THE AI TOOLCHAIN

NO. 103

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED NOVEMBER 28, 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 413 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 83 of 86 tools

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

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

Three unrelated things stand out. Modal can now deploy any container image as an autoscaling HTTP server with no Python in the loop.

vim ? help

AutoRound's RRQ makes one checkpoint serve several bit-widths, so you stop re-quantizing per deployment target. Mastra moves LLM API keys into a KMS, write-only everywhere, so a leaked config stops being a billable incident.

DEPLOY

Ship a non-Python service on Modal and quantize a model once for every target

`modal server deploy` takes any container image and fronts it as a scalable HTTP server, removing the usual rewrite-into-a-Python-wrapper step for Go, Rust or Node services. AutoRound's Recurrent Residual Quantization produces a single checkpoint that serves multiple bit-widths, so a 4-bit edge build and an 8-bit server build no longer mean two tuning runs; llama.cpp adds budget-based quantization that can resume rather than restart after a long run dies.

Modal · AutoRound · llama.cpp

GOVERN

Keep model keys out of reach and run the vendor's stack in your own AWS account

Mastra Platform holds LLM API keys in a KMS and makes them write-only everywhere, so no console view, no log line and no support screenshot can leak the credential that bills you. Composio's BYOC mode runs its full service stack inside the customer's AWS account behind an approval-gated release pipeline, which is what data-residency reviews actually ask for; SAM binds policy to OIDC claims and attests them into mesh credentials so agent identity survives the hop between services.

Mastra · Composio · SAM

Does **Opus 5** read better? 👍 👎

OFFENSIVE SECURITY

◆ Exploitation & C2

xalgorix

Feedback ⓘ v4.6.212 NOTES ↗ RANK

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

> xalgorix added an opt-in environment variable to cut repeated system prompt token usage during long-running scans.

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

01 **Compact core prompt mode for long scans** Not checked yet NEW

90

Enabling `XALGORIX_SYSTEM_PROMPT_CORE_ONLY` (default off) sends the full methodology prompt only during an initial establishment window, then switches to a compact core prompt for subsequent requests to reduce per-request token usage. The full prompt is automatically re-injected when a turn produces no tool call.

— Names exact env var, default, and fallback behavior. v4.6.212

WAS THIS USEFUL? 👍 👎

BUILD

◆ AI Coding Agents

Warp

Feedback ⓘ changelog · Nov 28, 2026 CHANGELOG ↗ RANK

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

> Warp's major addition this window is Warp Factories, new infrastructure for defining agent-driven software workflows as code that connect to GitHub, Slack, Linear, and Jira, shipped alongside a cloud Automation Platform, a set of factory configuration guides (multi-repo, microservices, monorepo), a cross-factory Home page, cross-repository skill loading, cloud agent runtime improvements, and new inference models.

└──▷ WHAT SHIPPED · 17 FEATURES most completely described first ? what's the number?

01 **Monorepo factory scoped to one product** Not checked yet NEW

100

Scope a factory to one product in a monorepo via a product-specific intake label such as `factory-ready:web` in `automations/web-issue-labeled/automation.md`, filtering on `github` `issue_labeled` events. Define product directory, shared-package glob, owner teams, install commands, and escalation rules in `skills/monorepo-ownership/SKILL.md` so foreman, implement, and review agents share one scope. Require the implement agent to test only affected packages and their dependents using `pnpm --filter '...[origin/main]' test` and `pnpm --filter '...[origin/main]' run typecheck` before opening a pull request. Open pull requests that touch shared packages as drafts with review requested from the owning team in `CODEOWNERS`, blocking the factory from marking them ready. Bootstrap by pasting a prompt into New Chat or Factory MCP, replacing placeholders like `OWNER/REPO`, `PRODUCT_DIR`, `SHARED_PACKAGES_GLOB`, and `factory-ready:web`.

Scope a runner to install only the web product's workspace on a large monorepo, avoiding the 30-minute setup timeout.

```yaml
description: >-
  Runner for the web product only. node:22-bookworm provides
Node, corepack,
  and git; the setup installs the apps/web workspace and the
packages it
  depends on, not the whole monorepo.
setupCommands:
  - 'sudo corepack enable || echo "Warning: corepack enable
failed; pnpm may not be on PATH" >&2'
  - '(cd /workspace/monorepo && pnpm install --frozen-lockfile --
filter "{apps/web}...") || echo "Warning: apps/web workspace
install skipped" >&2'
instanceShape:
  vcpus:
  memoryGb:
  platform:
    os: linux
    arch: x86_64
    linux:
      dockerImage: node:22-bookworm
```

Trigger the monorepo factory only for approved web product issues, keeping other products and labels from accidentally starting a run.

```yaml
---
triggers:
  - provider: github
    event: issue_labeled
    filter:
      repos: [acme/monorepo]
      labels: ["factory-ready:web"]
---
A maintainer labeled an issue `factory-ready:web`: approved work
for the
`apps/web` product. Read the issue, scope it to that product as
the
`monorepo-ownership` skill describes, and drive it to a pull
request.
Post progress as comments on the issue.
```

Test all packages affected by the current diff — including dependents of any changed shared package — before the implement agent opens a pull request.

```
$ pnpm --filter "...[origin/main]" test
pnpm --filter "...[origin/main]" run typecheck
```

— Named skill file, config keys and runnable filter commands product docs

02 **GitHub issue intake triggers for Factories** Not checked yet NEW

95

Configure a `github` `issue_labeled` trigger with a readiness label in `labels` (e.g. `factory-ready`) inside `automations/issue-labeled/automation.md` to start factory work automatically when a maintainer labels an issue. Configure a `github` `issue_assigned` trigger in `automations/issue-assigned/automation.md` with `assignees`, `labels`, and optional `authors` filters to start work when an issue is assigned to the factory with its routing label present. Control bot-event handling with `skip_bots: self` to allow other bots through, or `skip_bots: none` to also admit Warp-owned bots. Bootstrap assignment-based intake by pasting a ready-made prompt into a factory foreman conversation or a coding agent connected through Factory MCP, replacing `OWNER/REPO` and `ALIAS`.

Start factory work whenever a maintainer applies your readiness label to an issue — suitable for bulk triage workflows.

yaml
```
triggers:
- provider: github
  event: issue_labeled
  filter:
    repos: [acme/api-service]
    labels: [factory-ready]
```

Start factory work only when an issue already carries the routing label and is then assigned to the factory account — prevents accidental runs from bare assignments.

yaml
```
triggers:
- provider: github
  event: issue_assigned
  filter:
    repos: [acme/api-service]
    assignees: [warp, warp-factory]
    labels: ["factory:Acme"]
```

— Exact config file paths, trigger keys and runnable example product docs

03 **Multi-repository factory workflows** Not checked yet NEW

95

Configure multi-repository factories in `factory.yaml` by listing each repo under `repositories` with `owner`, `name`, and `codeForge`, and setting a shared `runner` image that carries every toolchain. Trigger coordinated cross-repository runs by configuring `automations/issue-labeled/automation.md` to watch the `factory-ready` label across multiple repos. Define foreman, implementer, and reviewer roles in `agents/foreman/agent.md`, `agents/implementer/agent.md`, and `agents/reviewer/agent.md` to control per-repository pull request order, backward-compatibility rules, and cross-PR linking. The workflow produces linked pull requests titled `[1/2]` and `[2/2]` with a merge order a person controls — the factory never merges automatically.

Connect two repos to the same factory so every run checks out both and can open one pull request per repo.

yaml

```yaml
schemaVersion: v1alpha1
name: related-services
alias: Services
repositories:
  - owner: acme
    name: api-service
  - owner: acme
    name: webapp
agentDefaults:
  model: auto
runner: linux-services
```

Fire a factory run when a maintainer applies `factory-ready` to an issue in any of the connected repos.

```yaml
---
triggers:
  - provider: github
    event: issue_labeled
    filter:
      repos: [acme/api-service, acme/webapp]
      labels: [factory-ready]
---
A maintainer labeled an issue `factory-ready` in one of the two
related
repositories. Read the issue, decide which repositories change
and in which
order, and drive the change to one pull request per repository.
Post your
decision, progress, and the final merge order as comments on the
issue.
```

— Named config file, keys, agent files, and runnable YAML `product docs`

---

04 **Microservices factory for provider/consumer services**

Not checked yet NEW 95

Group related microservices under one factory by listing provider and consumer repositories in `factory.yaml` with `schemaVersion: v1alpha1` and an `agentDefaults.runner` pointing to a shared Linux image. Configure a shared runner in `runners/linux-services.yaml` using `setupCommands` with per-service dependency prefetch (e.g. `go mod download`, `npm ci`) and a `dockerImage` such as `warpdotdev/dev-go:1.23` or `warpdotdev/dev-full:latest`. Set up intake automation in `factory.yaml` under `triggers` on the `issue_labeled` GitHub event filtered by repo list and the `factory-ready` label. Define provider-first, additive-only contract enforcement and cross-service verification in `agents/foreman/agent.md`, `agents/implementer/agent.md`, and `agents/reviewer/agent.md`, bootstrapped from the `01-single-repo-quickstart` template.

Declare a provider and consumer repository pair in a microservices factory, so the foreman agent knows which service changes first and what runner to use.

```yaml
# factory.yaml
schemaVersion: v1alpha1
alias: Services
repositories:
  - owner: acme
    name: api-service
  - owner: acme
    name: webapp
agentDefaults:
  model: auto
  runner: linux-services
```

Trigger the factory automatically whenever a maintainer applies the `factory-ready` label to an issue in any service repository.

```yaml
# factory.yaml (triggers section)
triggers:
  - provider: github
    event: issue_labeled
    filter:
      repos: [acme/api-service, acme/webapp]
      labels: [factory-ready]
```

— Named config file, runner image and trigger filters `product docs`

---

05 **Slack intake trigger for Factories** `Not checked yet` NEW

93

Configure an `app_mention` trigger with a `channels` filter in `automations/slack-intake/automation.md` to route Slack mentions in a specific channel to a Warp Factory while ignoring mentions elsewhere. Set `slack.autoRespondToThreadReplies: false` in the Slack integration entry to require an explicit app mention on thread replies rather than auto-continuing the work item. Use `message_dm` as a separate Slack event to handle direct messages independently from channel-mention automations. A prompt pasted into the foreman chat or Factory MCP can configure the full Slack intake workflow automatically.

Route all @`-mentions` of your factory app in a designated intake channel into automated pull requests, while ignoring mentions in other channels.

```yaml
triggers:
- provider: slack
  event: app_mention
  filter:
    channels: [engineering-requests]
```

— Named config file, event and field with worked example `product docs`

---

06 **Home page aggregating items waiting on you** `Not checked yet` NEW

90

A new Home page in the Warp Factories web app aggregates items waiting on your attention — asked questions, spec reviews, and PR reviews — across all factories without visiting each factory's Runs page individually. The `Waiting on you` status filter, selected by default, shows items directed to you including GitHub PR reviews requested from you; `Active` , `Complete` , `Blocked` , or `Failed` filters browse your own runs, narrowed with the `Search your runs…` field grouped by `Today` , `This week` , and `All time` . When an agent proposes a change to a Warp-managed definition, a spec review appears under `Waiting on you` where you can comment on the diff, send feedback to the agent, or merge the change.

