology settings.

— Names the fix but no recipe commands or paths given v0.7.1

WAS THIS USEFUL? 👍 👎

fizgig

Feedback ⓘ v9.8.2 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 v9.8.2 release adds full MiniMax H3 Image fine-tuning (not just LoRA training), lets Studio render that model at much larger sizes, and ships smaller UI conveniences for unloading LoRAs and saving mask-brush presets.

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

01 **MiniMax H3 Image fine-tuning** Not checked yet NEW 90

Train a full MiniMax H3 Image fine-tune (not a LoRA) from the Training tab's Fine-tune kind, using the built-in MiniMax H3 Image Fine-tune (5e-5) preset. It runs on a 24 GB card and produces a full H3 Image file (~21 GB) that can be loaded by Studio, LoRA Royale, and ComfyUI.

— Names exact preset, hardware requirement, output size and compatible tools v9.8.2

02 **Brush presets in Image Prep mask editor** Not checked yet NEW 70

Edit masks in Image Prep can now save up to three brush presets using buttons 1 2 3 (or number keys), each storing a tool's Size, Softness, and Strength via a Keep these settings… action.

— Names exact buttons, keys and saved parameters with clear UI path v9.8.2

03 **Larger MiniMax H3 Image render sizes** Not checked yet IMPROVED 65

Studio can now render MiniMax H3 Image at resolutions up to 12288 × 12288, covering the model's native range of 4–20 MP and formats such as 7680 × 2176 for 32:9 ultrawide.

— Gives concrete resolution limits and formats but no exact UI steps v9.8.2

THINNER COVERAGE BELOW

04 **Unload primary LoRA in Studio** Not checked yet NEW 55

A new Unload primary button next to the primary's Choose… control in Studio removes a loaded LoRA without requiring the whole model to be reloaded.

— Names the exact button but little mechanism beyond that v9.8.2

WAS THIS USEFUL? 👍 👎

NVIDIA Megatron-LM

Feedback ⓘ core_v0.20.1 NOTES ↗ RANK

Megatron-LM trains transformer models at scale with tensor, pipeline, sequence, and context parallelism, and ships Megatron Core as a reusable library.

> Megatron-LM core_v0.20.1 adds a new Mixture-of-Kittens MoE backend and bumps the TransformerEngine dependency to 2.21.

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

01 **TransformerEngine dependency upgraded to 2.21** Not checked yet

IMPROVED 35

The TransformerEngine dependency is upgraded from version 2.20 to 2.21.

— Exact version bump named, no further detail on changes. core_v0.20.1

02 **Mixture-of-Kittens MoE backend** Not checked yet NEW 25

Megatron-LM adds a Mixture-of-Kittens (MoK) MoE backend for training transformer models.

— Names the backend but no mechanism or usage detail given. core_v0.20.1

WAS THIS USEFUL? 👍 👎

◆ Coding Agent Add-ons

Token Optimizer

Feedback ⓘ v5.13.74 NOTES ↗ RANK

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

> Token Optimizer ships a pilot-report command for summarizing savings, an opt-out for model-routing nudges, a self-test for the Claude Code mod's Keep warm button, and smarter activity-based detection of stale sessions.

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

01 **Pilot-report summary command** Not checked yet NEW 75

The new `measure.py pilot-report` command outputs a paste-ready summary covering capacity gained, savings since install and for the current week, routing status, and nudges followed — useful for sharing recovered context/cost at the end of a sprint.

Generate a shareable savings summary at the end of a sprint to show how much context capacity and cost has been recovered since install.

```
$ measure.py pilot-report
```

— Names exact command and its outputs but not internals v5.13.74

02 **Model-routing opt-out environment variable** Not checked yet NEW 68

Setting `TOKEN_OPTIMIZER_ROUTING=0` disables model-routing guidance nudges, for environments where model selection is already locked by policy.

Disable model-routing nudges in an environment where your model selection is already locked by policy.

```
$ export TOKEN_OPTIMIZER_ROUTING=0
```

— Exact env var and purpose given, no further mechanism v5.13.74

03 **Activity-based stale session detection** Not checked yet IMPROVED 61

Session health now judges activity rather than age, so `kill-stale` only ends sessions that Claude Code confirms have been idle past the threshold — busy orchestrator or delegate sessions running for days are no longer flagged ZOMBIE.

— Explains before/after mechanism but no exact threshold value v5.13.74

04 **Keep warm button self-test for Claude Code mod** Not checked yet

NEW

60

The Claude Code mod's Keep warm button can now be verified with `measure.py mod-action --selftest`.

— Exact command given, but behavior/output unexplained v5.13.74

WAS THIS USEFUL? 👍 👎

◆ Image & Video Generation

Runway

Feedback ⓘ updated · Nov 20, 2026 NOTES ↗ RANK

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

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

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

01 **App Studio plain-language app builder** Not checked yet NEW

45

App Studio lets users describe a creative app in plain language and Runway builds it, brands it with the user's own files, and hosts it automatically. For example, typing a description like a branded video resizing tool and uploading brand assets, then clicking Build, generates and hosts the app.

When you need a branded creative tool for your team without writing code — describe what you want and App Studio generates and hosts it.

📍 In App Studio, type a plain-language description of your app (e.g. 'A video resizing tool branded with our logo and color palette'), upload your brand assets, then click Build to generate and host the app.

— Describes workflow and UI steps but no API or config detail snapshot-20261120

WAS THIS USEFUL? 👍 👎

AGENT

◆ Web Search & Scraping for Agents

Perplexity API

Feedback ⓘ updated · Nov 20, 2026 NOTES ↗

RANK

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

> Perplexity API swapped the model backing its `xhigh` effort level to a new version with no fallback, a breaking change for callers relying on the prior fallback chain.

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

01 **Model change for `xhigh` effort level** Not checked yet BREAKING

90

The `xhigh` effort level now routes to `xhigh@2026-11-18.0`, backed by `openai/gpt-6.2-sol` with no fallback model, replacing the previous configuration of Claude Opus 5.5 with an `openai/gpt-6.1-sol` fallback. Callers that depend on the old fallback chain must pin `xhigh@2026-11-11.0` to preserve that behaviour.

— Names exact model IDs, version pins, and migration path. snapshot-20261120

└──▷ BREAKING ON UPGRADE

! The `xhigh` effort level now uses `openai/gpt-6.2-sol` with no fallback model; callers that relied on the previous Claude Opus 5.5 / `openai/gpt-6.1-sol` fallback chain must pin `xhigh@2026-11-11.0` to preserve that behaviour.

WAS THIS USEFUL? 👍 👎

◆ AI Agent Frameworks

claw-orchestrator

Feedback ⓘ v7.28.0 NOTES ↗ RANK

Claw-orchestrator is a run Claude Code, Codex, Antigravity, Grok Build and OpenCode behind one runtime, and a run only counts as done when checks it didn't write pass. Cross-engine fan-out, councils, session handoff, spend caps, MCP server, ACP agent.

> claw-orchestrator introduced a shared run forum letting workflow agents post, list, and read coordination messages with score attachment, delivery controls and activity metrics, added mid-turn message delivery into running agent sessions, and shipped contract fingerprinting with drift detection on workflow resume alongside split run-status reporting.

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

01 Shared forum for workflow agents

Not checked yet NEW

98

A shared message board for all agents in a workflow, enabled by setting `forum` on a workflow spec and used via `clawo forum post | list | read`; posts can carry `push: "all"`, `push: "digest"`, or `push: "none"` to control how ASK and FAIL messages are pushed to agents, and an agent's current score can be attached to a post from a file or command (e.g. `--score-from results/score.txt`). Score attachment now supports field filtering with `score.pick` (e.g. `["ok", "failures"]` or `"summary.passed"`), the score is parsed whole and capped at 2,000 characters, and `read` shows the title first then the score as one line of at most 300 characters after the body; scores are read from the directory the agent works in rather than the shell the post came from. `clawo forum post` now warns when the named agent has finished with no active loop, has not started, is not on the forum, or when the post carries no tag, and `clawo forum list` shows each member's current state — running, finished, or not started — at the top of the output. A finished goal automatically re-opens on the same thread when a steer or an unseen forum post targeting it arrives.

Add a shared forum to a workflow so all fan-out agents and goal nodes can post and read coordination messages.

```yaml
forum:
  push: "all"
```

Post a message to the forum from an agent, including its current score from a file, to share progress mid-workflow.

```
$ clawo forum post --title 'scan complete' --score-from
results/score.txt
```

Inspect the live state of every forum member before triaging a stalled workflow.

```
$ clawo forum list
```

— Names every forum config key, command and behavior change verbatim. v7.26.0 v7.28.0

02 **Contract fingerprinting and drift detection on resume**

Not checked yet NEW 95

Resuming a workflow with `clawo workflow resume <id> --contract <file>` now detects rule drift: if the contract's rules have changed since the run started, the resume records `contractDrift` — both fingerprints plus whether checks were added, removed, or changed — without applying the new rules. Acceptance-contract rules are identified by a `sha256` fingerprint over a canonical serialisation of each check's spec, required status, and test-protection flag, recorded on the run record at start, on the verdict, and in every evidence bundle including `verify_run` output.

Detect contract drift when resuming a run after the acceptance contract has been updated, without silently applying changed rules.

```
$ clawo workflow resume abc123 --contract
./contracts/acceptance.json
```

— Exact command, flag, fingerprint mechanism and storage locations given. v7.27.0

03 **Forum activity metrics** Not checked yet NEW 65

`workflow_forum` and `clawo forum metrics` report forum activity: posts by agent and tag, ASK response times, open ASKs, and post-to-agent delivery latency.

View forum metrics to see open ASKs, per-agent post counts, and how long posts took to reach named agents.

```
$ clawo forum metrics
```

— Named metrics and exact command, but limited mechanism detail. v7.26.0

04 **Mid-turn message delivery into running sessions** Not checked yet

NEW

62

`session_send_to` delivers a message directly into a busy Claude Code or `codex-app` session's running turn, picked up at the next tool round, allowing coordination without waiting for the agent to finish its turn.

— Named function and mechanism but no runnable invocation shown. v7.26.0

THINNER COVERAGE BELOW

05 **Split status reporting for workflow runs** Not checked yet IMPROVED

55

`workflow_status` and `clawo workflow show` now report check run execution, resumes, and verification as three separate aspects. `autoloop_status` lists the messages a paused run is holding.

— Names the commands but gives no example or output detail. v7.27.0

WAS THIS USEFUL? 👍 👎

cotal

Feedback ⓘ v0.171.5 NOTES ↗ RANK

Cotal is the open standard for agent coordination

> Cotal's biggest change this window is interactive agent command and skill invocation for OpenCode 2.x agents with queueing and owner-only authority, alongside a cluster of CLI diagnostics improvements across `cotal status`, `cotal up`, `cotal update`, and connector seed repair tooling.

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

01 **Interactive agent commands for OpenCode 2.x agents**

Not checked yet NEW 90

Owners can run agent commands and skills from an interactive session against OpenCode 2.x agents using `commands`, `command`, `command-cancel`, and `command-state` on the `agent` endpoint (cluster revision 2); up to 8 commands queue with no time bound, the last 32 outcomes are remembered, and the broker refuses all non-owner callers. Connectors dispatch queued commands with dispatchNextCommand() when `MeshAgent` emits the `command` event, using the new `openAgentEndpoint` owner API, which replaces `openPromptDesk`.

— Full endpoint, queue limits, and API names given v0.171.0

02 **Connector seed interruption detection and repair** Not checked yet

NEW 75

`cotal update --check` now reports whether any connector seeds were interrupted after an update and need repair. `cotal ext seed --repair` repairs an interrupted or corrupt connector extension manifest seed so seeding can proceed.

After an interrupted update, check whether any connector seeds need repair before proceeding.

```
$ cotal update --check
```

Repair an interrupted connector seed identified by `cotal update --check`.

```
$ cotal ext seed --repair
```

Repair a corrupt extension manifest so that connector seeding can proceed.

```
$ cotal ext seed --repair
```

— Runnable commands given as examples v0.171.0 v0.171.5

03 **Status, up and update command guidance** Not checked yet IMPROVED

70

`cotal status` now reports an answering manager outside a broker-only stack and points to `cotal who` for the live roster, and shows how to list spaces and select a different one. `cotal up` clears stale process records on reboot — emitting a `STALE_PROCESS_RECORD` warning for each — instead of refusing to start a controller stack, and `cotal up` / `cotal status` after renaming a stopped space's project folder now report the commands needed to keep the new location. `input --json` returns a structured refusal instead of an unstructured error when reporting an unregistered space. `cotal update` now accepts a stopped local stack's broker URL without restarting it, and `cotal update --self` gained new guidance text.

Resume a self-update or trigger an update check against a stopped local broker without bringing the stack back up.

```
$ cotal update
```

Check which space is active and get prompted on how to list or switch spaces.

```
$ cotal status
```

— Names several commands but each change is incremental v0.171.2 v0.171.4

04 **Connection warning parity between `cotal_wait` and `cotal_inbox`**

Not checked yet IMPROVED

65

An empty `cotal_wait` on a connection that cannot reach the broker now opens with the same connection warning `cotal_inbox` gives, rather than returning a plain No new messages reply.

— Named commands with clear before/after behavior v0.171.2

05 **Installation and daemon setup reliability** Not checked yet

IMPROVED

60

`pnpm dlx` runs under pnpm 11 (launched from the global virtual store) are now recognised, with `setup` and `status` suggesting `pnpm dlx cotal-ai` commands.

Windows daemon installation now gives the initial Task Scheduler lookup up to 60 seconds to finish before refusing.

— Named flags and timeout, but limited scope each `v0.171.0` `v0.171.2`

THINNER COVERAGE BELOW

06 **Interrupted turns return seats to idle** Not checked yet IMPROVED

55

Claude Code and Pi seats now return to `idle` after a user interrupts a turn, closing the run with an `interrupt` outcome instead of `RUN_ERROR`.

— Named outcomes but no further mechanism given `v0.171.2`

07 **`cotal run show --limit` scoped to text output** Not checked yet

IMPROVED 50

The `cotal run show` help now explains that `--limit` applies only to text output and cannot be combined with `--json`.

— Named flag with exact constraint clarified `v0.171.0`

08 **Hosted workflow validation diagnostics** Not checked yet IMPROVED

50

Hosted workflow starts now include syntax diagnostics in JSON refusals, and simulations reject invalid handoff destinations, nonnumeric clock starts, and duplicate script keys.

— Lists error cases without deeper mechanism `v0.171.0`

09 **Loopback dashboard link refresh without restart** Not checked yet

IMPROVED 50

Repeating a loopback dashboard start refreshes the private single-use link even without a display or with `--no-open`, while existing browser sessions stay signed in.

— Named flag and clear behavior, limited mechanism `v0.171.5`

10 **Codex setup configuration and plugin reporting** Not checked yet

IMPROVED 40

Codex setup now lists its configuration and plugin files in the JSON receipt, and status reports disabled plugins.

— Brief description without further mechanism `v0.171.0`

11 **Time-based `--since` cutoff for live history follow**

Not checked yet IMPROVED 40

Live history follow now honors time-based `--since` cutoffs.

— Named flag with concrete, actionable behavior v0.171.0

12 **Direct connection instructions for local MCP clients**

Not checked yet NEW 25

Cotal adds direct connection instructions for local MCP clients.

— Only mentioned in summary, no further detail v0.171.4

13 **List meshes without installing connectors** Not checked yet NEW 25

Users can list meshes and save preferences without installing connectors.

— Single line with no further detail v0.171.0

14 **Websocket mesh reconnection after silent drops** Not checked yet

IMPROVED 20

Cotal improves websocket mesh reconnection after silent connection drops.

— Only summary mention, no detail or mechanism v0.171.5

WAS THIS USEFUL? 👍 👎

mateclaw

Feedback ⓘ v3.1.0 NOTES ↗ RANK

MateClaw — Your second brain with Multi-Agent Orchestration, MCP Protocol, Skills & Memory, Dream, and Multi-Channel Support. Built on Spring AI Alibaba.

> MateClaw v3.1.0 adds per-source Wiki entity extraction and snapshot preview restoration, raises new employees' default iteration budget tenfold, and tightens team-dispatch budget enforcement, alongside new Azure/DeepSeek model support and multi-channel messaging reliability fixes.

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

01 **DeepSeek Harness 0.2.2-rc.1 candidate** Not checked yet NEW

75

Adds the optional DeepSeek Harness `0.2.2-rc.1` candidate, requiring Node.js 24.9+ (24.x), which restores file observation guards and binds configured model capacities.

— Exact version and Node requirement given, no run command v3.1.0

THINNER COVERAGE BELOW

02 **Default iteration budget raised to 100 for new employees**

Not checked yet IMPROVED

55

The default iteration budget for newly created employees rises from 10 to 100 (#672, #674), while existing employee settings are retained and capped runs are explained to users.