Triage all outstanding agent questions and PR reviews across your factories at the start of a work session without opening each factory separately.

> 📍 1. Open the Warp Factories web app and click Home in the sidebar. 2. The Waiting on you filter is selected by default — review each grouped work item. 3. For each item, click the row to open its conversation or details in the side pane. 4. After handling an item, click Mark as resolved on the row or in the side pane. Click Undo in the confirmation if you resolved it by mistake.

Review and act on pending spec approvals and pull request reviews without navigating into individual factories.

> 📍 1. Open Warp and select Home from the sidebar. 2. The view defaults to the `Waiting on you` filter — review listed questions, spec approvals, and pull request reviews. 3. For any agent-proposed change, open the spec review, comment on the diff, then select `send feedback` to return notes to the agent or select `merge` to accept the change.

— Exact UI filters and step-by-step navigation given `product docs`

---

07 **Cross-repository skill loading for Factories** `Not checked yet`

NEW

85

Factories can now load skills from other repositories through skill reference files: create a `.yaml` or `.yml` file in `agents/<agent-name>/skills/`, setting `source` to the repository and `include` to the skills to load. Pin a referenced skill to a specific version by setting `source` to a full commit SHA, keeping the skill and its supporting files stable across factory updates. Factory branch runs on Warp-managed Factories now load the branch's own skills instead of the production branch's skills.

— Exact directory path and config fields, no runnable example changelog-20261128-b539be6a

---

08 **Environment variable persistence for cloud agent runs**

Not checked yet NEW 80

Persist environment variables into a cloud agent run by writing `KEY=VALUE` lines to `$WARP_VARS_FILE` from setup commands.

Pass a secret token from a cloud agent setup command into the rest of the run without hardcoding it in the task definition.

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

— Names exact env var and gives a runnable shell example changelog-20261128-b539be6a

---

09 **Issue triage agent example** Not checked yet NEW 70

Use `agents/triage/agent.md` from the `02-sdlc-issue-to-pr` example as a starting-point triage agent definition that reproduces bugs, records findings on the tracked issue, and supports GitHub, Linear, and Jira issue sources.

Bootstrap issue triage for a GitHub repo by pasting this prompt into a new chat with your factory's foreman in the Warp Factories web app or via Factory MCP.

📍 Read `https://docs.warp.dev/factories/use-cases/issue-triage.md` and set up issue triage in my factory for OWNER/REPO. Use the triage agent in `https://github.com/warpdotdev/warp-factory-examples/tree/main/examples/02-sdlc-issue-to-pr` as the starting point, adapted to my issue source. - Check my existing automations first. If one already handles new or labeled issues in OWNER/REPO, update it or narrow its filters instead of adding a second one. - Add or update the triage skill with my categories, priority levels, the evidence required before naming a root cause, and when to ask a maintainer. - Configure one intake path for my issue source using the setup procedure. If I have not selected a path, ask me before configuring it. Do not add both creation and label intake for the same backlog. Route requests to the foreman using the triage-only prompt: return evidence, category, priority, and the next action, then stop. Never write a specification, dispatch implementation, approve, or merge. - Validate the definition. If it's managed as code, open a pull request to the definition repository; otherwise tell me what to change in the factory dashboard.

> Then guide me through the page's test plan with a fresh test issue outside my production backlog.

— Names the example file but detail stays brief product docs

10 **New inference model options** Not checked yet NEW 70

Select GPT-6.1 Terra for inference with effort-level model IDs `gpt-6-1-terra-low`, `gpt-6-1-terra-medium`, `gpt-6-1-terra-high`, and `gpt-6-1-terra-xhigh`. Select Gemini 4 Argon for inference using the model ID `gemini-4-argon`. Claude Fable 5.2 is also added as a model option.

— Exact model IDs given, selection path implied but not detailed product docs

11 **Ambient agent transport fallback and cloud sandbox launch**

Not checked yet NEW 70

Run agent workflows from the Warp CLI against a pre-warmed cloud sandbox via `CLI.Execute.Agent.RunCloudStandby`. Ambient agents now fall back to PTY transport when the sandbox has no ACP adapter, tracked as `AmbientAgents.ThirdPartyHarness.Transport.Fallback`. Agent mode now records V4A hunk matching outcomes via `AgentMode.Code.V4AMatchFinished`.

— Named internal event identifiers but no user-facing command product docs

THINNER COVERAGE BELOW

12 **Warp Factories: codified software-factory workflows**

Not checked yet NEW 55

Warp Factories now lets teams define development workflows as code, connect them to repositories and tools, and run agents on tasks such as triaging issues from Slack, GitHub, Linear, or Jira and opening pull requests. It is open infrastructure for defining multi-step software-factory workflows — for example a factory can investigate an issue, implement a fix, and return a pull request for review, with evals and benchmarks to improve quality and reduce cost — and now supports pull request review workflows where you define the review criteria and an agent returns findings for human decision.

— Describes concept and scope but no concrete config surface product docs

13 **Managed MCP proxy credential refresh and Git auth routing**

Not checked yet IMPROVED 55

Managed MCP server proxy credentials now refresh before they expire, keeping servers working through long Oz agent runs, while local Oz uses existing Git authentication and

routes GitHub API operations through managed MCP.

— Describes mechanism but no config surface to act on changelog-20261128-b539be6a

14 **Automation Platform for cloud-scheduled agent runs**

Not checked yet NEW 50

Warp adds an Automation Platform that runs agents in the cloud on schedules, connected tools, or API requests without requiring a laptop to stay on, so agents can investigate failed builds, review pull requests, or handle recurring maintenance. It also adds CLI, API, and SDK access to the platform so practitioners can incorporate agents into their own systems, configure environments and tool access, and inspect runs programmatically.

— Explains capability but names no endpoint, flag or command product docs

15 **Stable namespace preview hosting** Not checked yet IMPROVED 45

Namespace previews now serve development apps with a stable localhost Host and multiple app ports through one driver proxy.

— Mechanism named briefly but no config or command changelog-20261128-b539be6a

16 **Cross-agent waiting notifications** Not checked yet NEW 40

Native Warp agents now receive a notification when another run is waiting for their output, and third-party agents can wait for messages from other agents without marking their tasks complete.

— Behavior described only in prose, no named surface changelog-20261128-b539be6a

17 **Runner-prepared sandbox reuse** Not checked yet IMPROVED 35

Runs can now reuse runner-prepared sandboxes while preparing their own repositories and configuration after assignment.

— Thin one-line description with no mechanism detail changelog-20261128-b539be6a

└──▷ ALSO FROM THESE RELEASES

WAS THIS USEFUL?  

Cotool

Feedback ⓘ changelog · Nov 28, 2026 CHANGELOG ↗ RANK

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

> Cotool expanded its integration and log-ingestion footprint with nine new connectors and five managed log sources, added new AI model options including GPT-6.2 and Gemini 4 Neon, and took Signals and streaming detections to general availability alongside MCP-based detection authoring.

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

01 New integrations and managed log ingestion sources

Not checked yet NEW 55

Cotool adds managed log ingestion for Linear, Cursor, incident.io, Iru, and Generic JSON / OCSF data stored in Azure Blob Storage, plus new integrations for Cyberhaven, iVerify, UpGuard, Airtable, Raindrop, MarkMonitor, OneLogin, Thinkst Canary, and Cato Networks.

– Names all sources/integrations but no setup mechanism changelog-20261128-12cbd8ac

02 New AI model support and custom endpoints Not checked yet NEW

50

Cotool adds GPT-6.2 Astra, GPT-6.2 Sol, and Gemini 4 Neon model support, plus custom model endpoints and organization-selected fallback models.

– Names models and endpoints but no setup steps changelog-20261128-12cbd8ac

03 Expanded integration workflow actions Not checked yet IMPROVED

45

Expanded integration workflows now support filtered Notion database queries, team assignment for incident.io incidents, and SentinelOne passthrough alert routing.

– Names integrations and actions but no config detail changelog-20261128-12cbd8ac

04 MCP detection authoring actions Not checked yet NEW 40

Cotool MCP detection authoring now includes actions to create, test, and update detection code.

– Describes capability but no command syntax changelog-20261128-12cbd8ac

05 Billing visibility for managed log ingest Not checked yet NEW

30

Managed log ingest usage, allowances, and alerts are now visible in billing.

– Thin description of a billing UI addition changelog-20261128-12cbd8ac

06 **Alerts severity sorting and detection source details**

Not checked yet IMPROVED 30

Alerts now include severity sorting across pages and source integration and table details for code detections.

– Names UI features but no exact navigation changelog-20261128-12cbd8ac

07 **Signals and streaming detections generally available**

Not checked yet IMPROVED 25

Signals and streaming detections are now generally available for real-time detection workflows.

– GA status stated with no further mechanism changelog-20261128-12cbd8ac

08 **Guided first-run setup** Not checked yet NEW 25

Cotool adds a guided first-run setup to help new organizations connect integrations and complete initial security configuration.

– Describes onboarding flow without specifics changelog-20261128-12cbd8ac

09 **Faster Hunt results with more threat intelligence** Not checked yet

IMPROVED 25

Hunt now delivers lower-latency results with more threat intelligence while preserving existing custom coverage.

– States improvement with no measurable detail changelog-20261128-12cbd8ac

WAS THIS USEFUL? 👍 👎

Cognition Devin Desktop

Feedback ⓘ

changelog · Nov 28, 2026 | CHANGELOG ↗ | RANK

Devin Desktop is an agent-native code editor for developing software locally, remotely, and in containerized environments.

> Devin Desktop adds an in-app browser tab for viewing web pages, local dev servers, and Devin Cloud exposed ports, enforces MDM sign-in policy via system.json, and overhauls window tab management with split-pane and multi-select support, alongside composer skill invocation and a handful of smaller editor refinements.

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

01 **Window tab split-pane and multi-select management** Not checked yet

IMPROVED

95

Option-click (macOS) or Alt-click (Linux/Windows) a session link now opens that session in a split pane next to the current session; Shift-click or Cmd/Ctrl-click on title-bar tabs selects multiple tabs, lets you drag them together, and close them with `Cmd/Ctrl+W` or the tab menu; and the right-click menu of a window tab now includes `Copy URL` for pages with a web URL.

Open a session side-by-side with your current session to compare two parallel Devin Local workstreams.

> 📍 Option-click (macOS) or Alt-click (Linux/Windows) a session link in the sidebar to open it in a split pane next to the current session.

— Exact key combos and menu items with a runnable example `changelog-20261128-2dc6062c`

02 **MDM sign-in policy via system.json** Not checked yet NEW

85

Devin Desktop now enforces an MDM sign-in policy on macOS, Linux, and Windows using `system.json` with the keys `enterprise_host` and `account_id`, restricting authentication on managed devices to a specified enterprise host and account.

Enforce a corporate Devin Desktop sign-in policy so employees on managed devices can only authenticate to your enterprise host.

```json
In system.json, set:
{
  "enterprise_host": "devin.yourcompany.com",
  "account_id": "<your-account-id>"
}
```

— Exact config file and both keys given, with example changelog-20261128-2dc6062c

03 **Bottom terminal panel from any page** Not checked yet IMPROVED

70

The bottom terminal panel can now be opened from any page — including Devin Review, Wiki, and Settings — using `Cmd/Ctrl+J` or `Ctrl+\`.

Open the bottom terminal panel from anywhere in Devin Desktop, including Review, Wiki, or Settings pages.

> 📍 Press Cmd/Ctrl+J or Ctrl+ from any page to toggle the bottom terminal panel.

— Exact shortcut and named pages, runnable action changelog-20261128-2dc6062c

04 **Skill invocation anywhere in composer** Not checked yet IMPROVED

65

Typing `/` after other text in the composer in Devin Local sessions now lists skills, enabling skill invocation anywhere in a message rather than only at the start.

— Exact trigger character and behaviour described changelog-20261128-2dc6062c

05 **In-app browser tab** Not checked yet NEW

60

Adds an in-app browser (Browser tab) to open web pages, local dev servers, and Devin Cloud exposed ports alongside a session, with persistent cookies, element and area selection for chat, and agent browser control.

— Describes capabilities but no concrete command or exact UI path

changelog-20261128-2dc6062c

06 **macOS composer spellcheck and plain-text paste** Not checked yet

IMPROVED

60

The chat composer now underlines misspelled words and offers corrections on right-click, and `Cmd+Shift+V` now pastes plain text in the chat composer and other text fields.

— Names the exact shortcut but mechanism is brief changelog-20261128-2dc6062c

THINNER COVERAGE BELOW

07 **Files tab and breadcrumb respect files.exclude** Not checked yet

IMPROVED 55

Files tab and editor breadcrumb pickers in local sessions now hide files matched by `files.exclude`, matching IDE Explorer behaviour.

— Names config key and UI surfaces affected changelog-20261128-2dc6062c

---

08 **Shell code block 'Insert in terminal' button** Not checked yet

NEW 50

Shell code blocks in Devin Local chats now have an Insert in terminal button that types the command into the terminal prompt without running it.

— Names the button and behaviour but no command example changelog-20261128-2dc6062c

---

09 **Past Conversations search matches message text** Not checked yet

IMPROVED 45

Searching in the Past Conversations dropdown now matches message text in addition to session titles.

— Clear but minor, no further mechanism given changelog-20261128-2dc6062c

---

10 **Removed Windows Alt+Tab tab-switch bindings** Not checked yet

45

BREAKING

On Windows, the Alt+Tab and Alt+Shift+Tab default keybindings for Next tab and Previous tab have been removed.

— Names exact bindings removed but no replacement given changelog-20261128-2dc6062c

└──▷ ALSO FROM THESE RELEASES

.
.
.

└──▷ BREAKING ON UPGRADE

! On Windows, the Alt+Tab and Alt+Shift+Tab keyboard bindings for Next tab and Previous tab have been removed.

WAS THIS USEFUL? 👍 👎

yetone magpie

Feedback ⓘ v0.1.5004 CODE ↗ RANK

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

> Magpie's latest releases focus on finer-grained agent and MCP-server configuration — including per-server tool call timeouts, DeepSeek's native tool-search API, and LAN sharing control from the CLI — alongside smaller session, account, and UI improvements.

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

01 Per-MCP-server tool call timeouts

Not checked yet NEW

90

Set a per-server tool call timeout with `callTimeout` (seconds) in the Library MCP server config; magpie writes it into each connected agent's entry under that agent's own key — `timeout` (ms) for Claude, Gemini, OpenCode, Copilot, Cline, and Kiro; `toolTimeoutMs` for Kimi; `tool_timeout_sec` for Codex and Grok; and `timeout` (seconds) for Goose and Hermes — overriding the hub's default 50s wait for that server.

Give a slow rendering MCP server 15 minutes to complete a tool call instead of the hub's default 50 s, so long build steps don't time out across all connected agents.

```json
{
  "name": "render",
  "transport": "stdio",
  "command": "npx",
  "args": ["render-mcp"],
  "callTimeout": 900
}
```

— Names config key, per-agent keys, units, and default override v0.1.4988

02 LAN sharing control from CLI and live gateway following

Not checked yet NEW

85

Control local-network sharing from a terminal with `magpie lan on|off`, or check the current state with `magpie lan` alone — when `on`, other computers on the network can reach the gateway with a key; when `off`, it listens only on `127.0.0.1`. The gateway now also follows the `LAN` setting in `settings.json` live, moving its listener off `[::]` immediately when sharing is turned off from any other magpie process, the `magpie web` page, or a direct edit of `settings.json`, without requiring a restart.

Restrict a magpie gateway running on a shared server so only the local machine can reach it.

```
$ magpie lan off
```

Re-enable local-network sharing and confirm where the gateway is now listening.

```
$ magpie lan on && magpie lan
```

— Exact commands and config key with before/after behaviour v0.1.4983

---

03 **Untruncated and offset session reads in sessions show**

Not checked yet IMPROVED 80

Read full, untruncated session messages in `sessions show` by passing `--full`, which sets `Whole: true` on the transcript read and bypasses the per-message character limit that previously cut messages at `partMax` characters. Separately, `sessions show --json` now respects the `--from` flag, so JSON output starts at the specified message number rather than always beginning from the first message.

Retrieve the complete text of a long session without hitting the per-message truncation limit — useful when a search matched words past the 8 000-character cut-off.

```
$ magpie sessions show --full
```

Dump a session to JSON starting from a specific message, so downstream scripts only process the relevant tail of a conversation.

```
$ magpie sessions show --json --from 5
```

— Runnable commands and the internal flags they set v0.1.4985

---

04 **Bot identity for GitHub sync commits** Not checked yet NEW

80

Attribute GitHub sync commits to a bot identity rather than the token owner by setting `gitName` and `gitEmail` in the davsync config — defaults to `github-actions[bot]` / `41898282+github-actions[bot]@users.noreply.github.com` so sync commits no longer appear on the token owner's contribution graph.

Route GitHub sync commits to a GitHub App bot so they never count toward any personal contribution graph.

```json
{
  "gitName": "magpie-sync[bot]",
  "gitEmail": "123456+magpie-sync[bot]@users.noreply.github.com"
}
```

— Named config fields, defaults, and a runnable config example v0.1.4996

05 **DeepSeek native tool-search API support** Not checked yet NEW

75

magpie now routes tool-search calls for `deepseek-flash` (V4.1 Flash) through DeepSeek's own Responses API as the client sends them, using `defer_loading` to keep the cached prefix intact. Other DeepSeek models and third-party hosts still get tool search converted to a function call with found tools appended in `additional_tools`.

— Names model, API, and internal parameters but no user command v0.1.4977

06 **ChatGPT API account Codex usage windows in account list**

Not checked yet NEW

75

The `magpie accounts` list and editor account view (`LoginUsageShown`) now show a ChatGPT API account's matching Codex usage windows when the two share the same ChatGPT account and user IDs, matched on token claims rather than account name.

— Names command and matching mechanism, no runnable example given v0.1.4998

07 **Skill router unnamed-skill tracking and warning** Not checked yet

NEW

70

The skill router's description now names as many skills as fit within the 1024-character limit (measured in UTF-16 units via `descLen`), and counts the rest as `routerUnnamed` so the UI can warn about hidden skills. The library UI now shows a warning when an agent's `skillRouter` is `'on'` and `routerUnnamed > 0` , telling users how many skills are only discoverable after the router is opened.

— Mechanism and fields named but no direct user action v0.1.4984

THINNER COVERAGE BELOW

08 **Optional access key for Tencent, Volcengine, and Alibaba providers**

Not checked yet IMPROVED 50

The access key field for Tencent Token Plan, Volcengine, and Alibaba providers is now optional and housed under the Advanced section of the provider editor, so the Usage page can show plan quota windows without requiring the key for requests.

– Names providers and UI location but no exact steps v0.1.4979

09 **Keep computer awake during active tool runs** Not checked yet

IMPROVED 45

magpie now keeps the computer awake while an agent is running a tool, even after the last request ended past the sleep grace period (`awake.Grace`).

– Names the grace-period setting but no user action v0.1.5004

10 **Credit reporters named in commit bodies** Not checked yet NEW

45

magpie now credits reporters named in a commit body — via patterns like `Reported on X by @handle` or a `Reported-by:` trailer — on the contributors wall, alongside existing GitHub author and Co-authored-by credits.

– Names exact patterns but nothing for a reader to do v0.1.5004

11 **Dropdown menu for crowded phone tab strips** Not checked yet

IMPROVED 45

On phones, the Library's and Settings' tab strips now collapse into a single dropdown menu at the top of the page when there are more than five tabs, rather than wrapping into a multi-line secondary bar.

– Describes the UI behaviour change with a concrete threshold v0.1.5003

12 **Group agent model lists by maker** Not checked yet NEW

40

Agent model lists can now group models by maker via a `By maker` toggle and `makerGroups` , matching the provider editor's grouping.

– Thin one-line description with a named toggle v0.1.4989

13 **Group agy subagent conversations under parent** Not checked yet

IMPROVED 35

Agy subagent conversations are now filed under the parent conversation that started them, so session listings and deletes treat a conversation and all its subagents as one unit.

— Explains effect but no command or config to act on v0.1.5000

WAS THIS USEFUL? 👍 👎

keploy

Feedback ⓘ v3.7.30 NOTES ↗ RANK

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

> Keploy focused on recording reliability this window, adding disk-spill overflow protection and staged startup-mock serving to the mock manager, a dedicated rejection reason for warm-cache recording misses, and per-endpoint noise exclusion for test replay.

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

01 **WARM_RECORDING_UNMOCKED rejection reason** Not checked yet NEW

70

Adds the `WARM_RECORDING_UNMOCKED` rejection reason so read-only calls missed because recording started after in-memory caches had already warmed are reported separately, with a re-record step, instead of as ordinary failures.

— Names exact rejection reason and explains its trigger and fix v3.7.29

THINNER COVERAGE BELOW

02 **Disk-spill overflow handling for mock sync queue** Not checked yet

IMPROVED

50

Keploy's mock sync queue now spills to disk when its in-memory bound is exceeded during heavy recording, draining in order and keeping memory use bounded, so bursts of calls no longer cause mocks to be dropped.

— Clear mechanism described but no config key or flag named v3.7.30

03 **Staged reader for startup-time mocks** Not checked yet NEW

40

Keploy adds a staged reader for startup-time mocks in the mock manager, letting the enterprise agent serve an app's startup calls consistently across recording and replay.

— Explains purpose but no concrete usage detail v3.7.30

04 **Per-endpoint response field noise exclusion** Not checked yet

IMPROVED

40

Keploy lets users ignore a specific response field for a single endpoint during noise detection, so volatile fields don't cause false test failures only on the routes where they appear.