— Exact before/after numbers given, but no config key v3.1.0

03 **Cumulative team-dispatch budget enforcement across Lead retries**

Not checked yet IMPROVED

55

Team-dispatch budgets are now enforced cumulatively across Lead retries (#688), instead of granting a fresh full budget on every retry.

— Clear before/after mechanism described v3.1.0

04 **Media handling in oversized request copies** Not checked yet

IMPROVED

50

Oversized images are downscaled and empty media filtered in request copies; oversized requests can omit historical media while persisted original attachments remain intact.

— Explains behaviour and before/after but no named config v3.1.0

05 **Multi-channel messaging reliability improvements** Not checked yet

IMPROVED

50

Long DingTalk Markdown messages are now split and delivery acknowledgements validated, while readable Feishu attachment names are preserved.

— Names both channel fixes but lacks mechanism detail v3.1.0

06 **Per-source entity extraction in Wiki panel** Not checked yet NEW

45

Extract entities from a selected document directly in the Wiki raw-material panel via the new per-source entity extraction feature (#673).

— Names feature and location but no mechanism detail v3.1.0

07 **Azure unified OpenAI endpoint support** Not checked yet NEW

40

MateClaw now supports Azure unified OpenAI endpoints while preserving required endpoint query parameters.

— Names integration but no setup steps v3.1.0

08 **Managed deployment operational safeguards** Not checked yet NEW

40

Managed deployments gain offline-upgrade preflight backups, trial migrations, and search-health checks.

— Lists three additions without operational specifics v3.1.0

09 **Preview snapshot restoration** Not checked yet NEW

35

Inspect skill changes and their impact before confirming a snapshot restore, via the new preview snapshot restoration capability.

— Describes capability but no UI path or command v3.1.0

10 **Quote answer text into follow-up drafts** Not checked yet NEW

35

Users can quote selected answer text directly into follow-up drafts.

— Thin description of a small UX addition v3.1.0

WAS THIS USEFUL? 👍 👎

LangChain

Feedback ⓘ langchain-anthropic==1.7.12 NOTES ↗ RANK

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

> LangChain shipped two small reliability fixes this window: consistent finish-reason reporting in the Anthropic integration and metadata tagging for synthetic human-in-the-loop tool messages in langchain-core.

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

01 **Standardized finish reason in Anthropic streaming** Not checked yet

IMPROVED 40

The Anthropic integration now standardizes the finish reason on message-finish events, so the `finish_reason` field is consistent across streaming responses.

— Names the field but no example of use langchain-anthropic==1.7.12

02 **Source metadata on synthetic HITL tool messages** Not checked yet

IMPROVED 30

langchain-core now tags synthetic human-in-the-loop (HITL) tool messages with source metadata to identify their origin.

— Describes change but no mechanism or key names langchain-core==1.8.4

WAS THIS USEFUL? 👍 👎

Google ADK

Feedback ⓘ v2.15.0 NOTES ↗ RANK

The Agent Development Kit is Google's code-first Python toolkit for building, evaluating, and deploying agents, with built-in tools, sessions, and deployment to Vertex AI.

> Google Agent Development Kit v2.15.0 adds Google Maps-adjacent BigQuery table discovery, multi-run eval metrics with partial-credit tool matching, buffered session writes, a unified fsspec-based artifact service, per-user/session sandbox scoping, and up to 6x faster RAG memory search, while removing the Visual Agent Builder and tightening MCP instruction argument handling.

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

01 Multi-run eval metrics and tool parameter matching

Not checked yet NEW 90

`adk eval` adds `pass_at_k` / `pass@k` and `pass_hat_k` / `pass^k` metrics via `multi_run_config`, reporting their mean across the eval set, printing a per-case multi-run summary table by default, and saving `MultiRunCaseSummary` results to `.evalset_result.json`. It also adds `tool_parameter_match` (`ToolParameterMatchCriterion`), backed by `ToolArgsMatcher` and `IN_ORDER`, `EXACT`, or `ANY_ORDER` call pairing, scoring tool calls with partial credit per argument instead of failing the whole call on one mismatched parameter.

Run an eval suite that measures both best-case (`pass@k`) and consistent (`pass^k`) reliability across multiple runs.

```
$ adk eval \
    my_agent/ \
    my_agent.evalset.json \
    --config_file_path eval_config.json
```

— Full metric names, config, output files, and a runnable command v2.15.0

02 Cache-friendly dynamic instruction routing Not checked yet NEW

83

Dynamic instructions from `load_artifacts` and `load_mcp_resource` are now placed in a user-role message before the current turn to keep the static instruction prefix cacheable, with `DYNAMIC_INSTRUCTION_ROUTING` enabled by default; opt out by setting `ADK_DISABLE_DYNAMIC_INSTRUCTION_ROUTING=1`.

Opt out of the new default dynamic instruction routing if your setup depends on instructions being appended to `system_instruction`.

```
$ ADK_DISABLE_DYNAMIC_INSTRUCTION_ROUTING=1 adk run
path/to/my_agent
```

– Names the default behavior and exact opt-out env var with command v2.15.0

03 **Unified artifact storage via FsspecArtifactService**

Not checked yet IMPROVED 78

`FsspecArtifactService` now supports `save_artifact`, `delete_artifact`, `get_artifact_version`, and `list_artifact_versions` across local and cloud object stores (`gs://`, `s3://`, `file://`, `memory://`), including S3 custom metadata, enabling a single artifact service across local development and cloud deployments.

– Names all methods and supported URI schemes v2.15.0

04 **BigQuery table discovery via search_scope** Not checked yet NEW 73

`BigQueryToolset`'s `ask_data_insights` now accepts a `search_scope` parameter with `includeProjectIds` and `includeDatasetReferences`, letting agents discover relevant BigQuery tables across projects or datasets without knowing exact table references upfront.

– Names exact param and fields but no runnable example v2.15.0

05 **Explicit arguments required for McpInstructionProvider**

Not checked yet BREAKING 73

`McpInstructionProvider` no longer auto-fills prompt arguments from `session.state`; callers must pass McpInstructionProvider(..., arguments=...) (a mapping or a callable over `ReadonlyContext`) explicitly for any arguments the prompt expects.

– Exact migration syntax given, clear actionable fix v2.15.0

06 **Per-scope sandbox isolation via BaseEnvironment.scope**

Not checked yet NEW 71

`BaseEnvironment.scope` (`EnvironmentScope.SHARED`, `USER`, or `SESSION`) scopes initialization locks per user or session so sandboxes can be isolated without blocking concurrent sessions during startup; `BaseEnvironment` methods now also receive the calling run's `ctx: Context`.

– Named enum values and signature change, no walkthrough v2.15.0

07 **Background tool execution and live context updates in run_live**

Not checked yet IMPROVED 70

`run_live` now runs every tool call in a background task to keep voice audio, transcriptions, and barge-ins flowing while slow tools execute, and `BaseLlmConnection.send_context_update` updates system instructions or tools mid-session on open Gemini 3.x Live connections without reconnecting.

— Mechanism and API named, but no example invocation v2.15.0

08 **Buffered session writes via BufferedSessionService**

Not checked yet NEW 65

Experimental `BufferedSessionService` and `BaseSessionService.append_events` buffer session events in memory and flush them in background batches, keeping slow database or network writes off the critical path of agent turns.

— Mechanism and purpose given, no usage example v2.15.0

THINNER COVERAGE BELOW

09 **Visual Agent Builder removed from ADK Web** Not checked yet 55

BREAKING

The `/dev/apps/{app_name}/builder*` routes and `google.adk.cli.built_in_agents` are removed; the interactive Visual Agent Builder is no longer available in ADK Web.

— Names removed routes but no migration path given v2.15.0

10 **Faster RAG memory search in VertexAiRagMemoryService**

Not checked yet IMPROVED 40

`VertexAiRagMemoryService` and `VertexAiRagRetrieval` cut RAG memory search latency up to 6x by reusing API clients per event loop.

— Gives a concrete speedup number but no usage detail v2.15.0

└──▷ **BREAKING ON UPGRADE**

! The `/dev/apps/{app_name}/builder*` routes and `google.adk.cli.built_in_agents` are removed; the interactive Visual Agent Builder is no longer available in ADK Web.

! `McpInstructionProvider` no longer auto-fills prompt arguments from `session.state`; pass McpInstructionProvider(..., arguments=...) (a mapping or a callable over `ReadonlyContext`) explicitly for any arguments the prompt expects.

WAS THIS USEFUL?  

AWS Strands Agents

Feedback ⓘ harness-typescript/v0.5.0

NOTES ↗ RANK

Strands Agents is AWS's open-source SDK for production agents in Python and TypeScript, with a model-driven loop, tool integration, and deployment targets.

> Strands Agents shipped a new MCP tool for discovering integrations, an auto-decision mode for human-in-the-loop control, and expanded model-routing options, alongside several breaking changes to default Bedrock model selection, history structure, offload behavior, and built-in plugin loading.

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

01 **Bedrock model info and default model change** Not checked yet

NEW INTEGRATION UPDATED SUPPORTED PLATFORM BREAKING 75

Adds `defaultEffort` to the shared model info, handles Bedrock `InvokeModel` mid-stream errors and API version, and changes the Anthropic Bedrock client's default model to Claude Opus 5.5, affecting both Python and TypeScript SDKs.

– Names the field, error handling, and default model change across both SDKs. harness-cli/v0.4.0

THINNER COVERAGE BELOW

02 **find_integrations MCP tool** Not checked yet NEW 55

Adds a `find_integrations` MCP tool backed by the integrations catalog feed, letting agents discover available integrations.

– Names the tool and its data source but no usage example. harness-cli/v0.4.0

03 **Auto mode for HumanInTheLoop** Not checked yet NEW 55

Adds an `auto` mode to `HumanInTheLoop` for automatic human-in-the-loop intervention decisions, removing the need for manual decision handling in that path.

– Names the mode and component but omits configuration detail. harness-cli/v0.4.0

04 **ClassifierStrategy routing with any DecisionModel** Not checked yet

NEW 55

`ClassifierStrategy` can now route with any `DecisionModel` in Python, widening which models can drive classification-based routing.

– Names both classes but gives no usage example. harness-cli/v0.4.0

05 **Injected text stored as history block** Not checked yet

BREAKING

50

Injected text is now written into history as an injected block in Python, changing the history structure for agents that use text injection.

— States the structural change but not the exact migration steps. harness-cli/v0.4.0

06 **todos plugin removed from default harness plugins** Not checked yet

50

BREAKING

The `todos` plugin is no longer included in the default built-in harness plugins; it must now be loaded from the SDK explicitly.

— Names the plugin and the required action, but no code snippet. harness-cli/v0.4.0

07 **Offload condition behavior change** Not checked yet BREAKING

30

The offload conditions that decide context granularity have changed behavior in both Python and TypeScript, though the exact new behavior is not detailed.

— Flags a behavior change without specifying what changed. harness-cli/v0.4.0

NEW INTEGRATION strands-acgs catalog entry

NEW INTEGRATION SageMakerDecisionModel for System One models on SageMaker AI endpoints

└──▷ BREAKING ON UPGRADE

! The Anthropic Bedrock client now defaults to Claude Opus 5.5 (changed from the previous default model).

! Injected text is now written into history as an injected block in Python, changing the history structure for agents using text injection.

! The offload conditions that decide context granularity have changed behavior in both Python and TypeScript.

! The `todos` plugin is no longer included in the default built-in harness plugins; it must now be loaded from the SDK explicitly.

harness-typescript/v0.5.0 NOTES ↗ STABLE

Strands Agents adds a Dogwood authorization handler, async multi-agent builders, `ModelInfo.merge`, configurable background-task tool names, `auto_reconnect` for MCP, and OpenAPI metadata publishing for agents.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch harness-typescript/v0.5.0
https://github.com/strands-agents/harness-sdk.git
# already have the repo? check out this version:
$ git checkout harness-typescript/v0.5.0
```

Add a Dogwood authorization handler for interventions in TypeScript.
Not checked yet

Add `ModelInfo.merge` to TypeScript to combine model info from multiple sources.
Not checked yet

Allow configuring the management tool name for background tasks. Not checked yet

Allow sorting tool specs in tool configurations. Not checked yet

Add a reminder interval to the todos plugin. Not checked yet

+6 more

└─▷ **BREAKING ON UPGRADE**

! The Anthropic Bedrock client now defaults to Claude Opus 5.5; agents that relied on the previous default model will switch models on upgrade.

! Context offload conditions now decide granularity differently (Python); agents using context offload may see changed behaviour on upgrade.

WAS THIS USEFUL? 👍 👎

ag2 Feedback ⓘ v1.3.3 NOTES ↗ RANK

AG2 (formerly AutoGen): The Open-Source AgentOS.Join us at: https://discord.gg/sNGSwQME3x

> ag2 adds google-genai 3.x support to its Gemini integration, introducing version control for the Gemini API, new transport options, and per-utterance Live transcription, while flagging the move to require google-genai 3.x in the next release.

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

01 Gemini API version control via api_version

Not checked yet

88

BREAKING

A new `api_version` field on `GeminiConfig` and `VertexAIConfig` lets you pin which Gemini API version ag2 targets — `'v1'` or `'v1beta'` for the Gemini Developer API, and `'v1'` or `'v1beta1'` for Vertex AI — applied to both generation and the Files API. This matters because upgrading to google-genai 3.x switches the default from `v1beta` to `v1`, which breaks models only available on `v1beta` (such as `-latest` aliases and some `-preview` models) unless you set `api_version='v1beta'` or pin `google-genai<3`.

Use a `-latest` or `-preview` Gemini model that is only available on `v1beta` after upgrading to google-genai 3.x.

```python
from ag2.config import GeminiConfig

config = GeminiConfig(model="gemini-flash-latest",
api_version="v1beta")
```

— Names exact field, allowed values, and migration fix v1.3.3

02 google-genai 3.x dependency and deprecation warning

Not checked yet DEPRECATED

70

The dependency constraint is now `google-genai>=1.68.0,<4`, installable via `pip install 'ag2[gemini]'`. `GeminiConfig` and `VertexAIConfig` now emit a `DeprecationWarning` on create() and create_files_client() when google-genai 2.x is installed, since AG2 1.4.0 will require `google-genai>=3,<4`.

Opt into google-genai 3.x for Gemini support in a fresh environment.

```
$ pip install 'ag2[gemini]'
```

— Names exact constraint, install command, and future requirement v1.3.3

03 **Per-utterance transcription in Gemini Live** Not checked yet

IMPROVED

65

Gemini Live sessions now emit `TranscriptionCompletedEvent` for user speech when InputConfig(transcribe=True) is set. With google-genai 3.x models this fires once per utterance as it arrives, whereas earlier models still join fragments and emit a single event at turn end.

— Explains mechanism and behaviour difference, no code sample v1.3.3

04 **httpx2 transport support for Gemini configs** Not checked yet NEW

62

You can now pass an `httpx2.AsyncClient` to the `http_client` field on `GeminiConfig` or `VertexAIConfig`, reflecting that google-genai 3.x defaults to the `httpx2` transport.

— Names field and class but no usage example v1.3.3

└──▷ **BREAKING ON UPGRADE**

! If your environment upgrades to `google-genai` 3.x, Gemini Developer API calls go to `v1` instead of `v1beta`. Models available only on `v1beta` (such as `-latest` aliases and some `-preview` models) will fail — set `api_version='v1beta'` on `GeminiConfig` or pin `google-genai<3` to restore the previous behaviour.

! `GeminiConfig` and `VertexAIConfig` now emit a `DeprecationWarning` on create() and create_files_client() when `google-genai` 2.x is installed. AG2 1.4.0 will require `google-genai>=3,<4`.

WAS THIS USEFUL? 👍 👎

AutoGPT

Feedback ⓘ autogpt-platform-beta-v0.12.0 NOTES ↗ RANK

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

> AutoGPT Platform's beta v0.12.0 expands Otto with five new tool integrations and automatic MCP server discovery, adds four new AI models, previews a native mobile app, and tightens sign-in security while replacing proxy-trust configuration.

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

01 Replace TRUST_X_FORWARDED_FOR with TRUSTED_PROXIES

Not checked yet

95

BREAKING

The frontend no longer reads `TRUST_X_FORWARDED_FOR`; it must be replaced with `TRUSTED_PROXIES` set to a comma-separated list of proxy addresses or CIDR ranges, or the frontend will log a warning and lose client-IP data for sign-in logs, IP blocks, and per-visitor store search. `FORWARDED_ALLOW_IPS` and `TRUSTED_PROXY_COUNT` are deprecated in favour of `TRUSTED_PROXIES`; they continue to work in this release but will be removed in a future version.

Migrate a self-hosted reverse-proxy setup from the old `TRUST_X_FORWARDED_FOR` variable to the new `TRUSTED_PROXIES` variable so the frontend, backend, and External API all share one consistent client-IP source.