— Describes behavior but no named config key or command v3.7.26

WAS THIS USEFUL?  

jcode

Feedback ⓘ v1.1.6 NOTES ↗ RANK

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

> jcode's latest releases add `--provider` / `--model` flags for attaching to a shared server, new `/login`, `/subscribe`, and `/cloud` TUI starter commands, and refine goal-delivery tracking with colored indicators and smarter auto-poke follow-up, alongside inline image previews, idle status timing, and background task wake handling.

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

01 **Provider/model flags for shared-server attach** Not checked yet

NEW

70

jcode now applies `--provider` and `--model` flags when attaching to a running shared server session.

– Names exact flags but not full behavior scope v1.1.5

02 **New TUI starter commands** Not checked yet NEW

65

jcode adds a `/login` command offering starter provider choices (OpenAI, Anthropic, Gemini, API key, Auto Import), plus `/subscribe` and `/cloud` commands that surface contextual account nudges in the TUI.

– Names commands and provider options, no exact syntax shown v1.1.5

03 **Goal delivery tracking and auto-poke follow-up** Not checked yet

IMPROVED

65

jcode separates goal delivery outcomes from verification evidence with colored delivery indicators in pinned todo cards. Auto-poke follow-up drives from undelivered goal outcomes and re-arms after an Escape-pause on the next user turn, todo update, or 5-minute timeout, and keeps auto-poking partially-delivered and undelivered goals as new todo work completes across turns, stopping only when no new todo progress was made.

– Mechanism and timing detailed but no user-facing command v1.1.6 v1.1.3

THINNER COVERAGE BELOW

04 **Background task wake delivery and terminal status** Not checked yet

IMPROVED

40

jcode now delivers reload-interrupted background task wakes and surfaces terminal status in background tail and poke operations.

— Describes behavior but no interface detail v1.1.6

05 **Inline image previews on soft-interrupted messages**

Not checked yet NEW

40

jcode now shows inline image previews on soft-interrupted messages and flushes the paced stream buffer immediately on Done.

— Behavior named but no usage path given v1.1.6

06 **Self-dev build-and-live-reload default** Not checked yet IMPROVED

35

Self-dev prompts now default to build-and-live-reload after code changes.

— States default change without further mechanism v1.1.5

07 **MCP tools token count notice** Not checked yet IMPROVED

35

The MCP tools notice now shows an approximate token count for late-loaded tools.

— Names the notice but no numbers or thresholds given v1.1.3

08 **Last response age in idle status line** Not checked yet NEW

30

jcode shows the age of the last response in the idle status line after one minute of inactivity.

— Thin one-line behavior description v1.1.6

WAS THIS USEFUL?  

Graphify Labs Graphify

Feedback ⓘ v0.9.122 NOTES ↗

RANK

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

> Graphify adds a self-contained Canvas 2D renderer for visualising large graphs directly in `graph.html`, alongside an AST cache schema bump that triggers automatic re-extraction for several languages.

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

01 **Canvas 2D renderer for large graphs** Not checked yet NEW

83

`graph.html` now includes a self-contained Canvas 2D renderer that activates automatically above the vis-network threshold and within a configurable `--node-limit` or `GRAPHIFY_VIZ_NODE_LIMIT` environment variable. It supports viewport culling and sparse labels at low zoom, and the payload is base64-embedded so no external scripts are needed.

Set the node limit via environment variable for a CI pipeline where you can't pass CLI flags directly.

```
$ export GRAPHIFY_VIZ_NODE_LIMIT=5000
graphify .
```

— Names flag, env var, and mechanism; lacks performance numbers v0.9.121

THINNER COVERAGE BELOW

02 **AST cache schema bump to v57** Not checked yet BREAKING

50

The AST cache schema bumps from version 56 to 57; C#, TypeScript, Rust, COBOL, and Python graphs re-extract automatically on the first build after upgrading.

— Clear before/after but no migration steps needed v0.9.121

NEW SUPPORTED PLATFORM Shopify/Liquid theme extraction: templates/*.json, layout/theme.liquid

└──▷ **BREAKING ON UPGRADE**

! The AST cache schema bumps from version 56 to 57; C#, TypeScript, Rust, COBOL, and Python graphs re-extract automatically on the first build after upgrading.

v0.9.122 NOTES ↗ LATEST STABLE

Graphify adds Shopify/Liquid theme extraction, fixes incremental JS/TS re-extraction on resolver changes, and preserves parallel Cypher call sites between the same node pair.

└──▷ GET THIS VERSION ▾

pipx

```
$ pipx install graphifyy==0.9.122
# already installed? upgrade to this version:
$ pipx install --force graphifyy==0.9.122
```

git

```
$ git clone --branch v0.9.122 https://github.com/Graphify-Labs/graphify.git
# already have the repo? check out this version:
$ git checkout v0.9.122
```

PyPI attestation · checked 28 Nov Wrong command?

Graphify now extracts Shopify themes into the knowledge graph: the Liquid extractor reads `.liquid` templates and their `render` / `include` `/section` references, `templates/*.json` wires sections to blocks, and CSS classes, settings, custom elements, and metafields are linked — activated automatically when a theme root (`layout/theme.liquid`) is detected, leaving non-theme `.css` / `.json` files unaffected. Not checked yet

Graphify's `graphify extract` command now re-extracts JS/TS code when a resolver input such as a tsconfig alias changes, even when importer bytes are identical, matching full-rebuild edge accuracy. Not checked yet

Graphify's Cypher exporter now preserves parallel call sites between the same node pair instead of collapsing them to a single Neo4j relationship. Not checked yet

└──▷ BREAKING ON UPGRADE

! The AST cache schema bumps from version 57 to 58; TypeScript, C++, Markdown, and Shopify-theme inputs re-extract automatically on the first build after upgrading.

WAS THIS USEFUL? 👍 👎

Earendil Works Pi

Feedback ⓘ v1.7.0 NOTES ↗ RANK

Pi is an AI coding agent toolkit that provides a unified LLM API, agent loop, TUI interface and CLI.

> Pi v1.7.0 adds terminal session resume via the OSC 7577 protocol and a new `--agent-dir` flag, extends codemode image generation to Radius models and OpenRouter sizing controls, improves GitHub Copilot prompt-cache behavior, and removes install telemetry entirely.

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

01 Session resume via OSC 7577 protocol

Not checked yet NEW

91

Pi can resume a session after a terminal restart — including its original profile, tool restrictions, and instructions — via the Program Resume Protocol (`OSC 7577`), which reports `pi --session <id>` in terminals that set `OSC7577_TOKEN` . The resume command now repeats the restriction and resource flags Pi was started with — `--tools` , `--no-extensions` , `-e` , `--skill` , `--system-prompt` , and similar — so a terminal cannot resume a session with broader access than it originally had. A new `--agent-dir` flag lets Pi start in a project-specific profile directory so the resume command points to the right config path instead of the default `~/.pi/agent` .

Start Pi in a project-specific profile directory so the session-resume command issued to the terminal includes the right config path, not the default `~/.pi/agent` .

```
$ pi --agent-dir ~/profiles/my-project
```

Start Pi with a restricted tool set so the OSC 7577 resume token re-enforces those restrictions if the terminal relaunches the session.

```
$ pi --tools read,edit --agent-dir ~/profiles/my-project
```

— Names exact protocol, env var, flags, and runnable commands v1.7.0

02 Codemode image generation with Radius and OpenRouter

Not checked yet NEW

70

Codemode can now generate images using Radius image models, listed in the built-in catalog and discoverable via provider discovery. For OpenRouter image models, aspect ratio and size can be controlled by passing optional `{ aspectRatio, size }` as the third argument to models.generateImages().

— Names the API call and parameters but no example v1.7.0

03 **Install telemetry removed** Not checked yet BREAKING 70

Telemetry reporting is removed from Pi: the `/api/report-install` endpoint, the `enableInstallTelemetry` setting (SettingsManager.getEnableInstallTelemetry() / setEnableInstallTelemetry()), and the `PI_TELEMETRY` environment variable are all gone for new installs and updates.

— Names exact endpoint, setting, and env var removed v1.7.0

THINNER COVERAGE BELOW

04 **GitHub Copilot prompt cache improvements** Not checked yet IMPROVED 55

GitHub Copilot now preserves the prompt cache when changing thinking levels on supported Claude 5.x and GPT-6 models, and when tools change on supported Claude 5.x models; Copilot Claude models also gain cache warming support.

— Names models and behavior but no mechanism or usage path v1.7.0

└──▷ **BREAKING ON UPGRADE**

! Telemetry reporting is removed: the `/api/report-install` endpoint, the `enableInstallTelemetry` setting (SettingsManager.getEnableInstallTelemetry() / setEnableInstallTelemetry()), and the `PI_TELEMETRY` environment variable are all gone for new installs and updates.

WAS THIS USEFUL?  

Yuval Sarel cones

Feedback ⓘ v0.4.39 NOTES ↗ RANK

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

> Cones shipped bulk CLI operations with stdin piping, a new `cones place` command for organizing sessions into folders/tags/pinned, and dashboard improvements including per-worktree grouping and a scriptable session ID field.

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

01 Session place management via `cones place`

Not checked yet NEW

93

A session can be moved to a specific place — a folder, a `# tag`, or `pinned` — with `cones place ID set PLACE`, which replaces the previous `add` alias (still supported), and returned to its folder with `cones place ID remove`, optionally verifying the current place. The dashboard now lists each session in exactly one place (its folder, a `# tag` group, a tag inside its folder, or `pinned`), removing the `+N` count and duplicate `┆` markers, and sessions can be pinned/unpinned with `ctrl+t` or moved by dragging a row to a new location.

Pin a session by ID from the CLI so it surfaces at the top of the dashboard without opening the TUI.

```
$ cones place abc123 set pinned
```

Move a session into a named tag group, then return it to its folder when the work is done.

```
$ cones place abc123 set '# review'
cones place abc123 remove
```

— Named subcommands, flags and examples given v0.4.38

02 Bulk stop/rm with stdin piping

Not checked yet NEW

85

`cones stop` and `cones rm` now accept multiple target IDs in one call, e.g. `cones stop ID1 ID2 ID3`, and `cones rm` accepts piped input from `cones ls` via `cones rm -`, so sessions matching a filter can be removed in one step, e.g. `cones ls --dir ~/projects/myapp --status done | cones rm -`, instead of copying IDs manually.

Remove all finished agents in a project directory without copying IDs manually.

```
$ cones ls --dir ~/projects/myapp --status done | cones rm -
```

Stop several specific agents at once instead of calling `cones stop` once per ID.

```
$ cones stop ID1 ID2 ID3
```

— Runnable commands shown, mechanism and scope clear v0.4.39

THINNER COVERAGE BELOW

03 **Top-level `id` field in `cones ls --json`** Not checked yet

IMPROVED 55

`cones ls --json` output now includes the session ID at the top level as the `id` field, making it scriptable without digging into nested fields.

— Names exact field and flag but no usage example v0.4.39

04 **Per-worktree headings in dashboard** Not checked yet IMPROVED

55

Every linked worktree with a session now gets its own `⑂ NAME` heading inside its repository's folder in the dashboard, so worktree sessions are visually distinct from the parent repository's sessions.

— Describes UI change but no command or config to act on v0.4.37

WAS THIS USEFUL? 👍 👎

The Open Engine Zeroshot

Feedback ⓘ v17.3.0 NOTES ↗

RANK

Zeroshot orchestrates executor-verifier agent loops that implement and validate software changes while logging workflow state to SQLite.

> Zeroshot's v17.3.0 release adds file workspace support across the core and Python SDK, extends shared editors with publication and host actions, and bumps wire protocol versions in a way that breaks compatibility with older peers.

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

01 **Self-update failure for v10.4.0–10.8.0 installs** Not checked yet

80

BREAKING

Installations of versions 10.4.0 through 10.8.0 cannot self-update: `zeroshot update` fails with 'release archive contains unexpected entry restic' (or `restic.exe` on Windows). These must be upgraded manually with `npm install -g @the-open-engine-company/zeroshot@17.3.0`.

Install or upgrade to v17.3.0, required for installations on versions 10.4.0–10.8.0 where `zeroshot update` fails.

```
$ npm install -g @the-open-engine-company/zeroshot@17.3.0
```

— Exact error text and runnable upgrade command provided v17.3.0

02 **Managed filepath inputs with authorized access** Not checked yet

NEW

60

Zeroshot accepts managed filepath inputs and human answers from authorized paths or private uploads, with checkpoint preservation and Git exclusion of private copies.

— Behavior detailed but no command to invoke it v17.3.0

03 **Wire protocol gates advanced to v4/v5** Not checked yet BREAKING

60

Wire gates advance to `cluster/v4`, `target/v5`, and `hosted-runs/v4`, so incompatible peers on older protocol versions refuse to connect before workspace observation.

— Names exact gate versions but no migration command v17.3.0

THINNER COVERAGE BELOW

04 **Async file workflows in Python SDK** Not checked yet NEW

55

The Python SDK gains async file workflows, including workspace selection, filepath answers, and artifact retrieval.

— Names SDK capabilities but no sample code shown v17.3.0

05 **In-place and managed file workspaces** Not checked yet NEW

55

Zeroshot now supports local in-place and managed file workspaces, while preserving repository defaults, graph-only memory, and workspace recovery.

— Mechanism described but no setup steps given v17.3.0

06 **Shared editors and run views gain publications and host actions**

Not checked yet IMPROVED

25

Shared editors and run views are extended with publications and authorized host actions.

— Only a one-line description, no mechanism named v17.3.0

└──▷ **BREAKING ON UPGRADE**

! Wire gates advanced to `cluster/v4`, `target/v5`, and `hosted-runs/v4`: peers on older protocol versions will refuse to connect before workspace observation.

! Installations of versions 10.4.0 through 10.8.0 cannot self-update: `zeroshot update` fails with 'release archive contains unexpected entry restic' (or `restic.exe` on Windows) — these versions must be upgraded manually with `npm install -g @the-open-engine-company/zeroshot@17.3.0`.

WAS THIS USEFUL?  

Charm Crush

Feedback ⓘ v0.104.0 NOTES ↗ CODE ↗ RANK

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

> Crush shipped experimental client/server mode with new `crushrc` and JSON config options, a configurable core-utilities backend, mid-run model switching, live streaming of tool output, and OSC 7577 session resume, alongside onboarding and provider-dialog UI polish.

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

01 **Experimental client/server mode** Not checked yet NEW 90

Enable client/server mode as an experimental feature with `option experimental client-server true`, `option experimental server-ready-timeout`, `option experimental server-idle-timeout`, and `option experimental server-detach-grace` in `crushrc`, or with `client_server: true` under `options.experimental` in the JSON config. This lets a background server hold sessions across multiple terminal windows.

Run Crush with client/server mode active so a background server holds sessions across multiple terminal windows.

ini

```
# crushrc
option experimental client-server true
option experimental server-ready-timeout 30
option experimental server-idle-timeout 60
option experimental server-detach-grace 10
```

— Exact config keys and runnable crushrc example given v0.103.0

02 **Configurable core utilities backend** Not checked yet NEW 85

Control which core utility implementation Crush uses with `option coreutils turtle`, `option coreutils system`, `option coreutils auto`, or `option coreutils off` in `crushrc`, backed by the new `charm.land/turtle` implementation for consistent built-in tool behaviour regardless of what's installed on the system.

Pin Crush to the turtle core utilities on all platforms, ensuring consistent built-in tool behaviour regardless of what is installed on the system.

```ini
# crushrc
option coreutils turtle
```

— Named flags, package, and config example included v0.103.0

03 **OSC 7577 Program Resume Protocol support** Not checked yet NEW

60

Crush now supports the Program Resume Protocol (OSC 7577), allowing terminals such as Rex to restore Crush to the same session and project folder after a restart, crash, or reboot.

— Names protocol and terminal but no user action steps v0.103.0

THINNER COVERAGE BELOW

04 **Live streaming of tool call output** Not checked yet IMPROVED

55

Bash and Job tool call output now streams live into the chat while commands run, switching to the final result on completion.

— Describes behaviour change but no config or flag v0.103.0

05 **Mid-run model and reasoning switching** Not checked yet NEW

55

Switch the active model and reasoning effort while the agent is working, without waiting for the current run to finish.

— Clear capability but no exact UI path or command given v0.103.0

06 **Onboarding flow and provider dialog mouse support** Not checked yet IMPROVED

45

Crush now shows the onboarding flow for all new installations, even when an API key environment variable is already set — the matching key is used automatically without prompting for a paste. Mouse interaction now works in the provider list dialog: scroll, click to select, or double-click to choose a model, available in both the models dialog and onboarding screens.

— Behavioural UI changes described without config or command v0.104.0

WAS THIS USEFUL? 👍 👎

GitHub Copilot CLI

Feedback ⓘ v1.0.117 NOTES ↗ RANK

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

> GitHub Copilot CLI v1.0.117 adds custom CA certificate trust for sandboxed commands, a toggle to disable MCP tool caching, case-insensitive MCP server matching, explicit tool exposure for custom agents, and several refinements to Assisted Permissions, plan mode, background agents, and the model picker.

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

01 **Custom CA certificate trust for sandboxed commands**

Not checked yet NEW 81

Sandboxed commands can now trust custom CA certificates when traffic passes through a TLS-intercepting proxy, by setting `sandbox.userPolicy.network.extraCaCertificates` in user or managed settings to point to a certificate file such as `/etc/ssl/certs/corporate-ca.pem`.

Allow sandboxed commands to reach internal services behind a TLS-intercepting proxy by trusting a corporate CA certificate.

yaml

```yaml
sandbox:
  userPolicy:
    network:
      extraCaCertificates:
        - /etc/ssl/certs/corporate-ca.pem
```

— Names exact config key and gives working example v1.0.117

02 **Disable MCP tool cache persistence** Not checked yet NEW

73

Setting `COPILOT_MCP_TOOL_CACHE=false` keeps the MCP tool cache disabled across reloads, so Copilot neither reads nor writes tool snapshots.

Disable the MCP tool snapshot cache entirely so Copilot always fetches fresh tool definitions on every reload.

```
$ export COPILOT_MCP_TOOL_CACHE=false
copilot
```

— Names env var and shows exact usage v1.0.117

03 **MCP command handling improvements** Not checked yet IMPROVED

67

`/mcp` commands and `--disable-mcp-server` now match server names regardless of capitalisation, with exact matches taking precedence, and `/mcp search` now explains why a registry server cannot be installed instead of showing an endless spinner.

— Names flags and subcommands, no example v1.0.117

THINNER COVERAGE BELOW

04 **Assisted Permissions improvements** Not checked yet IMPROVED

49

(improved) Assisted Permissions now lets Autopilot retry an automatically denied action once if requested in a later chat message, with the retry still going through permission review, and in plan mode it asks before actions like creating a Git branch or installing packages instead of automatically approving them before you approve the plan.

— Explains mechanism and limit of one retry Describes before/after behaviour with examples of actions

v1.0.117

05 **Model picker shows retirement notices** Not checked yet IMPROVED

48

The `/model` picker now shows service notices and marks retiring models so you can review warnings before choosing.

— Names command, describes behaviour, no example v1.0.117

06 **Subagent reasoning effort visibility in /tasks** Not checked yet

IMPROVED

48

Subagent details in `/tasks` now show the reasoning effort the agent uses when the requested level is not supported.

— Names surface and exact behaviour change v1.0.117

07 **Explicit tool exposure for custom agents** Not checked yet NEW

37

Custom agents can now explicitly opt in to tools via a new tool-exposure setting in their agent file.

— Setting named but no key or syntax given v1.0.117

08 **Background agents resume sessions after restart** Not checked yet

NEW

35

Background agents can now continue their conversations after restarting the CLI and resuming a session.

— Describes behaviour but no mechanism or command v1.0.117

WAS THIS USEFUL? 👍 👎

gentle-ai

Feedback ⓘ v5.1.0 NOTES ↗ RANK

Gentle-AI configures the AI coding agents you already use: Claude Code, Cursor, OpenCode, Codex, Pi, and more. Choose persistent memory, Organic-Driven Development, curated skills, MCP servers, personas, and optional bounded review.

> gentle-ai v5.1.0 tightens the review-host contract, introducing lineage-bound selection with read-only inspection on rejection, and bumps the provider contract to 1.7.0 while explicitly blocking an incompatible OpenCode version.

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

01 **Lineage-bound review selection with read-only rejection view**

Not checked yet NEW 55

Review hosts can now use the provider-issued selection on the active review lineage, preserving the frozen range and serving the negotiated selection schema. A rejected targeted validation now exposes read-only inspection instead of implying acceptance. This accompanies an advance of the provider contract from `1.6.0` to `1.7.0`, so hosts should negotiate support rather than assume byte-identical compatibility with the v5.0.0 contract.

– Describes mechanism and contract version but no concrete API or command v5.1.0

02 **OpenCode 2.4.1 refused for incompatible task-dispatch path**

Not checked yet 45

BREAKING

gentle-ai explicitly refuses OpenCode 2.4.1 because its task-dispatch path is incompatible with the updated provider contract.

– Names exact version refused but no migration guidance given v5.1.0

└──▷ **BREAKING ON UPGRADE**

! The provider contract advances from `1.6.0` to `1.7.0`; hosts should negotiate support rather than assume the v5.0.0 contract is byte-identical.

! OpenCode 2.4.1 is explicitly refused for the incompatible task-dispatch path.

WAS THIS USEFUL? 👍 👎

mvschwarz OpenRig

Feedback ⓘ v0.7.24 NOTES ↗ RANK

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

> OpenRig's v0.7.24 release adds return-route conversation routing and scoped MCP tokens alongside Orca terminal workspace support, plus a string of smaller improvements to workflow commands, daemon routes, seat resume handling and the TUI.

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

01 **Return-route routing to originating conversation** Not checked yet

NEW

90

Team questions, seat prompts, and results can be routed back to your own Claude Code or Codex conversation by passing `--return-route auto` to `rig queue create`, or by setting `returnRoute` in the MCP server configuration.