```env
TRUSTED_PROXIES=10.0.0.1,192.168.1.0/24
```

— Exact env vars, migration example, and consequence of not migrating

autogpt-platform-beta-v0.12.0

THINNER COVERAGE BELOW

02 Otto tool integrations and MCP discovery

Not checked yet NEW

55

Otto can now connect to `AnySearch`, `Supabase`, `Klaviyo`, `AI/ML API`, and `CompanyEnrich` as new tool integrations. Otto also discovers custom MCP servers connected by URL without manual configuration, and Supabase reads through Otto no longer require per-request user approval.

— Names all five integrations and two behaviour changes, no usage steps

autogpt-platform-beta-v0.12.0

03 **Four new models added to workflows and Otto Codex tier**

Not checked yet NEW 55

Adds four new models available in workflows and Otto's Codex tier: Gemini 4 Neon, `GPT-6.2 Astra`, `GPT-6.2 Sol`, and Qwen 4 Max — with Otto's advanced Codex tier now running on `GPT-6.2 Astra`.

— Names all four models and the Codex tier default, no further mechanism

autogpt-platform-beta-v0.12.0

---

04 **Extra verification step for new sign-in methods** Not checked yet

IMPROVED 55

Adding a new sign-in method (Google, Microsoft, GitHub, or Discord) now requires a six-digit code from the user's inbox, with an email notification sent each time a new method is added.

— Explains the new verification flow and notification, no config surface

autogpt-platform-beta-v0.12.0

---

05 **ClamAV upload scanning changes** Not checked yet IMPROVED

50

ClamAV upload scanning now refuses files it cannot scan whole rather than scanning in pieces, and the bundled ClamAV's file-size limit rises to 256 MB.

— Clear before/after behaviour and a concrete limit, no action for users

autogpt-platform-beta-v0.12.0

---

06 **Native mobile app preview** Not checked yet NEW 40

A native mobile app (work in progress) lets users sign in, chat with Otto, hire experts from the Marketplace, connect services from chat, and use live voice dictation.

— Describes scope of preview app but no download or access path

autogpt-platform-beta-v0.12.0

---

07 **Admin requeue for stuck executions** Not checked yet IMPROVED

40

Admin requeue now publishes stuck `QUEUED` executions to unblock stalled runs.

— Names the mechanism briefly but lacks UI or command path autogpt-platform-beta-v0.12.0

└──▷ **BREAKING ON UPGRADE**

! The frontend no longer reads `TRUST_X_FORWARDED_FOR`; replace it with `TRUSTED_PROXIES` set to a comma-separated list of proxy addresses or CIDR ranges, or the frontend will log a warning and lose client-IP data for sign-in logs, IP blocks, and per-visitor store search.

! `FORWARDED_ALLOW_IPS` and `TRUSTED_PROXY_COUNT` are deprecated in favour of `TRUSTED_PROXIES`; they continue to work in this release but will be removed in a future

version.

WAS THIS USEFUL? 👍 👎

CrewAI

Feedback ⓘ 1.15.44 NOTES ↗ RANK

CrewAI is an open-source Python framework for building multi-agent workflows using role-based AI agent Crews and event-driven Flows.

> CrewAI's lone release in this window added versioned, shareable artifact saving for flow outputs.

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

01 **Versioned artifacts for flow outputs** Not checked yet NEW

33

Flow outputs can now be saved as versioned, shareable artifacts, allowing results to be stored and retrieved across runs.

— Describes capability in prose with no API, flag, or command detail. 1.15.44

WAS THIS USEFUL? 👍 👎

CopilotKit OpenBot

Feedback ⓘ v1.1.1 NOTES ↗ RANK

OpenBot runs self-hosted AI coworkers that each get their own browser, files, and gated tools, with every action decided and recorded.

> OpenBot shipped Kubernetes Helm charts for production deployment alongside a new OpenAPI/AsyncAPI spec and a formal API versioning policy with concrete client-version enforcement, plus object-store file attachments, a LangGraph TypeScript bot engine with dedicated Claude/ChatGPT plan harnesses, Office file support in conversations, and per-user sandbox isolation for computers.

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

01 Object-store file attachments

Not checked yet BREAKING

98

Store file attachments in Amazon S3, Google Cloud Storage, or Azure Blob Storage — up to 100 MB per file — by setting `attachments.store` and `objectStorage.enabled`, `objectStorage.provider`, `objectStorage.bucket`, `objectStorage.prefix`, `objectStorage.region`, `objectStorage.endpoint`, `objectStorage.forcePathStyle`, `objectStorage.azure.accountUrl`, `objectStorage.deleteAfterDays`, `objectStorage.assumeAllServersCurrent`, `objectStorage.existingSecret`, and `objectStorage.inCluster` in the `openbot` Helm chart, or `FILES_STORE`, `FILES_PREFIX`, `FILES_DELETE_AFTER_DAYS`, `FILES_ASSUME_ALL_SERVERS_CURRENT`, `S3_*`, `GCS_BUCKET`, and `AZURE_STORAGE_*` in `.env` for Docker Compose. Alternatively keep attachments in PostgreSQL on `openbot-bundle` by setting `openbot.attachments.store: postgres` and `openbot.objectStorage.enabled: false` — the bundle now defaults to moving files into the `openbot-files` bucket in bundled object storage, and rolling back to 1.1.0 after migration requires setting `attachments.store: postgres` and waiting for files to migrate back, since a 1.1.0 server cannot read object-store files.

Point a production deployment at S3 for file attachments so users and Bots can exchange files up to 100 MB.

```yaml
attachments:
  store: s3

objectStorage:
  enabled: true
  provider: s3
  bucket: my-openbot-files
  prefix: openbot/
  region: us-east-1
  deleteAfterDays: 7
  existingSecret: openbot-s3-secret
  secretKeys:
    accessKeyId: AWS_ACCESS_KEY_ID
    secretAccessKey: AWS_SECRET_ACCESS_KEY
```

— Exhaustive config keys, runnable example, and migration caveat v1.1.1

02 **LangGraph TypeScript engine and plan agent harnesses**

Not checked yet

93

BREAKING

Run new Bots on the LangGraph TypeScript engine by configuring `agent-langgraph-ts` (port 4215) in Docker Compose and setting `LANGGRAPH_TS_AG_UI_URL` to point at it, or using `LANGGRAPH_TS_PORT` to move the port in the personal edition; keep all new Bots on the built-in engine with `OPENBOT_DEFAULT_BOT_ENGINE=built-in`. Route Claude and ChatGPT plan agents through dedicated Compose services `agent-claude-plan` and `agent-chatgpt-plan` via the `claude-plan` and `chatgpt-plan` profiles, setting `CLAUDE_PLAN_HARNESS_URL` or `CHATGPT_PLAN_HARNESS_URL` in a Compose override. New Bots now default to the TypeScript engine where available, and Bots previously on the Python LangGraph engine move to TypeScript at their next Install and Start; `agent-harness` no longer receives `CLAUDE_CODE_OAUTH_TOKEN`, `CHATGPT_AUTH_FILE`, or the `./.langchain` folder, so overrides running plans through it must switch to the new profiles and URLs.

Opt out of the LangGraph TypeScript engine so all new Bots stay on the server built-in engine across every environment.

```ini
OPENBOT_DEFAULT_BOT_ENGINE=built-in
LANGGRAPH_TS_AG_UI_URL=
```

— Named services, ports, env vars, and migration behavior given v1.1.1

03 **API versioning policy and app-version enforcement** Not checked yet

NEW

90

A new API versioning policy defines what counts as a breaking change, deprecation timelines, a 12-month support window, minimum app versions, the launcher contract, the security floor, and CI gates. It is enforced with `OPENBOT_MINIMUM_CLIENTS` (for example `desktop=1.2.0,mobile=1.2.0`), which rejects apps older than the threshold with a 400 `CLIENT_UPDATE_REQUIRED` response; apps now send `OpenBot-Client: <app>/<version>` on every request, and the `deployment` endpoint returns `api`, `minimumClients`, `launcherApi`, and `extensionApi` fields naming supported versions. Deprecated API operations, parameters, and request-body fields now respond with `Deprecation`, `Sunset`, and `Link` headers.

— Full policy plus concrete enforcement config and response headers `v1.1.1`

---

04 **Helm charts for Kubernetes deployment** `Not checked yet` NEW

85

Deploy OpenBot with the `openbot-bundle` Helm chart (OpenBot and Intelligence together on one PostgreSQL server) or the `openbot` Helm chart (OpenBot alone, pointed at an existing Intelligence instance), with each chart's guide available via `helm show readme`. Promote a release through the `openbot` chart using `bun scripts/update.ts kubernetes`; the chart's database wait now runs from the server's own image, so using the new chart with an older `image.tag` will not start.

Deploy OpenBot and Intelligence together on a single PostgreSQL server using the bundled Helm chart.

```
$ helm show readme openbot-bundle
```

Promote a new OpenBot release through the Kubernetes Helm chart in your repository.

```
$ bun scripts/update.ts kubernetes
```

— Names both charts, promotion command, and image-tag caveat `v1.1.1`

---

05 **Sandbox-per-user computer isolation** `Not checked yet` BREAKING

80

Use sandbox-per-user isolation by setting `computers.mode: sandbox` in Helm values. Deploying with shared or external computer modes — or no mode set — now requires acknowledging the risk with `computers.sharedModeAcknowledged: true` (or `openbot.computers.sharedModeAcknowledged` in `openbot-bundle`) before `helm install` or `helm upgrade` will proceed.

Gate upgrades to sandbox mode and acknowledge shared-computer risk in a Helm values file so `helm upgrade` does not block.

```yaml
computers:
  mode: shared
  sharedModeAcknowledged: true
```

— Named config keys with a runnable values-file example v1.1.1

06 **Chat Completions override for newer models** Not checked yet

70

BREAKING

Keep model gateways on Chat Completions for `gpt-5.4`, `gpt-5.5`, and their mini, nano, and dated variants by setting `COPILOTKIT_OPENAI_API=chat`; without it those models are now called on the Responses API (`/responses`) by default.

— Exact env var and affected models, no sample config file v1.1.1

THINNER COVERAGE BELOW

07 **Office file attachments in conversations** Not checked yet NEW

55

Attach and send `.docx`, `.xlsx`, and `.pptx` files in conversations, with Bots reading up to 120,000 characters of text content (sheets as tables, document tables, slides in order). Bots can now also hand files from their computer to users, displayed as a downloadable card in the conversation, task, or group view.

— Concrete formats and limit but no config key or endpoint v1.1.1

08 **Configurable graceful pod shutdown** Not checked yet IMPROVED

55

Configure graceful pod shutdown duration with `server.shutdown` on Kubernetes, up to 45 seconds. Misconfigured `SHUTDOWN_KEEP_SERVING_SECONDS`, `SHUTDOWN_DRAIN_SECONDS`, or `RELEASE_STATUS_PORT` now stop the server from starting at all, and a server that refuses to start exits with code 78 instead of 1, so exit-code monitoring must be updated.

— Named config key and limit given, no example file v1.1.1

09 **OpenAPI REST spec and AsyncAPI live channels** Not checked yet

NEW

45

New REST API spec in OpenAPI and a live-channels spec in AsyncAPI, with a generated client, documented at the API spec page.

— Names formats and client but gives no path or usage product docs

└──▷ **BREAKING ON UPGRADE**

! Helm charts deploying with `computers.mode: shared`, `computers.mode: external`, or no `computers.mode` set must add `computers.sharedModeAcknowledged: true` (or `openbot.computers.sharedModeAcknowledged: true` in `openbot-bundle`) before `helm install` or `helm upgrade` will proceed.

! The `openbot-bundle` chart now moves file attachments out of PostgreSQL and into the `openbot-files` bucket in the bundled object storage by default; set `openbot.attachments.store: postgres` and `openbot.objectStorage.enabled: false` to keep files in PostgreSQL.

! Rolling back to 1.1.0 after files have moved to an object store requires setting `attachments.store: postgres` and waiting for files to migrate back; a 1.1.0 server cannot read files stored in the object store.

! New Bots now run on the LangGraph TypeScript engine by default where it is available; Bots previously on the Python LangGraph engine move to TypeScript at their next Install and Start.

! `agent-harness` no longer receives `CLAUDE_CODE_OAUTH_TOKEN`, `CHATGPT_AUTH_FILE`, or the `./.langchain` folder; Compose overrides that ran plans through `agent-harness` must enable the `claude-plan` or `chatgpt-plan` profile and set `CLAUDE_PLAN_HARNESS_URL` or `CHATGPT_PLAN_HARNESS_URL`.

! A model gateway serving `gpt-5.4`, `gpt-5.5`, or their mini, nano, or dated versions must now serve `/responses`; set `COPILOTKIT_OPENAI_API=chat` to keep them on Chat Completions.

! A server that refuses to start now exits with code 78 instead of 1; monitoring that watches the exit code must be updated.

! Settings `OPENBOT_MINIMUM_CLIENTS`, `OPENBOT_DEFAULT_BOT_ENGINE`, `SHUTDOWN_KEEP_SERVING_SECONDS`, `SHUTDOWN_DRAIN_SECONDS`, `RELEASE_STATUS_PORT`, and file-store settings now stop the server from starting when misconfigured; an `S3_BUCKET`, `GCS_BUCKET`, or `AZURE_STORAGE_CONTAINER` already in the environment configures a file store even without `FILES_STORE` set.

! The `openbot` chart's database wait now runs from the server's own image, so using the new chart with an older `image.tag` will not start.

WAS THIS USEFUL? 👍 👎

◆ Agent Memory

EverMind AI EverOS

Feedback ⓘ v1.7.1 NOTES ↗ RANK

EverOS provides a local-first memory layer that stores and evolves context for AI agents.

> EverOS v1.7.1 adds per-device multi-speaker memory and language-aware episode writing, overhauls episode and agent search ranking, and removes the foresight feature along with several breaking pipeline-event and Python API changes.

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

01 Foresight feature removed

Not checked yet BREAKING 93

On first start of v1.7.1, EverOS deletes all foresight data — the `foresight` index table, its collections, and every `.foresights/` folder — from stores that previously ran foresight; copy `users/*/.foresights/` out before upgrading if you need to keep it. The `extract_foresight` strategy in `ome.toml` is now ignored (logs `config_override_unknown_strategy`) and should be removed, and `[llm.foresight]` plus `EVEROS_LLM__FORESIGHT__*` environment variables are now ignored (logs `llm.task.retired` on startup) and should also be removed.

— Exact removed paths, config keys, and env vars with migration step. v1.7.1

02 Pipeline events and Python API signatures changed

Not checked yet 90

BREAKING

The `UserPipelineStarted` event is removed; strategies must subscribe to `EpisodeExtracted` instead (one per memcell, carrying `memcell_id`). `EpisodeExtracted` no longer carries `owner_id`, `episode_entry_id`, or `owner_name` at the top level — read these from `copies` (one entry per owner) or subscribe to the new `EpisodeExtractedForOwner` event. In the Python API, `UserMemoryPipeline` no longer accepts `summary_llm` and now requires `language_llm` or `profile_store`; `EpisodeStore` gains a required new method `replace_summary`; and `ReflectionOrchestrator` now requires `language_llm` or `profile_store`. The `locomo` and `locomo-step` benchmark configs now also disable `split_episode_by_owner`, so a store ingested with an earlier config cannot be reused.

— Every renamed event, field, and required Python method named. v1.7.1

03 Per-device multi-speaker memory via device/add API

Not checked yet NEW 88

The new `POST /memory/device/add` endpoint lets one hardware device keep separate memory for each recognized speaker, filing each person's episodes under a `<device_id>@<slot>` `user_id` on existing routes; speakers marked `consented: false` are silently dropped.

Register utterances from a shared meeting-room device so each identified speaker gets their own private memory, with unidentified audio filed under the device itself.

```
$ curl -X POST https://everos.example.com/memory/device/add \
  -H 'Content-Type: application/json' \
  -d '{"device_id": "room-42", "utterances": [{"slot": "A",
"display_name": "Alice", "consented": true, "text": "Let us
review the roadmap."}, {"slot": "B", "consented": false, "text":
"I would rather not."}, {"slot": "A", "text": "OK, next item
then."}]}'
```

— Full endpoint, field names, and ID scheme given with a runnable example. v1.7.1

---

04 **Episode search quality improvements across hybrid, agentic, vector modes**

Not checked yet IMPROVED 86

`hybrid` search now ranks three recall lanes — BM25, episode vector, and an atomic-fact lane (200 facts max-pooled onto at most 50 episodes) — fused by two RRF passes, with `top_k` now counting episodes rather than episodes and facts combined (a breaking change to result sizing). `agentic` search now reads at most 500 atomic facts per MaxSim lane and recalls each episode by its own body vector and BM25 alongside its facts, replacing the un-scoped hybrid fallback for scopes over 1,000 episodes. `vector` search now reads 200 atomic facts regardless of `top_k` and returns at most 50 episodes when `top_k` is `-1`.

— Mechanism and numeric limits given for each mode, no direct usage example. v1.7.1

---

05 **Language-aware episode writing via llm.language task**

Not checked yet NEW 82

A new `[llm.language]` LLM task configuration controls a language-detection step that runs before each memcell's episode, summary, atomic facts, and profile are written. It defaults to temperature 0, 64 output tokens, a 3-second timeout, and no retries, and can be pointed at any model, as in `model = "gpt-4o-mini"`.

Point the language-detection step at a fast, non-thinking model and disable offline summaries entirely to minimise latency per memcell.