Hand off work from your own Codex conversation and have team questions routed back to that thread automatically.

```
$ rig queue create --return-route auto
```

— Named flag, config key and runnable command given v0.7.24

02 **Scoped MCP tokens** Not checked yet NEW

85

Issue an MCP token restricted to specific tools using `rig mcp token add --allow <object>:<actions>`, useful for delegating to an agent that should not have full MCP access.

Issue a narrowly scoped MCP token that allows only specific tools — useful when delegating to an agent that should not have full MCP access.

```
$ rig mcp token add --allow queue:read --allow queue:create
```

— Exact flag syntax and runnable example included v0.7.24

03 **Workflow command improvements** Not checked yet IMPROVED

85

`rig workflow instantiate --operation-key <key>` now records what a workflow start created, which can be read back with `rig workflow operation`

`<key>`, and `rig workflow watch` now catches up on outcomes missed while reconnecting.

— Three named workflow subcommands with described mechanism v0.7.24

04 **Orca terminal workspace support** Not checked yet NEW 75

Open seats in an already-running Orca workspace on macOS with `rig terminal open <view> --provider orca`, instead of spawning new terminal windows.

Open your team's seats inside an already-running Orca workspace on macOS instead of spawning new terminal windows.

```
$ rig terminal open topology --provider orca
```

— Named flag and runnable command, scope is macOS-only v0.7.24

05 **Daemon route improvements** Not checked yet IMPROVED 65

Reading a daemon route's declared model and effort catalog is now also reported for community harness seats, and daemon routes now accept a `?host=<id>` query parameter to read one host's attention.

— Named query parameter but no usage example v0.7.24

06 **Codex seat sandbox reachability warnings** Not checked yet IMPROVED 60

`rig up`, `rig grow`, and `rig seat set-permissions` now surface warnings about Codex seat sandbox reachability before anything is refused.

— Named commands but no example of the warning itself v0.7.24

07 **Community harness switching with recorded handover**

Not checked yet NEW 60

A seat can be switched to an installed community harness and back; each switch starts a new recorded conversation with a handover so a later resume can reopen it.

— Mechanism described but no specific command shown v0.7.24

08 **Seat resume waits as awaitingDecision** Not checked yet IMPROVED 60

Seats that have neither saved startup context nor a resume token after a reboot now wait as `awaitingDecision` while the rest of their rig resumes, instead of blocking the whole rig.

— Named state and before/after but no reader action v0.7.24

THINNER COVERAGE BELOW

09 **TUI Topology view loads while daemon busy** Not checked yet

IMPROVED

50

The TUI's Topology view now opens while the daemon is busy, showing anything still loading as pending rather than appearing empty.

— Names the UI view but gives no navigation detail v0.7.24

10 **Roster notice prompts re-run of whoami** Not checked yet NEW

50

When OpenRig records a new team, a new roster notice prompts each agent to run `rig whoami` again, phrased as information so agents do not treat it as work.

— Named command but behavior is automatic, not reader-actionable v0.7.24

WAS THIS USEFUL?  

Alibaba Qwen Code

Feedback ⓘ v0.28.7 NOTES ↗ RANK

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

> Qwen Code's v0.28.7 release introduces a new Harness kernel CLI package, Claude Code mod compatibility, per-model effort persistence, and configurable worktree snapshot retention, alongside several Web Shell UI and control-plane session improvements, while removing built-in model-loop detection and tool-call caps.

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

01 Configurable worktree snapshot retention

Not checked yet NEW

86

Worktree snapshot retention is now configurable via `worktree.maxRetained`; setting it to `0` disables automatic sweeping and keeps all snapshots indefinitely.

Disable automatic worktree snapshot sweeping for a workspace where you want to keep all snapshots indefinitely.

```yaml
worktree:
  maxRetained: 0
```

— Exact config key with a runnable example value v0.28.7

02 Usage-statistics control for Harness kernel package

Not checked yet NEW

65

The new Harness kernel CLI package ships with a `privacy.usageStatisticsEnabled` config key to control usage-statistics collection, which is off by default.

— Names config key and default but little mechanism beyond that v0.28.7

03 Per-workspace management menu in Web Shell sidebar

Not checked yet NEW

65

Every locally trusted workspace in the Web Shell sidebar now has its own management menu for MCP, Skills, Extensions, Channels, and Settings, plus a Trust action for untrusted workspaces.

— Names all menu surfaces and a UI path v0.28.7

04 **Guided recovery from repeated tool-call rejections**

Not checked yet IMPROVED 65

Repeated-tool-call rejections now explain the cause and suggest `/compress` or `/clear` to recover, across TUI, headless, and daemon sessions.

– Names exact recovery commands and affected surfaces v0.28.7

THINNER COVERAGE BELOW

05 **Claude Code mod compatibility** Not checked yet NEW 57

Qwen Code now loads mods written for Claude Code 2.1.332, registering their commands and rendering their band in the UI; unrecognised hook names are skipped with a one-line diagnostic instead of failing.

– Describes behaviour and compatibility target but no user-facing command v0.28.7

06 **Optional arrow-key prompt history in Web Shell** Not checked yet

NEW 55

Web Shell adds a Settings → UI toggle, on by default, to make arrow-key prompt history optional; Ctrl+R still opens history search.

– Names toggle location and retained shortcut v0.28.7

07 **Managed Web Terminal ticket storage moved to SQL** Not checked yet

IMPROVED 53

Managed Web Terminal tickets are now stored in SQL so any control-plane instance can serve the WebSocket upgrade, while enforcing single-use and cluster-wide ticket caps.

– Infra mechanism explained but no user-facing action v0.28.7

08 **Faster cross-instance session event sync** Not checked yet IMPROVED

53

Session event streams now pick up events committed by other control-plane instances within a few hundred milliseconds, instead of waiting for the previous 5-second store re-read interval.

– Clear before/after numbers but no user action v0.28.7

09 **Removal of model-loop detection and tool-call caps**

Not checked yet 48

BREAKING

Qwen Code no longer detects model loops or caps tool calls; tool results and errors now reach the model unchanged on all surfaces.

– States the change but no migration guidance given v0.28.7

10 **Thinking level applied at Web Shell session startup**

Not checked yet IMPROVED 43

Web Shell now applies the selected thinking level during session startup so the first prompt uses it without requiring a second request.

— Describes behaviour change with no further mechanism v0.28.7

---

11 **Per-model effort level persistence** Not checked yet NEW

40

Qwen Code now persists the selected effort level per model; the OpenTUI effort picker lets you choose a level (e.g. medium) and confirms the selection by updating the status line.

— Only known from screenshots, no written mechanism in bullets v0.28.7

└──▷ **BREAKING ON UPGRADE**

! Qwen Code no longer detects model loops or caps tool calls; tool results and errors now reach the model unchanged on all surfaces.

WAS THIS USEFUL? 👍 👎

OpenQodex

Feedback ⓘ openqodex@0.26.0 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 expands the `reviewChange` result with new diagnostic fields and introduces a `nothing_to_review` status for changes that have nothing to review, requiring host updates.

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

01 **nothing_to_review status replaces complete for empty changes**

Not checked yet 75

BREAKING

`reviewChange` now returns a `nothing_to_review` status instead of `complete` when there is nothing for a reviewer to read — covering empty changes and binary-file, rename, or file-mode-only changes sent to a model reviewer. Hosts that branch on the `status` field must now handle `nothing_to_review` and treat it as skipped, not passed, rather than the previous `complete` value.

— Clear before/after and explicit migration instruction for hosts openqodex@0.26.0

02 **New diagnostic fields on reviewChange results** Not checked yet

NEW 67

The `reviewChange` result now carries `reasonCode`, `filesExamined`, `hunksExamined`, and `neighbourhood` fields, plus — on incomplete results — an `incomplete` object listing causes, retry eligibility, and unexamined units.

— Names exact new fields but no usage example of consuming them openqodex@0.26.0

└──▷ **BREAKING ON UPGRADE**

! Any host that branches on the `status` field of a `reviewChange` result must now handle the new `nothing_to_review` value and treat it as skipped, not passed — previously these cases returned `complete`.

WAS THIS USEFUL? 👍 👎

◆ AI App Builders

Lovable

Feedback ⓘ Nov 27, 2026 (2026-11-27) BLOG ↗ RANK

Lovable builds web apps and sites from natural-language prompts.

> Lovable's Nov 26 release adds four new third-party connectors spanning CMS, commerce, and scheduling, brings Claude Haiku 5.5 into its model lineup ahead of Haiku 4.5's retirement, and reorganizes the Connectors catalog with a unified Custom section and Enterprise retention-policy indicators.

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

01 Claude Haiku 5.5 added as Haiku 4.5 retires

Not checked yet

80

BREAKING

Adds Claude Haiku 5.5 (`anthropic/claude-haiku-5-5`) as an available model for in-app AI features, described as Anthropic's fastest model for high-volume classification, extraction, and routing at lower cost than Claude Haiku 4.5. Claude Haiku 4.5 retires on December 2, 2026 and its requests will be routed to Haiku 5.5, which rejects some settings Haiku 4.5 accepted (such as a custom temperature), causing those requests to fail.

— Gives model id, retirement date, and exact failure cause Nov 26, 2026 (2026-11-26)

02 New Cal.com, Ghost, Magento, and Squarespace Commerce connectors

Not checked yet NEW

70

Adds a Ghost connector so apps can read and publish posts and pages, manage tags and members, subscribe members to newsletters, and upload images to a Ghost site. Adds a Magento (Adobe Commerce) connector so apps can read and manage products, orders, customers, and stock of a Magento Open Source or Adobe Commerce store. Adds a Squarespace Commerce connector so apps can read orders, products, transactions, and customer contacts, and adjust stock levels of a Squarespace store. A Cal.com connector is also added to the catalog.

Connect a Squarespace Commerce store to a Lovable project to build an order dashboard or packing list app.

> 📍 In your Lovable project, go to Connectors, find Squarespace Commerce in the catalog, and add it — or type `connect Squarespace Commerce` in the Lovable chat.

— Names each connector's operations but Cal.com's capabilities aren't detailed

Nov 26, 2026 (2026-11-26)

THINNER COVERAGE BELOW

03 Retention-policy badge for blocked models (Enterprise)

Not checked yet IMPROVED 55

On Enterprise plans, the Available models list on the AI connector now marks each model an app cannot use with a Blocked by retention policy badge, which clears when workspace admins or owners enable Extended-retention models.

— Names the badge and the admin setting but no exact UI path Nov 26, 2026 (2026-11-26)

04 Custom section in Connectors catalog Not checked yet IMPROVED

55

Custom connectors, custom MCP servers, and any MCP registries added to a workspace now appear together in a Custom section of the Connectors catalog, rather than scattered through the full catalog.

Find all your custom connectors, custom MCP servers, and MCP registries in one place to manage them without searching the full catalog.

> 📍 Open Connectors in your workspace, then scroll to the Custom section to see all custom connectors, custom MCP servers, and added MCP registries listed together.

— Exact navigation given but mechanism is just a UI grouping Nov 26, 2026 (2026-11-26)

NEW INTEGRATION Contentstack connector

NEW INTEGRATION Webflow connector

└──▷ ALSO FROM THESE RELEASES

.
.
.