```toml
[llm.language]
model = "gpt-4o-mini"
temperature = 0
max_tokens = 64
timeout = 3
retries = 0

[ome]
strategies = ["split_episode_by_owner"]  # summarize_episode
omitted to keep placeholder summaries
```

— Names the config block, defaults, and ordering, with a config example. v1.7.1

06 **New OME strategies: summarize_episode and split_episode_by_owner**

Not checked yet NEW 82

Two new strategies in `ome.toml` control episode handling: `summarize_episode` toggles offline episode summary writing, with episodes lacking a completed summary returning a placeholder made of the episode's opening truncated to 120 character-widths and closed with `…`; `split_episode_by_owner` controls per-owner episode splitting by dispatching an `EpisodeExtractedForOwner` event per owner with no extra model call.

Point the language-detection step at a fast, non-thinking model and disable offline summaries entirely to minimise latency per memcell.


```toml
[llm.language]
model = "gpt-4o-mini"
temperature = 0
max_tokens = 64
timeout = 3
retries = 0

[ome]
strategies = ["split_episode_by_owner"]  # summarize_episode
omitted to keep placeholder summaries
```


— Both strategy names and exact placeholder/event mechanics given. v1.7.1

THINNER COVERAGE BELOW

07 **Wider candidate pool for agent case and skill hybrid search**

Not checked yet IMPROVED 57

Agent case and skill `hybrid` search now recalls `top_k * 2` candidates per lane (50 when `top_k` is `-1`), giving the reranker a wider pool from which to pick its 10 results.

— Exact multiplier and defaults given but no config surface to act on. v1.7.1

---

08 **Atomic fact extraction deduplicated per memcell** Not checked yet

IMPROVED 54

Atomic facts are now extracted once per memcell and written under every owner, so a memcell with several user senders no longer triggers one model call per sender; clustering embedding likewise now runs once for all owners instead of per owner.

— Clear before/after behaviour but no flag or endpoint to act on. v1.7.1

---

09 **Reflection merges episodes in profile language** Not checked yet

IMPROVED 48

Reflection now merges episodes in the owner's profile language, falling back to a majority vote (greater than 60% of stored languages) when no profile language is set.

— Behaviour and threshold given, no direct config or API to invoke. v1.7.1

└──▷ BREAKING ON UPGRADE

! `hybrid` search's `top_k` now counts episodes (each carrying at most one atomic fact) instead of episodes and facts together — queries relying on the old combined count will return different result sizes.

! Foresight data (`foresight` index table, collections, and every `.foresights/` folder) is removed on first start of v1.7.1 on a store that previously ran foresight — copy `users/*/.foresights/` out before upgrading if you need to keep it.

! The `extract_foresight` strategy in `ome.toml` is now ignored (logs `config_override_unknown_strategy`); remove the block.

! `[llm.foresight]` and `EVEROS_LLM__FORESIGHT__*` environment variables are now ignored (logs `llm.task.retired` on startup); remove them.

! The `UserPipelineStarted` event is removed; strategies that subscribed to it must subscribe to `EpisodeExtracted` instead (one per memcell, carrying `memcell_id`).

! `EpisodeExtracted` no longer carries `owner_id`, `episode_entry_id`, or `owner_name` at the top level; read these from `copies` (one entry per owner) or subscribe to `EpisodeExtractedForOwner`.

! The Python API for `UserMemoryPipeline` no longer accepts `summary_llm`; it now requires `language_llm` or `profile_store`.

! The Python API for `EpisodeStore` gains a required new method `replace_summary`.

! The Python API for `ReflectionOrchestrator` now requires `language_llm` or `profile_store`.

! The `locomo` and `locomo-step` benchmark configs now also disable `split_episode_by_owner`; a store ingested with an earlier config is not reused.

WAS THIS USEFUL?  

◆ MCP Servers & Tooling

pullmd

Feedback ⓘ v3.20.0 NOTES ↗ RANK

Pullmd is a self-hosted URL- and file-to-Markdown service for humans and AI agents - web pages, documents, images, audio, YouTube. PWA + REST + MCP + Claude Code skill, Reddit-aware, refreshable share links.

> pullmd 3.20.0 adds configurable conversion worker limits and document complexity budgets that return 422 on oversized or runaway content, an OAuth connected-apps manager in settings, and a round of API hardening including admin-gated stats endpoints, uniform JSON error codes, and frame-embedding protection.

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

01 Conversion worker pool limits and timeouts

Not checked yet NEW

90

Conversion parallelism and timeouts are now controlled with `PULLMD_CONVERT_WORKERS` (default `2`) and `PULLMD_CONVERT_TIMEOUT_MS` (default `60000`); a conversion that times out or exhausts the 512 MB worker heap answers `422` with `"code": "conversion_timeout"` or `"code": "document_too_complex"`. Setting `PULLMD_CONVERT_WORKERS=0` converts in the server process without limits, documented as safe only for private instances.

Cap the conversion worker pool to 4 and raise the per-conversion timeout to 2 minutes to handle heavier documents without blocking the server process.

ini

```
PULLMD_CONVERT_WORKERS=4
PULLMD_CONVERT_TIMEOUT_MS=120000
```

— Names exact env vars, defaults, heap limit, and error codes v3.20.0

02 Admin API key required for stats and storage endpoints

Not checked yet

74

BREAKING

Monitors reading `/api/stats` or `/api/storage` now need an admin API key when sign-in is on or `DISABLE_PUBLIC_HISTORY` is set; unauthenticated callers still get `200` but only receive `window` or `retentionDays`. Monitors previously reading `total` or `requests7d` now receive empty responses unless an admin API key is presented.

— Names both endpoints, config flag, and exact before/after fields v3.20.0

03 **Document complexity budgets for HTML/XML/OpenDocument**

Not checked yet NEW 71

Over-nested or oversized HTML, XML, and OpenDocument files are now refused with `422` and `"code": "document_too_complex"` before parsing begins. Limits for HTML are 100,000 elements, 200,000 text nodes, and 256 nesting levels.

— Concrete numeric limits and error code, no config knob given v3.20.0

---

04 **CACHE_DB must live under /data** Not checked yet BREAKING 69

A `CACHE_DB` path pointing into `/app` stops working because the app user can no longer write to `/app`; operators must keep the database under `/data` (e.g. `CACHE_DB=/data/cache.db`).

— Exact config key and migration path given v3.20.0

---

05 **Uniform JSON API error responses** Not checked yet IMPROVED 68

Every API error response is now JSON with `{"error": ..., "code": ...}` — including `500` (`internal_error`), `413` (`payload_too_large`), `400` (`invalid_body`), and other `4xx` (`bad_request`) — with no stack trace exposed.

— Lists exact status codes and error codes, no config action v3.20.0

THINNER COVERAGE BELOW

06 **pullmd-playwright version must match pullmd** Not checked yet 59

BREAKING

A `pullmd-playwright` image older than 3.20.0 causes `/api/status` to report `degraded` with `policy-missing` and return HTTP `503` — `pullmd-playwright` must be updated together with `pullmd`.

— Names endpoint, status codes, and upgrade requirement v3.20.0

---

07 **OAuth connected-apps management** Not checked yet NEW 57

A new Connected apps section on `/settings` lists OAuth clients holding a valid refresh token, showing connection and last-used timestamps with a per-client Revoke button.

— Names UI location but no API surface given v3.20.0

---

08 **Cross-origin iframe embedding blocked** Not checked yet BREAKING 50

Embedding pullmd in a cross-origin frame is now blocked by the new `X-Frame-Options: SAMEORIGIN` and `Content-Security-Policy: frame-ancestors 'self'` response headers.

— Names exact headers but no remediation path given `v3.20.0`

└──▷ BREAKING ON UPGRADE

! A `CACHE_DB` path pointing into `/app` stops working; the app user can no longer write to `/app` — keep the database under `/data` (e.g. `CACHE_DB=/data/cache.db`).

! Monitors reading `total` or `requests7d` from `/api/stats` or `/api/storage` now receive empty responses unless the caller presents an admin API key when sign-in is on or `DISABLE_PUBLIC_HISTORY` is set.

! A `pullmd-playwright` image older than 3.20.0 causes `/api/status` to report `degraded` with `policy-missing` and return HTTP `503` — update `pullmd-playwright` together with `pullmd`.

! Embedding pullmd in a cross-origin frame is now blocked by the new `X-Frame-Options: SAMEORIGIN` and `Content-Security-Policy: frame-ancestors 'self'` headers.

WAS THIS USEFUL? 👍 👎

morluto REA

Feedback ⓘ rea-agents-6.10.0 NOTES ↗ RANK

REA is a reverse engineer anything with agents, from app behavior down to native binaries.

> REA 6.10.0 adds offline UEFI firmware inspection, broad new format support (Android/OTA, NES 2.0, package metadata, CPython bytecode), an Evidence schema revision, and an upgraded verified Ghidra 12.2 lane with more ELF targets.

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

01 **MCP output schema compact profile** Not checked yet NEW

80

Setting the `REA_MCP_OUTPUT_SCHEMA_PROFILE=compact` environment variable advertises each MCP output schema within a 4,096-byte budget.

— Exact env var and byte limit given, directly usable rea-agents-6.10.0

02 **Setup agent registration and conflict detection** Not checked yet

IMPROVED

65

Setup now manages fx, Junie, Goose, and Kilo registrations, supports explicit MCP catalog profile selection, and flags a shared `enable: false` as a manual conflict.

— Names specific agents and config flag, modest mechanism detail rea-agents-6.10.0

03 **Ghidra 12.2 lane with DOS COM and new ELF targets** Not checked yet

IMPROVED

60

The verified Ghidra 12.2 lane now opens recognized DOS COM files and supports more RISC-V, MIPS32, and PowerPC64 ELF targets.

— Names versions and architectures but no CLI invocation rea-agents-6.10.0

THINNER COVERAGE BELOW

04 **UEFI firmware inspection and extraction tools** Not checked yet

NEW

55

Three new offline CLI and MCP tools inspect and extract UEFI firmware volumes, files, and AMI NVRAM.

— Names tool count and targets but no exact command syntax rea-agents-6.10.0

05 **New format and container analysis support** Not checked yet NEW

55

REA can now inspect Android boot images and OTA payloads, NES 2.0 containers, and Maven, Gradle, and NuGet package metadata.

— Lists formats but no commands or flags for invoking analysis `rea-agents-6.10.0`

06 **Dependency evidence export and extraction** Not checked yet NEW

55

REA can export Go and Rust dependency Evidence as CycloneDX, and read cargo-auditable dependencies from WebAssembly.

— Names format and languages but no command or flag `rea-agents-6.10.0`

07 **Evidence revision 2 with subject identities** Not checked yet

IMPROVED

40

Evidence revision 2 gives file, process, debugger, and browser observations explicit subject identities.

— Describes change but no mechanism for how identities are set or used `rea-agents-6.10.0`

08 **CPython 3.14/3.15 bytecode decoding** Not checked yet NEW

30

REA can decode CPython 3.14 and 3.15 bytecode.

— Bare statement of capability, no detail on usage `rea-agents-6.10.0`

WAS THIS USEFUL? 👍 👎

Prefect FastMCP

Feedback ⓘ v4.2.3 NOTES ↗ RANK

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

> FastMCP's single release in this window adds nested tracing for background tasks and bumps a dependency requirement for its tasks extension.

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

01 **Nested background-task tracing via parent_task_spans**

Not checked yet NEW 75

A new `parent_task_spans` setting nests each worker span inside the `tools/call` request span that submitted it, so the whole job reads as one trace in observability tooling. The setting is off by default.

— Names exact setting and behaviour but no code example v4.2.3

THINNER COVERAGE BELOW

02 **fastmcp-tasks requires newer pydocket** Not checked yet BREAKING 45

`fastmcp-tasks` now requires `pydocket>=0.29.1`, a breaking dependency bump for anyone pinning an older pydocket version.

— Names exact version requirement but no migration detail v4.2.3

└──▷ **BREAKING ON UPGRADE**

! `fastmcp-tasks` now requires `pydocket>=0.29.1`.

WAS THIS USEFUL? 👍 👎

gram

Feedback ⓘ server@2.38.0 NOTES ↗ RANK

Gram provides MCP servers and AI-agent tools from APIs with access controls and observability.

> Gram's latest releases add operator-driven MCP client registration and credential-scoped access controls, plus admin tooling for hooks rollout and team usage visibility in the catalog.

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

01 **MCP client registration for operators and agents** Not checked yet

NEW 85

Operators can register an MCP client against a user session issuer, with Speakeasy minting the client ID and secret — the secret is returned once and only its hash is stored. Agents can also be registered manually from the Connections tab under Identity > MCP Sessions via a new Register agent button, supplying a name, redirect URIs, and token endpoint auth method; registered agents appear in the tab before they sign in.

— Names exact UI path and registration flow but no API endpoint dashboard@0.152.0 server@2.38.0

02 **Hooks plugin rollout controls for staff** Not checked yet NEW

78

Staff can manage hooks plugin version rollout from the admin dashboard, setting a platform-wide default pin and per-organization overrides on the new Hooks rollout page and each organization's Features tab. Staff Admin MCP also gains read-only `get_hooks_rollout` and `get_organization_hooks_rollout` tools to query rollout state.

— Names exact tools and dashboard pages, no runnable command server@2.38.0

03 **Redirect URI replacement with permission checks** Not checked yet

NEW 70

Operators can replace the redirect URIs of a registered client (except a CIMD client) using `project:write` on the issuer's project or `org:admin` for an organization-owned issuer; both actions are recorded in the audit log.

— Names exact permission scopes and audit behavior server@2.38.0

04 **Credential-scoped roles and agent permissions** Not checked yet

NEW 65

Roles and agents can now be limited to particular upstream credentials — the role editor and an agent's Permissions page let you tick all credentials, one kind, or one credential, with a block that wins over any tick.

— Clear UI mechanism but no config key or API named `server@2.38.0`

---

05 **Catalog team usage data from Shadow MCP** Not checked yet NEW

60

Catalog cards now show team usage data sourced from Shadow MCP — for example 'Used by 5 people · 51 calls' — and the table view adds a Your team column with a Most used by your team sort for org admins.

— Concrete example data and UI column named `dashboard@0.152.0`

THINNER COVERAGE BELOW

06 **Auto-detect authorization server for credentials** Not checked yet

IMPROVED

50

Adding a credential for a user's own account now auto-detects the authorization server the MCP server advertises and preselects or adds it, removing the need to visit Authorization Servers first.

— Describes before/after but no named surface `server@2.38.0`

---

07 **Auto-naming for MCP servers and credentials** Not checked yet

IMPROVED

48

New MCP servers, credentials, and authorization servers now receive a readable name automatically — either the name the server reports for itself or one derived from its address (for example, Canva for `mcp.canva.com`).

— Describes mechanism with example, no action required `server@2.38.0`

WAS THIS USEFUL? 👍 👎

davinci-resolve-mcp

Feedback ⓘ v5.4.0 NOTES ↗ RANK

davinci-resolve-mcp is an MCP server integration for DaVinci Resolve Studio.

> davinci-resolve-mcp v5.4.0 adds a pluggable external transport provider that lets third-party tools report and control Resolve playback, including status queries and precise in-flight frame capture.

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

01 **Pluggable transport provider for playback control** Not checked yet

NEW 91

Set `RESOLVE_MCP_TRANSPORT_PROVIDER` to a local executable or a JSON argv array (e.g. `["python3", "-m", "my_transport"]`) to plug in an external playback controller, and `RESOLVE_MCP_TRANSPORT_PROVIDER_CWD` to control its working directory. Play state can be queried via resolve_control(action="transport_status"), which returns `playing_forward`, `playing_reverse`, `looping`, `paused`, or `unknown` from the configured provider. `timeline_frame` capture calls gain a `during_playback` option with modes `refuse` (default, unchanged behaviour), `live` (return the provider's newest frame without stopping), or `pause` (provider pauses, exports the exact frame, then restores play state); when a provider is configured, the `PLAYBACK_ACTIVE` remediation message now names the `live` and `pause` options. The transport provider contract is documented in `src/utils/transport_provider.py`.

Point the server at a site-specific transport provider so it can query and control playback on Resolve 19.x where the scripting API has no Play/Stop.

```
$ export RESOLVE_MCP_TRANSPORT_PROVIDER='["python3", "-m",
"my_transport"]'
export RESOLVE_MCP_TRANSPORT_PROVIDER_CWD='/opt/my_transport'
# start the server as normal; transport calls now route through
the provider
```

— Names env vars, action values, modes and contract file; full mechanism given v5.4.0

WAS THIS USEFUL? 👍 👎

3dsmax-mcp

Feedback ⓘ v2.0.1 NOTES ↗ RANK

3dsmax-mcp is an MCP server for 3ds Max

> 3dsmax-mcp 2.0.1 collapses its toolset into a single `run_python` entry point backed by a native C++ `_mx` SDK, slashing prompt overhead from 103k to ~2k tokens, while adding renderer-agnostic render/light/camera control and removing the old core/progressive profiles and safe mode.

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

01 **Single run_python entry point replaces 180+ tools** Not checked yet

NEW 88

All 3ds Max operations now go through one `run_python` tool with full `mx.tools.*` access, mx.help(), and mx.find_tools() for client-side discovery, replacing 180+ individual tool definitions and dropping prompt overhead from 103k to ~2k tokens.

– Names exact tool, helper calls, and token numbers. v2.0.1

02 **Native C++ _mx SDK for scene access** Not checked yet NEW

83

Scene data can be read at microsecond speed via the native C++ `_mx` SDK, including `sdk.nodes`, `sdk.materials`, `sdk.classes`, sdk.animate(), modifiers, tracks, and keys, supported across 3ds Max 2023–2027. All changes are wrapped in one atomic undo step with automatic rollback on exception.

– Mechanism and named SDK surfaces, but no runnable snippet. v2.0.1

03 **Renderer-agnostic lighting and render settings control**

Not checked yet NEW 73

`mx.params` (`render_setup` and `parameters`) now controls lighting, sun/sky, cameras, and render settings across V-Ray, Corona, Octane, FStorm, ART, and Scanline, with full class and instance parameter introspection.

– Names renderers and param surfaces but little behavioural detail. v2.0.1

04 **Removal of core/progressive profiles and safe mode**

Not checked yet 65

BREAKING

The `core` and `progressive` tool profiles are removed; any `mcp_config.ini` or `MCP_TOOL_PROFILE` setting referencing them automatically falls back to `code`. Safe mode is also removed, so Code mode now requires safe mode to be off.

– Clear migration note with config key and env var named. v2.0.1

└──▷ **BREAKING ON UPGRADE**

! The `core` and `progressive` tool profiles are removed; any `mcp_config.ini` or `MCP_TOOL_PROFILE` referencing them automatically falls back to `code`.

! Safe mode is removed; Code mode requires safe mode to be off.

WAS THIS USEFUL?  

memwhale

Feedback ⓘ v0.29.0 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 v0.29.0 adds automatic lesson drafting from recurring errors with a review queue, lets you seed a new store by importing shell history, and introduces Codex pre-run command blocking.

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

01 Shell history import to seed new store

Not checked yet NEW

83

`mw import --shell-history` reads `$HISTFILE`, `~/.zsh_history`, or `~/.bash_history` and imports the newest 5000 commands by default; use `--limit N` to change that cap.

Bootstrap a fresh store on a new machine so the first search actually returns results.

```
$ mw import --shell-history
```

Cap the import to the most recent 500 commands to keep the initial store small.

```
$ mw import --shell-history --limit 500
```

— Exact command, flag, default, and source files given. v0.29.0

02 Automatic lesson drafting with review queue

Not checked yet NEW

80

memwhale auto-drafts a lesson when the same error has failed on three or more days and no lesson already covers it, placing it in a review queue without affecting recall until approved. Review and approve drafts with `mw memory pending` and `mw memory approve <id>`, or trigger drafting on demand with `mw memory suggest`.

Review what memwhale has drafted from recurring failures, then approve a specific draft to put it into active recall.

```
$ mw memory suggest
mw memory pending
mw memory approve 12
```

— Names trigger condition, review commands, and approval flow. v0.29.0

THINNER COVERAGE BELOW

03 **Codex pre-run command blocking** Not checked yet NEW 15

memwhale adds Codex pre-run command blocking, mentioned alongside this release's other additions but not further detailed in the changelog.

— Named only in summary, no mechanism or usage given. v0.29.0

WAS THIS USEFUL? 👍 👎

gograph

Feedback ⓘ v1.7.22 NOTES ↗ RANK

Gograph is a local-only Go static analysis engine with a built-in MCP server. Gives AI coding agents deterministic structural awareness: call graphs, impact analysis, symbol search, and more.

> gograph's biggest update this window is a hook-guard access-control system for AI coding agents (Claude Code, Codex, Antigravity) with granular opt-in rules for read/write/session restrictions, alongside dropping Windows binary builds.

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

01 **Hook-guard access control for AI coding agents** Not checked yet

NEW 95

gograph adds a hook for Claude Code, Codex and Antigravity that auto-detects the payload format and returns exit 2 with reason on stderr for Claude Code and Codex, or `{"decision":"allow"|"deny"}` on stdout for Antigravity. The previous all-or-nothing `--strict` flag was split into separate opt-in rule flags: `--deny-go-writes` restricts shell writes to Go files inside indexed repositories; `--deny-region-reads` restricts `cat`, `head`, `sed` and similar reads of Go files inside indexed repositories (counts, checksums, `git show` and `git diff` remain allowed, and the rule does not apply to Codex); `--require-reason` requires a `# gograph-cannot: <reason>` comment on Go text searches; and `--require-session` requires an open gograph session before edits are allowed. A new `--ask` flag switches enforcement to interactive prompting in Claude Code and Antigravity rather than denying outright, and hook-guard now serves Codex and Antigravity alongside the existing gograph-first policy.

Block an AI coding agent from silently writing Go source files while still letting it read counts and diffs — good for read-heavy code-review sessions.

```
$ gograph --deny-go-writes
```

Require the agent to justify every Go text search with a comment before the session begins, surfacing its intent before it touches the codebase.

```
$ gograph --require-reason --require-session
```

Run all four access-control rules and prompt interactively in Claude Code and Antigravity rather than denying outright, so you can approve borderline actions in real time.

```
$ gograph --ask
```

— Names every flag and exact behaviour per agent, with runnable commands. v1.7.22

└──▷ **BREAKING ON UPGRADE**

! Windows binaries are no longer built or published; macOS and Linux targets are unchanged.

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ Inference Serving & Optimization

Niko1221 Strata

Feedback ⓘ v0.2.17 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 v0.2.17 reworks several CUDA/CPU performance-tuning internals — including a PCIe-aware CPU prompt share, new per-card fetch defaults for RTX 50-series cards, and a higher RAM budget for experts on 64 GB PCs — while changing two long-standing defaults that will alter behaviour from prior versions.

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

01 Expert tail-skip control via STRATA_ROUTE_TAIL_SKIP

Not checked yet BREAKING 93

Control how often the engine skips rarely used model experts by setting `STRATA_ROUTE_TAIL_SKIP` in the `"env"` of a model's run config, via Expert settings, or with `strata launch --set STRATA_ROUTE_TAIL_SKIP=7`; ranks 1–9 are available, and 0 means off. It now defaults to off (0) on every platform, whereas it previously defaulted to rank 7 on NVIDIA and Radeon RX 9000 cards, so answers will differ from 0.2.16 defaults unless explicitly set.

Re-enable the route tail skip on a system where the slight speed loss from turning it off matters more than the small quality difference.

```
$ strata launch --set STRATA_ROUTE_TAIL_SKIP=7
```

— Named flag, exact values, command, and before/after default behaviour. v0.2.17

02 PCIe-aware auto-tuning of CPU prompt share

Not checked yet NEW 90

The CPU prompt share on compute capability 12.0 cards now reads the PCIe link speed and automatically raises the token limit and expert block size; set `STRATA_PREFILL_CPU_SHARE_AUTO=0` to revert to 3,072 tokens and 8-token experts, or tune with `STRATA_PREFILL_CPU_SHARE_MAX` and `STRATA_PREFILL_CPU_SHARE_NT`.

Pin the PCIe-aware CPU prompt share limits by hand on a Blackwell card connected via a slow PCIe link, instead of letting auto-detection decide.

```
$ strata launch --set STRATA_PREFILL_CPU_SHARE_MAX=12288 --set
STRATA_PREFILL_CPU_SHARE_NT=64
```

— Mechanism, default values, and runnable override command all given. v0.2.17

03 **RAM budget formula for experts held in RAM** Not checked yet

88

BREAKING

The RAM budget for experts in RAM on a PC where the model does not fit is now `RAM - 17 - streamed KV cache` GiB instead of `RAM - 24`, raising a 64 GB PC from 40 GiB to 47 GiB of experts held in RAM; override with `--ram-keep-gib` or `STRATA_RAM_KEEP_GIB`. This leaves less free RAM (9.3 GiB instead of 16.6 GiB), and existing installs migrate automatically on the first `strata` command after the update unless a custom budget is set.

— Exact formula, numbers, and override flags given, no runnable example. v0.2.17

04 **Adaptive fetch default for RTX 50-series cards** Not checked yet

NEW

80

Enables `STRATA_ADAPT_FETCH=2` by default on single RTX 50-series cards of 16 GB or less (compute capability 12.x) for faster decode, with `STRATA_ADAPT_FETCH=0` to turn it off; also lets you revert to the 64-token short-read rule on Blackwell cards with `--short-read 64` or `STRATA_SM120_PERF=0`.

— Multiple named flags and exact fallback values, no example command. v0.2.17

05 **CPU expert pool dependency mode** Not checked yet NEW

70

Enable the CPU expert pool dependency mode with `STRATA_POOL_DEP` to run each expert's down rows as soon as its gate and up rows finish; revert with `STRATA_POOL_DEP=0`.

— Mechanism described with named flag, no example command. v0.2.17

THINNER COVERAGE BELOW

06 **CPU prompt share enabled by default on dual-CUDA splits**

Not checked yet IMPROVED

55

Enable the CPU prompt share on a split of two CUDA cards by default; turn it off with `STRATA_SPLIT_CPU_SHARE=0`.

— Named flag but thin, no mechanism beyond default-on/off. v0.2.17

07 **Windows shared-memory reservation for multi-GPU setups**

Not checked yet NEW

55

Reserve Windows shared-memory for two-or-more CUDA card setups with `STRATA_WDDM_SHARED_RESERVE_MIB` (default 2048 MiB).

— Named flag with default value but no further mechanism. v0.2.17

---

08 **Transparent huge pages for the arena** Not checked yet NEW

55

Enable transparent huge pages for the arena with `STRATA_ARENA_THP=1` for a decode speed increase on supported hosts.

— Named flag and benefit stated, no mechanism or example. v0.2.17

└──▷ **BREAKING ON UPGRADE**

! `STRATA_ROUTE_TAIL_SKIP` now defaults to off (0) on every platform; previously it defaulted to rank 7 on NVIDIA and Radeon RX 9000 cards, so answers will differ from 0.2.16 at the defaults unless you explicitly set `STRATA_ROUTE_TAIL_SKIP=7`.

! The RAM budget for experts on 64 GB PCs changes from 40 GiB to 47 GiB, leaving less free RAM (9.3 GiB instead of 16.6 GiB); existing installs migrate automatically on the first `strata` command after the update unless you have set your own budget.

WAS THIS USEFUL?  

Experiential Labs Experiential

Feedback ⓘ

v0.7.257 NOTES ↗

Experiential is the open source, zero markup gateway for BYOK, self-hosted and 1000+ marketplace models. It learns from your traffic to cut costs, recommend better models, and train a specialized model you own.

Adds `POST /v1/images/edits` with new billing fields: `image_input_nano_usd_per_million_tokens`, `image_input_tokens`, `GatewayUnitPrices.input_rates`, and `input_quantity_milli`.

└──▷ GET THIS VERSION ▾

pipx

```
$ pipx install experiential==0.7.257
# already installed? upgrade to this version:
$ pipx install --force experiential==0.7.257
```

git

```
$ git clone --branch v0.7.257
https://github.com/experientiallabs/experiential.git
# already have the repo? check out this version:
$ git checkout v0.7.257
```

PyPI attestation · checked 20 Nov Wrong command?

NEW INTEGRATION Call image edits on GPT (OpenAI and Azure OpenAI) and xAI Grok Imagine models using the `POST /v1/images/edits` endpoint. Not checked yet

Bill GPT image edits by token using the new `image_input_nano_usd_per_million_tokens` price field on price cards and `image_input_tokens` usage field. Not checked yet

Bill xAI image edits per image using `GatewayUnitPrices.input_rates` for consumed source images, the `'edit'` variant for returned images, and `input_quantity_milli` on settled units. Not checked yet

WAS THIS USEFUL? 👍 👎

NVIDIA TensorRT-LLM

Feedback ⓘ v1.3.0 NOTES ↗ RANK

TensorRT-LLM compiles and serves large language models on NVIDIA GPUs with quantization, in-flight batching, and speculative decoding.

> TensorRT-LLM v1.3.0 adds a VisualGen vertical for image and video generation, an Anthropic Messages API and embeddings endpoint, MXFP8/NVFP4 quantization, and a gRPC server, while making PyTorch the sole backend and removing the legacy TensorRT backend, Triton engine models, TRTLLMSampler and two-model speculative decoding.

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

01 OpenAI-compatible API endpoints and request controls

Not checked yet NEW 85

Adds new OpenAI-compatible endpoints `/v1/messages` for Anthropic Messages API clients and `/v1/embeddings` with an llm.encode() fast path for encoder-only model embeddings, plus per-request `priority`, named-function `tool_choice`, `parallel_tool_calls` and logprobs support across the OpenAI-compatible APIs, including the Completions API.

Obtain text embeddings from an encoder-only model for downstream retrieval or reranking.

python

```python
from tensorrt_llm import LLM

llm = LLM(model="nvidia/NV-Embed-v2")
embeddings = llm.encode(["Represent this sentence for
retrieval:", "TensorRT-LLM is fast."])
print(embeddings[0].outputs.embedding)
```

— Named endpoints plus a runnable embedding code example Names request fields but gives no usage example

v1.3.0

02 PyTorch-only backend; legacy APIs removed Not checked yet

85

BREAKING

The TensorRT backend is removed; PyTorch is the only backend, removing `trtllm-build`, `TrtLlmArgs`, `BuildConfig`, `MedusaDecodingConfig`, `LookaheadDecodingConfig`, `trtllm-bench build` and `trtllm-bench --engine_dir`. The legacy C++ Executor API, legacy runtime and batch-manager APIs,

and the `cpp_only` build option are removed. TensorRT-engine Triton backend models are removed, leaving only the `llmapi` Triton model. `TRTLLMSampler` is removed in favor of `TorchSampler` (requiring FlashInfer); the `sampler_type` argument is ignored with a warning and `disable_flashinfer_sampling` is removed. Two-model speculative decoding is removed — the `eagle3_one_model` and `mtp_eagle_one_model` fields are gone, replaced by one-model implementations. AutoDeploy, Star Attention and WIDEEP are removed.

— Names every removed flag, class and option for migration v1.3.0

03 **New trtllm-serve CLI options** Not checked yet NEW 80

Add new `trtllm-serve` options `--served_model_name`, `--kv_cache_dtype` and `--middleware` to control model aliasing, KV cache precision and request middleware.

Serve a model with a custom alias and FP8 KV cache to control memory use and match a client-facing name.