└──▷ BREAKING ON UPGRADE

! Claude Haiku 4.5 retires on December 2, 2026; requests for it will be routed to Haiku 5.5, which rejects some settings Haiku 4.5 accepts (such as a custom temperature), causing those requests to fail.

Nov 27, 2026 (2026-11-27) BLOG ↗ LATEST STABLE

Lovable adds IP groups for API key allowlists, two new in-app AI models, 301 permanent primary domain redirects, desktop auto-update, and a Bitbucket Reconnect flow.

Manage shared IP addresses for API key allowlists in one place with IP groups, available to workspace admins and owners on Business and Enterprise plans under Privacy & security in workspace settings. Not checked yet

Use `anthropic/claude-sonnet-5-5` (Claude Sonnet 5.5) in your app's AI features for fast everyday coding, data analysis, and multi-step tasks. Not checked yet

Use `openai/gpt-live-1.1` (GPT-Live 1.1) in your app's AI features for live voice conversations with more accurate transcription and steadier accent handling across languages than GPT-Live 1. Not checked yet

Primary domain redirects are now 301 permanent redirects, telling search engines the primary domain is the site's main address; data-carrying requests such as form submissions still get a 307 temporary redirect. Not checked yet

Lovable's desktop app now checks for updates regularly while open, starting with version 1.10.25, including when the window is restored from the menu bar or system tray. Not checked yet

+1 more

WAS THIS USEFUL?  

◆ Image & Video Generation

Higgsfield

Feedback ⓘ announced · Nov 23, 2026 BLOG ↗ RANK

Higgsfield is a generative video and image platform for creators, with a developer API.

> Higgsfield launched GPU Harness, a cloud GPU rental service controllable from Claude or ChatGPT via a `/gpu` slash command, offering dedicated machine types, an open model library with LoRA adapter support, one-click project presets, and six NVIDIA GPU tiers with per-hour pricing.

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

01 GPU rental via Claude/ChatGPT slash command

Not checked yet NEW

85

Start a cloud GPU rental directly from Claude or ChatGPT using the `/gpu` command, which opens a rental picker to choose GPU type, count, and approve pricing before billing begins. Connect Higgsfield by adding a custom MCP connector pointed at `https://mcp.higgsfield.ai/mcp` in Claude's Customize → Connectors settings, then type `/gpu` to launch the picker.

Connect Higgsfield to Claude and launch a GPU rental picker to start a Blender render job.

> 📍 In Claude, go to Customize → Connectors → + Add → Add custom connector. Name it Higgsfield, paste `https://mcp.higgsfield.ai/mcp`, click Continue, then Add, then Connect. In a new chat, type `/gpu` and select Rent a GPU to configure a Blender Studio machine.

— exact connector URL and command with step-by-step flow given `launch-20261123-a71cbe0d`

02 One-click slash-command presets for render projects

Not checked yet NEW

80

Use 13 one-click slash-command presets — `/destruction-studio`, `/exploded-view`, `/scene-builder`, `/cartoon-shaders`, `/use-after-effects`, `/illustration-animation`, `/localization-motion`, `/paper-collage`, `/presentation-animation`, `/saas-animation`, `/whiteboard-animation`, `/shot-composer`, and `/shot-cleanup` — to provision a matching Blender or EffectCraft machine pre-loaded with a ready project.

Provision a Blender destruction simulation with a single preset command, skipping manual machine setup.

```
$ /destruction-studio
```

— names all 13 presets as directly runnable commands `launch-20261123-a71cbe0d`

03 **Open model library on Compute machines** Not checked yet NEW

75

Run seven open models on Compute machines: MiniMax H3, Wan 2.2, Wan 2.1 I2V 480P, and LTX Video for video generation, plus Qwen Image, Z-Image Base, and Z-Image Turbo for images. Import up to four LoRA adapters per generation from Hugging Face or Civitai as `.safetensors` files up to 2 GiB.

— names all seven models and LoRA import limits `launch-20261123-a71cbe0d`

04 **Six NVIDIA GPU types with per-hour pricing** Not checked yet NEW

75

Rent six NVIDIA GPU types — L40S (48 GB, 71 credits/hr), RTX PRO 6000 (96 GB, 98 credits/hr), H100 (80 GB HBM3, 120 credits/hr), H200 (141 GB HBM3e, 134 credits/hr), B200 (180 GB HBM3e, 176 credits/hr), and B300 (288 GB HBM3e, 197 credits/hr) — with up to eight GPUs per machine.

— names all six GPU types with memory and pricing `launch-20261123-a71cbe0d`

THINNER COVERAGE BELOW

05 **Four dedicated GPU machine types** Not checked yet NEW

55

Rent four machine types — Blender Studio, ArtCraft, Desktop, and Compute — each pre-configured for rendering, motion graphics, real-time scenes, or running custom scripts and open models.

— names four machine types and purposes, no exact command `launch-20261123-a71cbe0d`

WAS THIS USEFUL? 👍 👎

AGENT

◆ Browser & Computer Use

agent-s

Feedback ⓘ v0.3.8 NOTES ↗ RANK

Agent S: the state-of-the-art open source computer use agents

> Agent S's macOS app gained richer multi-agent conversation controls — attachments, agent-to-agent task handoff, follow-up/turn management, and approval gating — alongside new OpenRouter spend and pricing visibility.

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

01 **OpenRouter spend tracking and model pricing** Not checked yet NEW

90

See OpenRouter input/output token rates, Sai credit costs, and billing details for local and account-backed models while choosing a model. Check per-session OpenRouter spend for the selected agent via Computer → Session and instructions, and check daily, weekly, monthly, and lifetime OpenRouter spend for a saved key via `Settings → Accounts → Check OpenRouter spend`.

Review how much OpenRouter credit an agent session has consumed before running further tasks.

> 📍 In the app, go to Computer → Session and instructions to see the OpenRouter spend Agent S has tracked for the selected agent.

Audit total OpenRouter spend across time periods to manage API key budgets.

> 📍 In the app, go to Settings → Accounts → Check OpenRouter spend to see daily, weekly, monthly, and lifetime spend for the saved key.

— Exact navigation paths and time breakdowns, runnable from the steps given v0.3.6

02 **Floating chat follow-ups and turn management** Not checked yet

IMPROVED

70

Send follow-up messages from the floating This Mac chat with new Take over, Resume, and Stop controls, and new Mac actions pause while you type. Queued follow-up messages now reach a live turn when the harness can accept them, and shared sandbox turns wait for confirmed cleanup before proceeding.

— Names exact controls and queuing/cleanup behaviour v0.3.8 v0.3.7

03 **Agent-to-agent task handoff via mentions** Not checked yet NEW

65

Mention another agent in the macOS app to send it a task or share a previous reply, with inline message review and control over whether a reply starts a new turn. Agent S also drafts the exact message to another agent from your request and conversation context, letting you review it before it is sent.

— Explains mechanism and review step but stays UI-level v0.3.8 v0.3.7

THINNER COVERAGE BELOW

04 **Image and file attachments in chat** Not checked yet NEW

57

Add images and files to sandbox, This Mac, and Sai chats using the plus button or drag and drop.

— Names the chat surfaces and interaction but little mechanism beyond that v0.3.8

05 **Approval controls for This Mac tasks** Not checked yet NEW

50

Choose Ask for approval or Auto-approve for the current or next This Mac task directly from the chat header.

— Clear UI path but no deeper mechanism described v0.3.8

06 **Grouped Passwords menu in Computer toolbar** Not checked yet

IMPROVED

45

The Computer toolbar now groups Chrome Password Manager and 1Password together in a single Passwords menu.

— Specific UI location and providers named, but thin change v0.3.7

07 **Compact tool activity display** Not checked yet IMPROVED

35

Tool activity now shows a compact list of steps with clearer titles and expandable details; press Shift–Return to add a line break to a draft.

— Minor UI tweak described briefly v0.3.8

08 **Expand and copy long prompts** Not checked yet NEW

25

Expand long prompts in chat and hover to copy the full prompt text.

— Bare one-line description with no further detail v0.3.6

WAS THIS USEFUL?

browser-use

Feedback ⓘ 1.0.9 NOTES ↗ RANK

The browser-use Python library automates web browsers for AI agents.

> browser-use 1.0.9 adds a rotate_proxy() method for rotating a cloud browser's exit IP mid-session and extends Claude prompt caching to OpenRouter and Bedrock routes, cutting redundant prompt re-billing.

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

01 **rotate_proxy() for cloud browser IP rotation** Not checked yet

NEW 86

The new rotate_proxy(country=None) method rotates the cloud browser's exit IP without losing open tabs, cookies, or logins; the agent calls it automatically at most once per site when a page blocks its IP, and a country code (e.g. `.ru`, `.de`, `.co.uk`, `.kr`) can be passed for country-specific domains. It now raises a clear error when the cloud reports no new exit IP is available, and any Browser(cdp_url=...) pointing to a `browser-use.com` host is recognised as a cloud browser.

— Names exact method signature, parameters, and detection logic 1.0.9

02 **Claude prompt caching on OpenRouter and Bedrock routes**

Not checked yet IMPROVED 77

Claude prompt caching now works on `openrouter:anthropic/...` routes, `bedrock:` Claude models, and ChatOpenAI(..., base_url='https://openrouter.ai/api/v1'), ending prior behaviour where every step re-billed the full prompt because `cache_read` was always 0.

— Names exact routes and before/after billing behaviour 1.0.9

WAS THIS USEFUL?  

◆ MCP Servers & Tooling

gram

Feedback ⓘ server@2.46.0 NOTES ↗ RANK

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

> Gram's latest releases add intent-aware assistance and audit trails around tool-call policy decisions, alongside dashboard, CLI and admin tooling improvements.

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

01 render-platform-mcp server URL configuration

Not checked yet

85

BREAKING

`render-platform-mcp` no longer falls back to a built-in host; the server URL must now be supplied via `--server-url` or the `GRAM_SERVER_URL` environment variable, e.g. `render-platform-mcp --server-url https://gram.example.com` or `GRAM_SERVER_URL=https://gram.example.com render-platform-mcp`.

Point a self-hosted `render-platform-mcp` instance at your own gram server instead of the removed built-in default.

```
$ render-platform-mcp --server-url https://gram.example.com
```

Supply the gram server URL via environment variable when running `render-platform-mcp` in a container or CI environment.

```
$ GRAM_SERVER_URL=https://gram.example.com render-platform-mcp
```

— Exact flag, env var and runnable commands provided. server@2.46.0

02 speakeasy whoami host reporting

Not checked yet IMPROVED

83

`speakeasy whoami --json` now reports which host it checked via the `api_url` field, and uses the host saved by `speakeasy auth` as the default when neither `--api-url` nor `SPEAKEASY_AI_API_URL` is set, instead of always falling back to `https://app.getgram.ai`.