```
$ trtllm-serve "nvidia/Llama-3.1-8B-Instruct-FP8" --
served_model_name my-llama --kv_cache_dtype fp8
```

— Exact flags with a runnable serve command v1.3.0

04 **Locality domain control for NVFP4 MoE on Rubin** Not checked yet

NEW 75

Control locality domain (uGPU) localization for NVFP4 MoE on Rubin — enabled by default; disable with `DISABLE_LOCALITY_DOMAINS=1`.

— Exact env var toggle, limited surrounding context v1.3.0

05 **MXFP8 and NVFP4 weight-only quantization support** Not checked yet

NEW 75

Adds MXFP8 (W8A8) linear and MoE quantization, including loading MXFP8 checkpoints such as `MiniMax-M3`, and a new W4A16 NVFP4 weight-only quantization path through Marlin to run NVFP4 checkpoints on Hopper and Ada Lovelace.

— Names formats, checkpoints and hardware but no command v1.3.0

06 **VisualGen image/video generation in trtllm-serve** Not checked yet

NEW

70

Adds a new VisualGen vertical to `trtllm-serve` for image and video generation alongside LLM inference, supporting `FLUX.1`, `FLUX.2`, GLM-Image, Qwen-Image, `Wan 2.2`, `LTX-2`, `HunyuanVideo 1.5`, Cosmos3 and `MiniMax-H3`, plus a new `trtllm-eval` pipeline to evaluate VisualGen models.

— Lists supported models and eval pipeline but no runnable command v1.3.0

07 **Prometheus metrics and profiling endpoints** Not checked yet NEW

60

Monitor inference with Prometheus metrics and control profiling via new `/start_profile` and `/stop_profile` endpoints.

— Named endpoints but no example of invocation v1.3.0

08 **RL workflow support: router replay and inflight weight updates**

Not checked yet NEW

60

Adds RL workflow support with Router Replay's `return_routed_experts` field for returning per-token MoE routing decisions, and inflight weight updates for updating model weights in-flight.

— Named field but no command or example shown v1.3.0

THINNER COVERAGE BELOW

09 **Reasoning token budget cap** Not checked yet NEW

50

Add an experimental `thinking_token_budget` parameter to cap reasoning-model token budgets.

— Named parameter but no usage example given v1.3.0

10 **gRPC server and multi-process HTTP frontends** Not checked yet

NEW

40

Serve models with a new gRPC server and multi-process HTTP frontends.

— Bare description without configuration or usage detail v1.3.0

└──▷ **BREAKING ON UPGRADE**

! The TensorRT backend is removed; PyTorch is the only backend. `trtllm-build`, `TrtLlmArgs`, `BuildConfig`, `MedusaDecodingConfig`, `LookaheadDecodingConfig`, `trtllm-bench build` and `trtllm-bench --engine_dir` are removed.

! The legacy C++ `Executor` API, the legacy runtime and batch-manager APIs, and the `cpp_only` build option are removed.

- ! TensorRT-engine Triton backend models are removed; only the `llmapi` Triton model remains.
- ! `TRTLLMSampler` is removed; `TorchSampler` (requiring FlashInfer) is the only sampler. The `sampler_type` argument is ignored with a warning, and `disable_flashinfer_sampling` is removed.
- ! Two-model speculative decoding is removed; the `eagle3_one_model` and `mtp_eagle_one_model` fields are removed — use the one-model implementations.
- ! AutoDeploy, Star Attention and WIDEEP are removed.

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

MLX Serve

Feedback ⓘ v26.11.3 NOTES ↗ RANK

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

> mlx-serve's v26.11.3 release adds text-to-speech via Ming-3.0-tts, request-priority scheduling, several new API parameters and environment variables, and new media-generation models including a Quick Create feature for videos, memes, and songs.

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

01 Text-to-speech via Ming-3.0-tts

Not checked yet NEW 75

The existing `/v1/audio/speech` endpoint now supports the Ming-3.0-tts model, including voice cloning and French language synthesis.

Generate speech from text using the new Ming-3.0-tts model, including voice cloning, via the standard audio endpoint.

```
$ curl http://localhost:8080/v1/audio/speech \
  -H 'Content-Type: application/json' \
  -d '{"model": "Ming-3.0-tts", "input": "Hello, welcome to the demo.", "voice": "clone"}'
```

— Named endpoint and model with a runnable curl example. v26.11.3

02 High-priority request scheduling

Not checked yet NEW 75

Send `X-MLX-Priority: high` in the request header to schedule a request ahead of a long prompt's in-flight chunks.

Insert a high-priority request between chunks of a long in-flight prompt so it is processed sooner.

```
$ curl http://localhost:8080/v1/chat/completions \
  -H 'Content-Type: application/json' \
  -H 'X-MLX-Priority: high' \
  -d '{"model": "your-model", "messages": [{"role": "user", "content": "Quick question"}]}'
```

— Exact header name with a runnable curl example. v26.11.3

03 **New request parameters and environment variables** Not checked yet

NEW

70

Added the `suppress_cjk` parameter to suppress Chinese, Japanese, and Korean characters from a reply; `logit_bias` support in requests to engines that support it; the `HF_ENDPOINT` environment variable to route model downloads to a mirror; llama.cpp-style timings now readable from the Responses stream for proxies to consume; and `/v1/models` now answers from a cache for faster queries.

Route all model downloads to a private Hugging Face mirror instead of the public hub.

```
$ export HF_ENDPOINT=https://your-hf-mirror.example.com
```

— Names every parameter and env var but each is a thin addition. v26.11.3

THINNER COVERAGE BELOW

04 **New media generation models and Quick Create** Not checked yet

NEW

55

Added MiniMax-H3-Image for text-to-image and multi-reference image editing, HyperFlow H3 Flash for H3 video generation from keyframes or a reference image in a few steps, and a Quick Create feature for generating videos with sound and captions, meme images, and original songs from photos of people and pets.

.

— Names the models and feature but no usage mechanism given. v26.11.3

WAS THIS USEFUL? 👍 👎

◆ ML Platforms & SDKs

OpenAI TypeScript SDK

Feedback ⓘ v7.62.0 NOTES ↗

RANK

The OpenAI TypeScript SDK provides typed JavaScript and TypeScript clients for OpenAI APIs and streaming responses.

> The OpenAI TypeScript SDK added Live WebRTC standby support and a new selectable model, `gpt-live-1.1`, for the Live API.

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

01 **Live WebRTC standby mode** Not checked yet NEW 35

The SDK now supports Live WebRTC standby mode via the API, enabling standby session management for real-time WebRTC connections.

– Describes what it does but no usage mechanism v7.62.0

02 **gpt-live-1.1 model added to Live API** Not checked yet NEW 35

Adds `gpt-live-1.1` as a selectable model in the Live API model choices.

– Names exact model id but no further detail v7.61.0

WAS THIS USEFUL? 👍 👎

OpenAI Python SDK

Feedback ⓘ v3.57.0 NOTES ↗ RANK

The OpenAI Python SDK provides typed Python clients for OpenAI model APIs, streaming, and asynchronous requests.

> The OpenAI Python SDK added standby support for Live WebRTC sessions and a new typed model choice for the Live API.

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

01 **gpt-live-1.1 model choice for Live API** Not checked yet NEW

50

Adds `gpt-live-1.1` as a typed model identifier usable in Live API calls via the SDK.

— Names exact model identifier, clear starting point v3.56.0

02 **Live WebRTC standby support** Not checked yet NEW

35

Adds Live WebRTC standby support to the SDK, enabling standby-mode connections in real-time WebRTC sessions.

— Names the capability but no mechanism or usage detail v3.57.0

WAS THIS USEFUL? 👍 👎

Anthropic TypeScript SDK

Feedback ⓘ sdk-v1.11.0

NOTES ↗ RANK

The Anthropic TypeScript SDK provides typed JavaScript and TypeScript clients for Claude APIs, streaming, and tool use.

> The Anthropic TypeScript SDK's v1.11.0 release expands the Admin API with programmatic Claude Code configuration and a new sandboxed plugin error type.

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

01 **Sandbox egress error for Admin API plugins** Not checked yet NEW

45

Adds the `sandbox_egress_restricted` error to Admin API plugin methods, surfacing a new failure mode for sandboxed plugin calls.

– Names exact error value but not endpoint or handling code sdk-v1.11.0

02 **Claude Code managed settings in Admin API** Not checked yet NEW

35

Adds Claude Code managed settings to the Admin API, enabling programmatic control of Claude Code configuration.

– No concrete endpoint or settings fields named sdk-v1.11.0

WAS THIS USEFUL? 👍 👎

Anthropic Python SDK

Feedback ⓘ v2.11.0 NOTES ↗ RANK

The Anthropic Python SDK provides typed Python clients for Claude APIs, streaming responses, and tool use.

> The Anthropic Python SDK's v2.11.0 release expands the Admin API with Claude Code managed settings and a new sandbox egress error type for plugin calls.

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

01 **Sandbox egress error for plugin calls** Not checked yet NEW

50

Admin API plugin methods now surface a `sandbox_egress_restricted` error, indicating a new failure mode for sandboxed plugin calls.

— Names exact error code but no usage context v2.11.0

02 **Claude Code managed settings in Admin API** Not checked yet NEW

30

The Admin API gains Claude Code managed settings, expanding programmatic control over Claude Code deployments.

— Names the surface but no mechanism or fields given v2.11.0

WAS THIS USEFUL? 👍 👎

AWS Amazon Bedrock AgentCore

Feedback ⓘ

AWS Agent Registry: Search types and search pagination (2026-11-19) BLOG ↗

RANK

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

> Amazon Bedrock AgentCore expanded the AWS Agent Registry with richer search and skill package distribution, added JWT-based and organization-wide access controls, and gave Runtime agents direct S3 bucket mounts plus elevated-privilege and virtualization options on Instances.

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

01 Mount your own S3 bucket in Runtime agents

Not checked yet NEW

90

AgentCore Runtime agents on platform version V2 can mount an S3 bucket via the `s3BucketMount` file system configuration; AgentCore runs Mountpoint for Amazon S3 on the agent's behalf so the container needs no mount client. Including `{runtimeSessionId}` in the mount prefix isolates each session's files within the bucket.

Configure an S3 bucket mount on a V2 agent runtime so every session writes to its own prefix — no mount client required inside your container.

json

```
{
  "fileSystemConfiguration": {
    "s3BucketMount": {
      "bucketName": "my-agent-bucket",
      "prefix": "sessions/{ runtimeSessionId}/",
      "mountPath": "/mnt/s3"
    }
  }
}
```

— Named config key with mechanism, scope limit, and a runnable example

Runtime: Mount your own Amazon S3 bucke…

02 JWT claim condition keys for Gateway invocations

Not checked yet

NEW INTEGRATION NEW

85

Gateway resource policies can filter `InvokeGateway` requests using new `bedrock-agentcore:InboundJwtClaim/*` IAM condition keys, letting callers be restricted based on claims carried in their JWT — for example allowing only tokens where a given claim equals a specific value.

Restrict gateway invocations to callers whose JWT carries a specific claim value — for example, allowing only tokens where the `role` claim equals `agent`.

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "bedrock-agentcore:InvokeGateway",
      "Resource": "arn:aws:bedrock-agentcore:<region>:<account-id>:gateway/<gateway-id>",
      "Condition": {
        "StringEquals": {
          "bedrock-agentcore:InboundJwtClaim/role": "agent"
        }
      }
    }
  ]
}
```

— Named IAM condition key and action with a runnable policy example

Gateway: JWT claim condition keys for g…

---

03 **Search types and pagination in Agent Registry** Not checked yet

NEW

73

The AWS Agent Registry search API now supports keyword, semantic, and hybrid search via the `searchType` parameter, with `nextToken` pagination through results; submitting an empty query lists all records matching a filter. The registry's Model Context Protocol (MCP) endpoint gains a new `search_discoverable_registry_records_paginated` tool accepting the same `searchType` and `nextToken` options.

— Named API params and MCP tool, but no full request example

AWS Agent Registry: Search types and se…

THINNER COVERAGE BELOW

04 **Skill packages and S3/GitHub sync in Agent Registry**

Not checked yet NEW

57

Skill records in the AWS Agent Registry can now carry an attached skill package — a `.zip` archive containing a `SKILL.md` and supporting files — so consumers can download a validated package directly from the registry. Skill records can also be synced from Amazon S3 or GitHub so the registry stays current without manual uploads.

— Names package format and sync sources but no command or API call shown

AWS Agent Registry: Skill packages and …

05 **Elevated privileges and virtual machines on Runtime Instances**

Not checked yet NEW 50

On the Instances compute type, agents can now run with elevated privileges — enabling coding agents to build and run containers, install system packages, and change kernel settings, configured via the privilege mode setting documented under 'Configure the privilege mode for Runtime Instances' — and can enable nested virtualization to run their own KVM-based virtual machines and emulators inside a session.

— Describes capabilities and a doc reference but no concrete flag or command

Runtime: Run agents with elevated privi… Runtime: Run virtual machines in agents…

06 **Resource control policies for AgentCore resources** Not checked yet

NEW 45

AWS Organizations resource control policies (RCPs) can now set organization-wide access limits on Amazon Bedrock AgentCore resources, including for callers that are not IAM principals within the organization.

— Describes the control but gives no concrete policy syntax

Resource control policies for Amazon Be…

Gateway: Discover and call products from AWS Marketplace with the AWS Marketplace conn

BLOG ↗ STABLE

AgentCore Gateway adds a built-in AWS Marketplace connector exposing `SearchListings`, `GetListing`, and `ListFulfillmentOptions` MCP tools with x402 payment support.

New built-in AWS Marketplace connector lets agents search listings, read listing details, and retrieve callable endpoints and payment details via three MCP tools: `SearchListings`, `GetListing`, and `ListFulfillmentOptions`. Not checked yet

Agents can pay for AWS Marketplace products using x402 payments through the connector. Not checked yet

WAS THIS USEFUL? 👍 👎

ZenML

Feedback ⓘ 0.99.0 NOTES ↗ RANK

ZenML is an MLOps and LLMOps framework for building portable pipelines that run the same code across local, cloud, and orchestrator backends.

> ZenML 0.99.0 adds a deployment-origin filter for pipeline runs and per-pod Kubernetes image pull control, alongside several breaking changes to RBAC, logging, and dependency requirements.

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

01 Deployment-origin filter for pipeline runs

Not checked yet NEW

90

The new `triggered_by_deployment` parameter in `PipelineRunFilter` and `Client.list_pipeline_runs` lets you filter pipeline runs by whether they were triggered by a deployment, separating deployment-triggered runs from manual or batch runs in run statistics.

List only pipeline runs that were triggered by a deployment, to keep deployment-triggered runs separate from manual or batch runs.

python

```python
from zenml.client import Client

client = Client()
deployment_runs =
client.list_pipeline_runs(triggered_by_deployment=True)
for run in deployment_runs:
    print(run.name, run.status)
```

— Named API with runnable filtering example 0.99.0

02 Kubernetes pod image pull policy control

Not checked yet NEW

90

`image_pull_policy` in `KubernetesPodSettings` can be set to `Always`, `IfNotPresent`, or `Never` to control per-pod image pull behaviour, for example forcing a fresh pull when reusing a mutable tag like `:dev` to avoid stale cached images.

Force Kubernetes to always pull the latest image when reusing a mutable tag such as `:dev`, avoiding stale cached images.

```python
from
zenml.integrations.kubernetes.flavors.kubernetes_orchestrator_fla
vor import KubernetesPodSettings
from zenml import pipeline

@pipeline(
    settings={
        'orchestrator.kubernetes':
KubernetesPodSettings(image_pull_policy='Always')
    }
)
def my_pipeline():
    ...
```

— Named config key with runnable pipeline example `0.99.0`

03 **Stricter RBAC for stack component access** Not checked yet BREAKING

70

ZenML Pro custom roles that grant only `read` on stack components or service connectors can no longer use those resources when creating or updating stacks or components — the `attach` action must also be granted to preserve access.

— Names exact role action required but no command `0.99.0`

THINNER COVERAGE BELOW

04 **OpenTelemetry log exporter force_flush requirement**

Not checked yet BREAKING

55

Custom OpenTelemetry log exporters must now implement a force_flush() method to remain compatible with the upgraded OpenTelemetry SDK.

— Names required method, no migration example `0.99.0`

05 **Minimum certifi version requirement** Not checked yet BREAKING

50

ZenML now requires `certifi>=2026.10.19`; environments pinning an older version must upgrade to meet the new minimum.

— Exact version pin named, minimal further detail `0.99.0`

06 **Reproducible orchestrator run name suffixes** Not checked yet

IMPROVED

40

Orchestrator run names can now optionally follow the user's random seed, making the generated name suffixes reproducible across runs.

— No specific flag or config key named `0.99.0`

└──▷ BREAKING ON UPGRADE

! ZenML Pro custom roles that grant only `read` on stack components or service connectors can no longer use those resources when creating or updating stacks or components — grant the `attach` action as well to preserve access.

! Custom OpenTelemetry log exporters must now implement a force_flush() method to remain compatible with the upgraded OpenTelemetry SDK.

! ZenML now requires `certifi>=2026.10.19`; environments that pin an older version must update to meet the new minimum.

WAS THIS USEFUL?  

Snowflake Cortex AI

Feedback ⓘ

Nov 17, 2026: Data governance policy inheritance for Cortex Search (Public Preview)

NOTES ↗ RANK

Snowflake Cortex AI provides managed language-model functions, search, and agents that work with data governed in Snowflake.

> Snowflake Cortex AI now enforces per-caller row access and masking policies on Cortex Search services, closing a gap where search results could bypass governed data restrictions.

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

01 **Data governance policy inheritance for Cortex Search**

Not checked yet NEW 88

Setting `INHERIT_DATA_GOVERNANCE_POLICIES = TRUE` on a Cortex Search service definition enables per-caller enforcement of row access and masking policies, now applied to queries using the `CORTEX_SEARCH_BATCH` and `CORTEX_SEARCH_RETRIEVE` SQL functions. Available in Public Preview.

Activate policy inheritance on an existing Cortex Search service so that each caller only sees rows and columns their own row access and masking policies permit.

```
$ ALTER CORTEX SEARCH SERVICE my_search_service SET
INHERIT_DATA_GOVERNANCE_POLICIES = TRUE;
```

— Names exact config key, functions, and a runnable ALTER command.

Nov 17, 2026: Data governance policy in…

WAS THIS USEFUL? 👍 👎

◆ Inference Providers & GPU Clouds

AWS Amazon SageMaker AI

Feedback ⓘ November 19, 2026

NOTES ↗

Amazon SageMaker AI provides managed tools and infrastructure for building, training, fine-tuning, and deploying machine-learning models.

UPDATED INTEGRATION SageMaker HyperPod observability add-on and extension now export metrics to Amazon CloudWatch

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

1c-ai-connector

Feedback ⓘ v1.9.6 NOTES ↗ RANK

1c-ai-connector is a иикона — AI-коннектор для 1С:Предприятие 8.3: LLM-провайдеры, агентская петля (function calling), RAG, MCP-сервер, мониторинг ошибок с ИИ-анализом, генератор диаграмм

> 1c-ai-connector v1.9.6 improves how tool-call results are fed back to LLMs and adds draft preservation for the question field.

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

01 **Named-column row arrays for tool result tables** Not checked yet

IMPROVED

55

Tool result value tables are now passed to the model as an array of rows with named columns, covering Gemini and other providers that run multi-round tool calls.

— Explains mechanism and provider scope but no config name given v1.9.6

02 **Draft preservation for question field** Not checked yet IMPROVED

16

The question field now preserves its draft content.

— Only a bare mention with no mechanism or location detail v1.9.6

WAS THIS USEFUL? 👍 👎

synaplan

Feedback ⓘ v5.8.9 NOTES ↗ RANK

Synaplan is an own the flow: one open-source AI platform for your knowledge, models and channels – web, WhatsApp, email, Outlook, Nextcloud & ownCloud. Local and cloud models side by side, with the cost on every answer.

> Synaplan v5.8.9 ships a safer default posture for new installs, a Projects workspace for organizing work, and one-click deployment on Coolify.

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

01 **Coolify one-click deployment support** Not checked yet NEW

45

Synaplan adds a Coolify service file enabling one-click deployment on Coolify, and the update notice on Coolify installs now links directly to a Coolify-specific upgrade guide.

– Names Coolify deployment surface and upgrade guide link v5.8.9

02 **Sign-up disabled by default on new installs** Not checked yet

40

BREAKING

New installations now start with sign-up disabled and no Free-plan usage enabled by default, reducing the attack surface on first deploy.

– States the change but no config key or override path v5.8.9

03 **Projects page and sidebar organization** Not checked yet NEW

35

Synaplan adds a Projects page, a dedicated sidebar section, and the ability to move chats, giving users a way to organize work.

– Describes the feature but not its workflow mechanics v5.8.9

WAS THIS USEFUL? 👍 👎

liliang-cn CortexDB

Feedback ⓘ v2.157.0 NOTES ↗ RANK

CortexDB is a local-first AI agent memory with a knowledge graph, ontology constraints, and traceable reasoning—all in one SQLite file.

> CortexDB adds entity merging and alias resolution to its knowledge graph, letting callers consolidate duplicate nodes while keeping reads and writes consistent.

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

01 **Entity merge/unmerge with alias resolution** Not checked yet NEW

68

CortexDB adds `merge_entities` and `unmerge_entities` graph operations that accept a caller-supplied alias table. Graph writes and `find_nodes` calls now follow entity alias resolution records automatically, so queries resolve merged entities without manual remapping.

— Named functions given but no signature or usage example v2.157.0

WAS THIS USEFUL? 👍 👎

synthadoc

Feedback ⓘ v1.5.3 NOTES ↗ RANK

Synthadoc: An open-source LLM knowledge compilation engine that turns raw documents into structured, local-first wikis. A transparent, human-readable alternative to traditional RAG, which can be self-managed and self-improved without the use of any tools.

> Synthadoc v1.5.3 reworks how `serve --all` handles memory pressure, moves the `allow_external_paths` setting server-side as a breaking change, adds per-resolver model/effort routing, and introduces a blocked-domain guard for URL ingest.

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

01 Server-side `allow_external_paths` setting

Not checked yet

88

BREAKING

`allow_external_paths` moves from a per-request CLI flag to a server-side `[server]` section key in `config.toml`, or a `SYNTHADOC_ALLOW_EXTERNAL_PATHS` environment variable (default off), with the environment variable taking precedence over the file value in both directions. The per-request CLI flag is removed entirely, breaking any workflow that passed it per request, and new installs now write `allow_external_paths = false` into the generated `config.toml` template.

— Full before/after and both config surfaces named. v1.5.3

02 Blocked-domain guard for URL ingest

Not checked yet NEW

86

URL ingest jobs targeting unwanted hostnames, including subdomains, can be skipped by listing them in `.synthadoc/blocked_domains.json`; jobs are blocked before any network or LLM call, and the resolver is told why the job was skipped.

Prevent the resolver from wasting retries on domains that always time out or return unusable content.

```json
["example-spam.com", "blocked-host.internal"]
```

— Names config file path and exact blocking mechanism. v1.5.3

03 Resolver model routing and effort tuning

Not checked yet NEW

83

Contradiction, orphan, and content-health resolvers can be routed to a separate, smaller model by adding a `workflow = { provider, model }` entry under `[agents]` in

`config.toml`, distinct from the `default` agent model. Setting `agents.default.effort` to `low` (now the default for Anthropic Sonnet/Opus) trims adaptive-thinking overhead on resolver calls.

Route resolver agents to a cheap small model while keeping the default model for all other LLM calls.

```toml
[agents]
default = { provider = "anthropic", model = "claude-haiku-5-5" }
workflow = { provider = "anthropic", model = "claude-haiku-5-5" }

[agents.default]
effort = "low"
```

— Names both config keys and a working example. v1.5.3

04 **Memory-aware `serve --all`** Not checked yet IMPROVED 80

`serve --all` now skips wikis under memory pressure instead of exhausting host memory, with a `--max-memory` flag (e.g. `synthadoc serve --all --max-memory 6GB`) to cap total memory use. When a wiki is skipped for this reason, the new error code `ERR-SRV-005` is emitted.

Start all wikis on a machine with limited RAM, capping total memory use so the host stays responsive.

```
$ synthadoc serve --all --max-memory 6GB
```

— Names the flag, example command, and error code. v1.5.3

└──▷ **BREAKING ON UPGRADE**

! The `allow_external_paths` per-request flag is removed from the CLI; the setting now lives exclusively in `[server]` in `config.toml` or the `SYNTHADOC_ALLOW_EXTERNAL_PATHS` environment variable — any workflow that passed the flag per request will stop working.

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

LangChain LangSmith

Feedback ⓘ updated · Nov 20, 2026

NOTES ↗ RANK

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

> LangSmith's latest snapshot brings deep LLM Gateway observability and IAM-based Bedrock authentication, alongside a breaking overhaul of thread automations, new GPT-5.6 chat models, dual-stack networking, and several smaller dataset, tagging, and sandbox improvements.

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

01 IAM-role authentication for Bedrock via LLM Gateway

Not checked yet IMPROVED 85

The LLM Gateway can now authenticate 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. Configured under Settings > LLM Gateway > Bedrock Credentials by selecting Assumed IAM Role and entering a Role ARN and optional external ID.

Route LLM Gateway requests to Bedrock using an assumed IAM role instead of a static API key, so credentials rotate automatically.

📍 In the LangSmith UI, go to Settings › LLM Gateway › Bedrock Credentials and select Assumed IAM Role, then enter your Role ARN and optional external ID. Existing Bedrock API keys remain active and take precedence if also configured.

— Exact UI path and credential precedence rule given snapshot-20261120

02 LLM Gateway trace visibility and metadata header Not checked yet

NEW 80

Trace details now show a dedicated Gateway section with outcomes, policy limits, selected models, and fallback attempts, plus a Blocked badge in the trace tree for runs the gateway blocks. Gateway API responses include an `X-LangSmith-Gateway-Metadata` header (routing, limits, guard activity, upstream errors) that clients can attach to their own traces as `ls_gateway_info` metadata.

— Names exact header and metadata key but no UI walkthrough snapshot-20261120

03 Turns view deprecated in thread detail Not checked yet DEPRECATED

70

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 switch to the Messages or Details view.

— Gives exact deprecation dates and migration path `snapshot-20261120`

04 **Thread automations require SmithDB queries** Not checked yet

65

BREAKING

Thread automations now require SmithDB queries and show thread filters instead of run filters, no longer defaulting to `Is Trace = true`, and automations grouped by thread can now add one annotation queue item for each matched conversation so reviewers can inspect the whole thread instead of only one run. Existing thread webhook and annotation-queue automations using run filters must be updated to the new query format.

— Explains before/after but no exact migration steps `snapshot-20261120`

THINNER COVERAGE BELOW

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

IMPROVED

50

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

— Names the exact key pattern but no export command shown `snapshot-20261120`

06 **Sandbox resource monitoring and WebSocket connectivity fixes**

Not checked yet IMPROVED

50

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

— Describes behavior but gives no config keys or commands `snapshot-20261120`

07 **GPT-5.6 Sol, Terra, and Luna 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 the existing OpenAI models.

— Just names the new models, no further detail `snapshot-20261120`

08 **Tag visibility for tag-on-create permission** Not checked yet

IMPROVED

30

Users with the tag-on-create permission for datasets, projects, or prompts can now see and apply tags when creating those resources.

— Thin description, no concrete permission scope details `snapshot-20261120`

09 **IPv4/IPv6 dual-stack support** `Not checked yet` NEW 25

LangSmith services now accept both IPv4 and IPv6 connections.

— Minimal description with no configuration detail `snapshot-20261120`

└──▷ **BREAKING ON UPGRADE**

! Thread webhook and annotation-queue automations now require SmithDB queries; existing automations using run filters must be updated.

! 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; switch to the Messages or Details view.

WAS THIS USEFUL?  

Arize Phoenix

Feedback ⓘ arize-phoenix-v22.0.0 NOTES ↗ RANK

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

> Arize Phoenix extended per-span cost tracking to DECISION and EMBEDDING spans with model-type visibility, added CLI commands for managing dataset label assignments, and improved its built-in pxi assistant.

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

01 **Dataset label management via CLI** Not checked yet NEW

90

The CLI gains four `px dataset-labels` subcommands — `list`, `add`, `remove`, and `replace` — to manage dataset label assignments addressable by dataset name or GlobalID, including explicitly clearing all assignments.

– Exact runnable subcommands with addressing scheme given `@arizeai/phoenix-cli@1.24.0`

02 **Cost tracking extended to DECISION and EMBEDDING spans**

Not checked yet IMPROVED

80

Cost tracking now calculates costs for `DECISION` spans and for `EMBEDDING` spans, syncing embedding model prices into the cost manifest. The cost manifest and LiteLLM sync now include model types, generative model type is exposed in the GraphQL API, and a new `type` column was added to the `generative_models` table to record each model's type.

– Names spans, manifest, GraphQL field and DB column but no query example

`arize-phoenix-v22.0.0`

THINNER COVERAGE BELOW

03 **pxi assistant model list and chat persistence** Not checked yet

IMPROVED

35

Phoenix's built-in AI assistant (pxi) now refreshes its recommended model list and preserves chat transcripts across sessions.

– Describes behavior change but no mechanism or command `arize-phoenix-v21.3.0`

WAS THIS USEFUL? 👍 👎

openlit

Feedback ⓘ openlit-2.2.0 NOTES ↗ RANK

Openlit is a same agent Smaller bill. The tools and infrastructure to get more work from your AI.

> openlit's biggest shift this window is native Claude Code session telemetry — surfacing sessions, tool calls, tokens and cost on the Sessions and Users pages without requiring the OpenLIT hook — alongside new custom metric tagging, Microsoft/Okta SSO, self-hosted OpenAI-compatible model support, and several trace-view and search UI improvements.

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

01 Native Claude Code session telemetry

Not checked yet NEW

95

openlit now surfaces Claude Code sessions on the Sessions and Users pages via native telemetry, with native token, tool-call, and cost handling — sessions are keyed by Claude Code's `session.id` and cost is derived from `api_request` log events, though repository, working folder, code changes, and outcome columns remain empty unless the OpenLIT hook also runs. Enable export with `CLAUDE_CODE_ENABLE_TELEMETRY=1`, `OTEL_METRICS_EXPORTER=otlp`, `OTEL_LOGS_EXPORTER=otlp`, `OTEL_EXPORTER_OTLP_PROTOCOL=grpc`, and `OTEL_EXPORTER_OTLP_ENDPOINT`; add `CLAUDE_CODE_ENHANCED_TELEMETRY_BETA=1` and `OTEL_TRACES_EXPORTER=otlp` to enable beta per-tool and per-LLM-turn trace spans, which are required for sessions to appear without the OpenLIT plugin.

Stream Claude Code native traces and cost logs into openlit's Sessions view without installing the OpenLIT hook — useful for read-only environments or quick trials.

```
$ export CLAUDE_CODE_ENHANCED_TELEMETRY_BETA=1
export OTEL_TRACES_EXPORTER=otlp
export OTEL_LOGS_EXPORTER=otlp
```

Export Claude Code metrics and logs to OpenLIT over OTLP/gRPC — the minimum required to get telemetry flowing into the platform.

```
$ export CLAUDE_CODE_ENABLE_TELEMETRY=1
export OTEL_METRICS_EXPORTER=otlp
export OTEL_LOGS_EXPORTER=otlp
export OTEL_EXPORTER_OTLP_PROTOCOL=grpc
export OTEL_EXPORTER_OTLP_ENDPOINT=$OPENLIT_OTLP_ENDPOINT
export OTEL_EXPORTER_OTLP_HEADERS="Authorization=Bearer
$OPENLIT_API_KEY"
```

Also capture per-tool and per-LLM-turn trace spans so Claude Code sessions appear on the Sessions and Users pages without the OpenLIT plugin.

```
$ export CLAUDE_CODE_ENHANCED_TELEMETRY_BETA=1
export OTEL_TRACES_EXPORTER=otlp
```

— Exact env vars and runnable commands given for setup `openlit-2.2.0`

02 **Custom attributes on metrics** Not checked yet NEW 90

Tag every metric recording with arbitrary key-value pairs using `--custom_metrics_attributes` on the CLI or the `OPENLIT_CUSTOM_METRICS_ATTRIBUTES` environment variable, passing attributes as a JSON string (e.g. `{"team": "ml"}`) to group metrics by client ID, team, project, or other custom dimensions.

Group all LLM cost and latency metrics by team and project so you can slice dashboards without touching instrumentation code.

📍 `OPENLIT_CUSTOM_METRICS_ATTRIBUTES`='{"team": "ml", "project": "recommendation-engine"}' openlit ...

— Exact flag, env var, and example payload given `product docs`

THINNER COVERAGE BELOW

03 **Case-insensitive span content search** Not checked yet NEW 55

Search span content case-insensitively across Traces and Spans lists for ClickHouse sources, covering prompts, responses, errors, and status messages with bounded length and time range.

— Scope named but no query syntax or UI path given `openlit-2.2.0`

04 **Hour/minute custom time-window filters** Not checked yet IMPROVED 50

Custom time-window filters now support hour and minute selection, with previous periods aligned to equal-duration windows immediately before the selected range.

— Behavior described but no UI path or config given `openlit-2.2.0`

05 **Microsoft and Okta SSO sign-in** Not checked yet NEW 40

Add Microsoft and Okta as sign-in options, with a new instance admin area for managing authentication settings.

— Names providers but no config details given `openlit-2.2.0`

06 **Self-hosted OpenAI-compatible model support** Not checked yet NEW 35

Support self-hosted OpenAI-compatible models, including configuration without a vault ID and provider setup.

— No config keys or exact setup steps given `openlit-2.2.0`

07 **Trace-level token totals and Hermes guidance** Not checked yet IMPROVED 35

Trace views now show trace-level token totals and system prompts for agent spans, alongside Hermes setup guidance.

— Names fields shown but no mechanism or path `openlit-2.2.0`

08 **Markdown and structured rendering for span content** Not checked yet IMPROVED 30

Span content now renders safe Markdown and structured values as chat bubbles or JSON trees.

— Brief description, no scope or example given `openlit-2.2.0`

WAS THIS USEFUL?  

Braintrust

Feedback ⓘ updated · Nov 20, 2026 NOTES ↗ RANK

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

> Braintrust's November 2026 release adds a Decisions API for LLM-as-judge scorers, a thread view for multi-trace sessions, a generally-available Terraform provider with new resources, and native tracing for LiveKit Agents and Pipecat voice pipelines — alongside a substantial set of breaking changes to both the Terraform provider and the voice-pipeline span schema.