Confirm which Speakeasy host your CLI is authenticated against, useful when working across multiple environments.

```
$ speakeasy whoami --json
```

— Named flags, env var, field and a runnable command. `cli@0.19.5`

---

03 **Intent-policy audit trail via gram streams** Not checked yet NEW

80

Audit intent-policy decisions by streaming every tool-call judgment as a `gram.intentpolicy.v1.Assessment` event via `gram streams`, written to the data lake as daily-partitioned Parquet under `gram.intentpolicy.v1.LakePrimary/`. Each row captures organization, project, user, session, tool call, guardrail version, evaluator, model, labels, contract outcomes, final decision, cost, latency, and timestamps; tool arguments are omitted and only their size is stored.

— Rich mechanism and schema detail, but no runnable command. `server@2.45.0`

---

04 **Intent-aware Project Assistant** Not checked yet NEW 77

A new `assistants.inferIntent` endpoint reads a short browser-side activity log to return the user's likely goal, a one-line opener, and up to three task-specific prompts, shown in the docked Project Assistant above its usual suggestions. Each Project Assistant message now carries the activity log and inferred goal in a `<dashboard_context>` block so follow-up questions are answered in context of the user's last action, enabled with the `assistant-activity-context` feature flag.

— Endpoint and feature flag named, but no usage example given. `server@2.46.0`

---

05 **Risk-policy evaluate grant server restriction** Not checked yet

60

BREAKING

A risk-policy evaluate grant that names a server via `server_url` or `server_identity` is now rejected on save; existing grants of this type must be removed or rewritten without a server target.

— Names exact fields but no migration command. `server@2.45.0`

---

06 **Secret backfills page in admin dashboard** Not checked yet NEW

60

Staff can preview and apply plugin marketplace token backfills and function runner key organization backfills from a new Secret backfills page in the admin dashboard, without needing database write access or the platform key locally.

— UI path named, but no API or config surface given. `admin@0.14.3`

THINNER COVERAGE BELOW

07 **Project homepage widgets** Not checked yet NEW 35

Adds project homepage widgets for MCP servers, risk findings and policies, device coverage and budgets, and skills and assistants.

— Only a list of widget names, no behavioral detail. dashboard@0.158.0

└──▷ **BREAKING ON UPGRADE**

! `render-platform-mcp` no longer falls back to a built-in host: a server URL must now be supplied via `--server-url` or `GRAM_SERVER_URL`.

! A risk-policy evaluate grant that names a server via `server_url` or `server_identity` is now rejected on save — existing grants of this type must be removed or rewritten without a server target.

WAS THIS USEFUL?  

hayhooks

Feedback ⓘ v2.5.0 NOTES ↗ RANK

Hayhooks is an easily deploy Haystack pipelines as REST APIs and MCP Tools.

> Hayhooks shipped a streaming terminal chat client for deployed Agents and pipelines, plus idempotent durable resume to make human-in-the-loop workflows safe against duplicate retries.

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

01 Idempotent durable resume with answer_key

Not checked yet

IMPROVED

93

Durable resume is now idempotent: passing `answer_key=...` and `answer_digest=...` to Resume() makes the same key-and-answer pair a no-op, while a different answer raises `ExecutionIdempotencyConflictError`. A new answer_digest() helper canonically hashes a resume input into a stable digest, and `ExecutionIdempotencyConflictError` is importable from both `hayhooks.durable.engine` and `hayhooks.durable.store`.

Resume a durable human-approval wait idempotently — a webhook or queue consumer can safely retry delivery without double-applying the answer.

python

```python
from hayhooks.durable.engine import Resume, answer_digest

approval = {"approved": True}

await execution.resume(
    Resume(
        answer_key="approval-ticket-42",
        answer_digest=answer_digest(approval),
        answer=approval,
    )
)
```

— Names exact API, helper, and error class with example v2.4.0

02 Terminal chat for deployed Agents

Not checked yet NEW

85

Run `hayhooks chat <name>` (installed via `pip install 'hayhooks[chat]'`) to start a streaming terminal chat session with a deployed Agent or pipeline. Pipeline wrappers can add `stream_agent_chat` to stream tool calls and results to the client, and tool cards can be expanded in the terminal to inspect arguments, results, and reasoning summaries when the provider supplies them.

Chat with a deployed Agent from your terminal to verify its responses and watch tool calls stream in real time.

```
$ pip install --upgrade "hayhooks[chat]==2.5.0"
hayhooks chat my-agent
```

— Exact command, install extra, and wrapper method named `v2.5.0`

WAS THIS USEFUL? 👍 👎

Model Context Protocol Servers

Feedback ⓘ v2.6.0

NOTES ↗ RANK

MCP Servers provides reference server implementations for connecting AI applications to external tools and data.

> MCP Servers v2.6.0 adds idle Streamable HTTP session eviction and Events extension support to server-everything, improves empty-result feedback in server-filesystem's search_files, and ships two breaking changes around package version ranking and git repository validation.

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

01 Package manager version ranking fix

Not checked yet BREAKING

85

Package managers rank `@modelcontextprotocol/server-memory` 2.2.2 and `@modelcontextprotocol/server-filesystem` 2.4.1 below every 2025.x and 2026.x date version, so they will not offer these as upgrades automatically; install explicitly with `npm install @modelcontextprotocol/server-memory@2.2.2` or `npm install @modelcontextprotocol/server-filesystem@2.4.1`.

— Exact install commands and version numbers given. v2.6.0

02 Events extension support in server-everything

Not checked yet NEW

75

`server-everything` now serves the Events extension (poll and push) on the `2026-07-28` protocol version, letting clients receive real-time server-side events over Streamable HTTP, e.g. via `npx @modelcontextprotocol/server-everything streamableHttp`.

Connect to server-everything's Events extension (poll and push) using the 2026-07-28 protocol to receive real-time server-side events in an MCP client.

```
$ npx @modelcontextprotocol/server-everything streamableHttp
```

— Runnable command and protocol version given. v2.6.0

THINNER COVERAGE BELOW

03 Improved search_files feedback in server-filesystem

Not checked yet IMPROVED

50

The `search_files` tool in `@modelcontextprotocol/server-filesystem` now reports what it searched and how to widen the pattern when results are empty.

— Behavior described but no example output shown. v2.6.0

---

04 **Git repository root validation in mcp-server-git** Not checked yet

45

BREAKING

A `--repository` root whose `.git` file names any other Git directory is now refused by `mcp-server-git` 2026.11.27.

— No migration step or example of the refusal shown. v2.6.0

---

05 **Idle session eviction in server-everything** Not checked yet

IMPROVED

35

`server-everything` now evicts idle Streamable HTTP sessions.

— No threshold, config flag, or example given. v2.6.0

└──▷ **BREAKING ON UPGRADE**

! Package managers rank `@modelcontextprotocol/server-memory` 2.2.2 and `@modelcontextprotocol/server-filesystem` 2.4.1 below every 2025.x and 2026.x date version, so they will not offer these as upgrades automatically — install explicitly with `npm install @modelcontextprotocol/server-memory@2.2.2` or `npm install @modelcontextprotocol/server-filesystem@2.4.1`.

! A `--repository` root whose `.git` file names any other Git directory is now refused by `mcp-server-git` 2026.11.27.

WAS THIS USEFUL? 👍 👎

context7 Feedback ⓘ @upstash/context7-mcp@4.9.0 NOTES ↗ RANK

Context7 maintains up-to-date documentation and context for AI agents through an MCP server.

> Context7 added support for serving teamspace Agent Skills from its hosted MCP server, alongside its built-in skill.

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

01 **Agent Skills support in hosted MCP server** Not checked yet NEW

50

The hosted HTTP server now serves Agent Skills installed in a teamspace via the Context7 dashboard over the MCP skills extension, alongside the built-in `context7-mcp` skill.

— Names mechanism and skill but no setup steps given @upstash/context7-mcp@4.9.0

WAS THIS USEFUL? 👍 👎

Composio

Feedback ⓘ announced · Nov 23, 2026 BLOG ↗ RANK

Composio connects AI agents to external tools and services with managed authentication and integrations.

> Composio's biggest shift this window is a BYOC (Bring Your Own Cloud) deployment mode that runs its full service stack inside a customer's own AWS account behind an approval-gated release pipeline, alongside a wave of toolkit docs adding custom OAuth credential support (Azure Monitor, HubSpot, Google Drive, Gmail), new Notion webhook ingress and API key auth, and Gmail session attachment sending plus trigger/webhook controls.

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

01 **Custom OAuth credentials for Azure Monitor Activity Log**

Not checked yet NEW 95

Connect Azure Monitor Activity Log using your own Microsoft Entra ID app registration by configuring Client id , Client secret , and Redirect URI in Composio's Auth Configs dashboard, setting the OAuth redirect URI to `https://backend.composio.dev/api/v1/auth-apps/add` in both the Entra app and the `oauth_redirect_uri` field. Grant the `user_impersonation` delegated permission under Azure Service Management in the Entra app's API permissions , supply the Azure Subscription ID as the `generic_id` field when connecting an account, and ensure the signed-in user holds `Microsoft.Insights/eventtypes/values/read` (e.g. via the Monitoring Reader role) on that subscription. The default auth config requests the `https://management.azure.com/user_impersonation` and `offline_access` scopes, with Azure RBAC controlling which subscription logs the account can read.

Register an Entra app and wire up custom OAuth credentials in Composio so your Azure Monitor Activity Log integration reads a specific subscription's logs without using a Composio-managed app.

📍 1. In the Azure portal, open App registrations › New registration. Set Supported account types to your tenant option, add Redirect URI (Web): `https://backend.composio.dev/api/v1/auth-apps/add` , then Register. 2. Under API permissions, add Azure Service Management › Delegated › user_impersonation. 3. Under Certificates & secrets, create a client secret and copy its Value. 4. In Composio dashboard: Auth Configs › Create › OAuth 2.0 › Your Own Credentials. Paste Application (client) ID into Client id and the secret into Client secret . 5. Click Connect Account, enter the Azure Subscription ID (generic_id), and sign in with an account that has the Monitoring Reader role on that subscription.

— Full Entra setup, permissions and scopes named end to end `product docs`

02 **Webhook ingress endpoint provisioning for Notion** Not checked yet

NEW 88

Set up a Notion webhook ingress endpoint for custom OAuth apps by calling composio.webhooks.endpoints.create(toolkit_slug='notion', client_id='YOUR_NOTION_OAUTH_CLIENT_ID'), then reading the verification token from endpoint.data.get('webhook_signing_secret') and pasting it into Notion to complete verification.

Provision a Notion webhook ingress endpoint when using your own Notion OAuth app, so Composio can receive Notion events.

python

```
from composio import Composio
composio = Composio()
endpoint = composio.webhooks.endpoints.create(
    toolkit_slug="notion",
    client_id="YOUR_NOTION_OAUTH_CLIENT_ID",
)
print(endpoint.root.id, endpoint.root.webhook_url)
```

Read the Notion verification token after Notion has called your ingress endpoint, so you can paste it back into Notion's verify field to complete setup.

python

```
from composio import Composio
composio = Composio()
endpoint = composio.webhooks.endpoints.get("we_123")
token = endpoint.data.get("webhook_signing_secret") if
endpoint.data is not None else None
if token is None:
    raise RuntimeError("Notion has not sent a verification token
yet")