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

01 LiveKit and Pipecat voice pipeline tracing

Not checked yet NEW

95

Python SDK v0.50.0 traces LiveKit Agents voice pipelines (STT-LLM-TTS, realtime models, speech-to-speech) and Pipecat pipelines as session spans with `user_turn` and `agent_turn` children, and captures caller or agent audio as attachments to the session span when `BRAINTRUST_CAPTURE_USER_AUDIO_ATTACHMENTS` or `BRAINTRUST_CAPTURE_AGENT_AUDIO_ATTACHMENTS` is set, with `braintrust[audio]` installable to compress recordings to Opus/Ogg.

Attach caller and agent audio to voice-agent session spans for a LiveKit pipeline, with Opus/Ogg compression.

```
$ export BRAINTRUST_CAPTURE_USER_AUDIO_ATTACHMENTS=1
export BRAINTRUST_CAPTURE_AGENT_AUDIO_ATTACHMENTS=1
pip install braintrust[audio]
```

— Exact env vars and install command provided snapshot-20261120

02 Terraform provider breaking changes

Not checked yet BREAKING

90

The provider now requires Terraform or OpenTofu 1.11.1 or later. `braintrust_ai_secret` now requires `secret_version` alongside `secret`, and `braintrust_env_var` now requires `value_version` alongside `value`; changing only the value no longer updates it, and references to `secret` or `value` return null. The `braintrust_org_automation` resource is removed — delete its blocks and run `terraform state rm` on each address before running `terraform plan`.

— Exact migration steps and config keys given snapshot-20261120

03 LiveKit Agents tracing breaking changes

Not checked yet BREAKING

90

LiveKit tracing now requires livekit-agents v1.8.9 to v1.8.10 (v0.51.0 extends this to v1.8.11+); other versions are not instrumented. Span names are renamed: `livekit_agent_session` → `session`, `user_speaking` → `user_speech`, `agent_speaking` → `agent_speech`, `stt_processing` → `stt`, `tts_request` → `tts`, `llm_request_run` → `llm_response`, and saved queries, filters, and scorers referencing old names must be updated. The TTS time-to-first-audio metric is renamed from `time_to_first_token` to `time_to_first_audio`, the `vad_endpointing` and `eou_detection` spans (and per-component `livekit_metrics` including end-of-utterance and silence-detection data) are removed, and audio now attaches to the session span instead of individual speech spans.

— Exact old/new span names and version range given `snapshot-20261120`

---

04 **Pipecat tracing breaking changes** Not checked yet BREAKING

90

Pipecat span names are renamed: `pipecat_pipeline` → `session`, `pipecat_llm_response` → `llm_response`, `stt_transcription` → `stt`, `tts_response` → `tts`, `user_speaking` → `user_speech`. `BraintrustPipecatObserver` is no longer exported — use setup_pipecat() or `wrap_pipeline_` instead. Audio is now recorded on the session span instead of as separate WAV or PCM files on individual spans, and token usage now appears on provider spans instead of `llm_response`, requiring the provider integration to be set up or auto_instrument() used.

— Exact renamed spans and replacement functions given `snapshot-20261120`

---

05 **Terraform provider reaches GA with new resources** Not checked yet

NEW

85

The Braintrust Terraform provider is now generally available at v1.0.0. It adds `braintrust_view` with `view_type = patterns` to manage saved Patterns views as code, the ephemeral `braintrust_service_token` resource to create service tokens without storing the key in Terraform state or plan files, and the `braintrust_users` data source to look up multiple workspace members in one request.

Manage a Braintrust Patterns saved view as Terraform infrastructure.

hcl

```hcl
resource "braintrust_view" "my_patterns_view" {
  view_type = "patterns"
  name      = "Latency regressions"
  project_id = var.project_id
}
```

— Names exact resources plus a runnable HCL example `snapshot-20261120`

06 Thread view for multi-trace sessions

Not checked yet · NEW

62

A new thread view reads multi-trace sessions in order by grouping traces on `metadata.conversation_id`, including traces that log the ID only on child spans, and nests tool calls under the request that started them.

— Mechanism and field named, but no UI path given `snapshot-20261120`

THINNER COVERAGE BELOW

07 Decisions API for LLM-as-judge scorers

Not checked yet · NEW

25

A new Decisions API lets LLM-as-judge scorers and classifiers pick a choice directly, returning confidence scores and per-choice probabilities.

— Named in summary only, no endpoint or usage shown `snapshot-20261120`

BREAKING ON UPGRADE

- ! The Terraform provider now requires Terraform or OpenTofu 1.11.1 or later.
- ! `braintrust_ai_secret` now requires `secret_version` alongside `secret`, and `braintrust_env_var` now 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 tracing in Python SDK v0.50.0 now requires livekit-agents v1.8.9 to v1.8.10 (v0.51.0 extends this to v1.8.11+); other versions are not instrumented.
- ! LiveKit span names are renamed: `livekit_agent_session` → `session`, `user_speaking` → `user_speech`, `agent_speaking` → `agent_speech`, `stt_processing` → `stt`, `tts_request` → `tts`, `llm_request_run` → `llm_response`; saved queries, filters, and scorers referencing old names must be updated.
- ! LiveKit TTS time-to-first-audio metric is renamed from `time_to_first_token` to `time_to_first_audio`.
- ! LiveKit `vad_endpointing` and `eou_detection` spans are removed; per-component `livekit_metrics` including end-of-utterance and silence-detection data are no longer recorded.
- ! LiveKit audio now attaches to the session span instead of individual speech spans.
- ! Pipecat span names are renamed: `pipecat_pipeline` → `session`, `pipecat_llm_response` → `llm_response`, `stt_transcription` → `stt`, `tts_response` → `tts`, `user_speaking` → `user_speech`.
- ! `BraintrustPipecatObserver` is no longer exported; use setup_pipecat() or `wrap_pipeline_` instead.

! Pipecat audio is now recorded on the session span instead of as separate WAV or PCM files on individual spans.

! Pipecat token usage is now on provider spans instead of `llm_response`; set up the provider integration or use auto_instrument().

WAS THIS USEFUL? 👍 👎

GOVERN

◆ AI Red Teaming & Testing

iFixAi

Feedback ⓘ v4.4.0 NOTES ↗ RANK

Ifixai is an independent Auditing of AI Agents. Run by human or the agent itself, to answer the most crucial question in the AI Agent Economy.

> iFixAi's v4.4.0 release adds a full command-line self-check pipeline for AI agents — audits, findings export, trace ingestion from five observability platforms, CI/CD and pre-commit wiring, and automatic MCP provisioning across 14 coding agents — alongside breaking changes to default models and the interactive run flow.

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

01 **Self-check command for agent endpoints** Not checked yet NEW

95

Adds `ifixai audit <agent-url>` to run a self-check against any OpenAI-shaped agent endpoint, with `--yes` to confirm and `IFIXAI_AGENT_KEY` for the agent's credential. Exit codes are 0 (nothing failed), 1 (check or pinned finding failed) and 2 (no verdict); `--judge-provider mock` runs only structural checks with no external API calls.

— full command with flags, env var and exit codes v4.4.0

02 **Pro audit offer and MCP provisioning** Not checked yet BREAKING

90

Adds `ifixai offer` to print the free Pro audit offer for a coding agent and `ifixai offer --add-server` to automatically add the MCP server `https://mcp.ifixai.ai/mcp` to every coding agent found on the machine, auto-provisioning it into Claude Code, Codex, Gemini CLI, `GitHub Copilot CLI`, Devin, Cline, Amp, `opencode`, Cursor, VS Code Copilot, Kiro, Junie, Warp and Antigravity when Add it for me is picked. Every real interactive `ifixai run` now opens with this Pro audit offer by default; pass `--no-promo` or set `IFIXAI_NO_PROMO=1` to restore the previous behaviour, and picking the free Pro audit ends the run before any call to your agent, exiting 0.

— default behaviour change with full flag/provisioning list v4.4.0

03 **Findings subcommands for self-check results** Not checked yet NEW

80

Adds `ifixai findings explain` to explain a self-check failure in plain language (what was asked, what the agent should have done, what it did, and the framework controls it maps to), `ifixai findings sarif` to export findings as SARIF 2.1.0,

`ifixai findings comment` to write a pull-request comment, and `ifixai findings validate-regressions` to validate the regressions file is well-formed.

— four named commands, thin mechanism on each v4.4.0

04 **Trace import from observability platforms** Not checked yet NEW

78

Import traces from OpenTelemetry GenAI, Langfuse, LangSmith, Arize/OpenInference and Braintrust exports with `ifixai traces import`; input over 64 MiB or 100,000 events is refused, not truncated.

— named command, five platforms and hard limits given v4.4.0

05 **CI/CD and pre-commit integration for self-check** Not checked yet

NEW

78

Integrates the self-check into pipelines: GitHub Actions via `actions/self-check`, which uploads SARIF to code scanning and can post or update a verdict comment; GitLab CI via `ci/gitlab/ifixai-self-check.gitlab-ci.yml`, which keeps the report and SARIF as artifacts and can post the verdict on the merge request; and a pre-commit hook via `.pre-commit-hooks.yaml` that checks the regressions file on commit and runs the self-check on push.

— named action, pipeline file and hook config, thin on internals v4.4.0

THINNER COVERAGE BELOW

06 **Default model changes for Gemini and Bedrock** Not checked yet

45

BREAKING

The default Gemini model changed from `gemini-2.0-flash` (now shut down) to `gemini-3.6-flash`, and the default Bedrock model changed to `global.anthropic.claude-sonnet-4-6`.

— named defaults but no migration guidance given v4.4.0

└──▷ **BREAKING ON UPGRADE**

! The default Gemini model changed from `gemini-2.0-flash` (now shut down) to `gemini-3.6-flash`.

! The default Bedrock model changed to `global.anthropic.claude-sonnet-4-6`.

! Every real interactive `ifixai run` now opens with a Pro audit offer; pass `--no-promo` or set `IFIXAI_NO_PROMO=1` to restore the previous behaviour. Picking free Pro audit ends the run before any call to your agent and exits 0.

WAS THIS USEFUL?  

◆ AI Runtime Defense & Guardrails

Prismor

Feedback ⓘ v1.78.0 NOTES ↗ RANK

Prismor is a self-hosted runtime control plane for AI agents. Observe or HITL approve or Block rogue tool calls before it executes: secret leaks, prompt injection, supply chain etc in a local dashboard. Agent agnostic (Claude, codex, langchain etc.)

> Prismor's two releases harden enforcement with fail-closed sandbox mode and expand its secrets/cloak layer with Vault/OpenBao, Infisical, and AWS Secrets Manager integrations, alongside new supply-chain and prompt-injection detection, per-session spend budgets, per-tool contracts, and Claude Code background-task capture.

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

01 **Secrets integrations and cloak layer hardening** Not checked yet

NEW

80

Integrates secrets from `Vault/OpenBao`, Infisical, and AWS Secrets Manager as cloak sources for credential injection. The cloak layer now scrubs encoded secret forms and binds secrets to their originating hosts, and org-vault secrets that the console seals can be synced to a specific device.

— Names three integrations and specific cloak behaviors, no config key shown. v1.77.0

02 **Context window panel limits and usage visibility** Not checked yet

IMPROVED

60

The context panel now limits source, model, and subagent switching and drains the usage spool to prevent stale state; per-session context window usage is also viewable across the dashboard, CLI, and API.

— Names dashboard, CLI, and API surfaces but no exact flags or commands. v1.78.0

v1.77.0

03 **Supply-chain and injection detection checks** Not checked yet NEW

60

Prismor now detects Unicode tag smuggling, performs version-level supply-chain age checks, flags registry 404 conditions, and enforces a minimal MCP stdio environment.

— Detection behaviors named but no way to invoke or configure them shown. v1.77.0

THINNER COVERAGE BELOW

04 **Fail-closed sandbox enforce mode** Not checked yet IMPROVED

50

Sandbox mode now fails closed in enforce mode, requiring a person to manually turn it off rather than allowing automated override.

— Explains the behavior change but no exact toggle or config key given. v1.78.0

05 **Per-tool contracts with narrow-only session label** Not checked yet

NEW

50

Apply per-tool contracts with a narrow-only session label using the new feat(tags) capability.

— Names the capability but gives no usage example. v1.77.0

06 **Claude Code background work capture** Not checked yet NEW

50

Prismor now captures Claude Code background work including task-notifications, failed calls, and background tool activity.

— Names the agent-specific capture scope but no config path. v1.77.0

07 **Per-session model and spend budgets** Not checked yet NEW

40

Set allowed models and spend budgets scoped per session, agent, user, or device.

— Describes scope only, no config key or command given. v1.77.0

08 **Semantic guard bypass-framing detection** Not checked yet IMPROVED

40

Prismor's semantic guard now distinguishes descriptions of people skipping checks from actual bypass framing.

— Explains behavior change but no mechanism or example given. v1.77.0

WAS THIS USEFUL?

◆ Model & Agent Protection

agent-safe-pipeline

Feedback ⓘ v0.6.1 NOTES ↗

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

└──▷ GET THIS VERSION ▾

npm

```
$ npm install -g @decionis/agent-safe-pipeline@0.6.1
```

git

```
$ git clone --branch v0.6.1 https://github.com/decionis/agent-safe-pipeline.git
# already have the repo? check out this version:
$ git checkout v0.6.1
```

signed build provenance · checked 20 Nov Wrong command?

NEW DETECTION Vendor bank-detail change exposure route

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Open-Weight Models

Alibaba Qwen Models

Feedback ⓘ updated · Nov 20, 2026

NOTES ↗

Qwen is Alibaba's language and multimodal model family, including coding, vision, image, speech, and reasoning models.

NEW SUPPORTED PLATFORM mindlab/macaron-v1.1-venti-pro available in Singapore region

WAS THIS USEFUL? 👍 👎

Xiaomi MiMo-V2.5-Pro Feedback ⓘ

context change · Nov 19, 2026 MODEL ↗ STABLE

MiMo-V2.5-Pro is a chat model from Xiaomi.

Context window shrank from 1024K to 262K tokens on OpenRouter (xiaomi/mimo-v2.5-pro), first seen 19 Nov 2026. Not checked yet

▾ 1 more release in this issue · 2026-11-19

> price increase · Nov 19, 2026 MODEL ↗ STABLE
>
> Price increase on OpenRouter (xiaomi/mimo-v2.5-pro): $0.435 → $0.48 in, $0.87 → $1.80 out per million tokens (+107%), first seen 19 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

◆ Frontier Models

Google Gemini API

Feedback ⓘ updated · Nov 20, 2026

NOTES ↗ RANK

Gemini API provides Gemini models for generating text, images, and other multimodal content in applications.

> Gemini API introduces on-demand tool schema retrieval, letting large tool sets be loaded lazily instead of sent in full on every request, with support for client-side retrieval logic and remote MCP servers.

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

01 **On-demand tool schema retrieval** Not checked yet NEW 88

Mark tools with `defer_loading: true` and add a `{"type": "tool_search"}` tool so Gemini retrieves only the schemas it needs instead of receiving the full tool set on every request. You can run your own retrieval logic instead of Gemini's by setting `execution: "client"` on the tool search entry, and remote MCP servers now also support `defer_loading` for the same on-demand behaviour. Tool retrieval works with Gemini 3.6 and later models, including all Gemini 4 models, via the Interactions API.

— Names exact flags and config values with version/model scope. snapshot-20261120

WAS THIS USEFUL? 👍 👎

OTHER / UNCATEGORIZED

◆ UNCATEGORIZED

Tokenrip

Feedback ⓘ changes since 2026-10-21 CODE ↗

Tokenrip's command-line client.

Tokenrip adds named sign-ins, job/output/feedback commands, task-queue ordering, schedules, table creation, and `rip auth logout` across versions 2.6.4–2.6.33

Add named sign-ins per agent with `rip auth add`, resolving `AMBIGUOUS_IDENTITY` on a shared computer where multiple agents share one account. Not checked yet

Add job, output, and feedback commands (`rip job`, `rip output`, `rip feedback`) along with job definitions and job delete. Not checked yet

Add `rip message` to replace `rip comment` for workspace threads. Not checked yet

Add `rip task list --waiting` to show tasks the caller is waiting on (`waiting=me`). Not checked yet

Add `rip task list --type` and `rip task list --step` to filter by free-tag type and step in task rounds. Not checked yet

+5 more

└─▷ BREAKING ON UPGRADE

! The `--audience`, `--membership`, and `folder share-contents` options for workspace access are removed (replaced by flat workspace access model in 2.6.24).

! `rip comment` is replaced by `rip message` for workspace threads as of 2.6.24.

! `billingUrl` is dropped from `DomainErrorDetails` as of 2.6.13.

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · November 20, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue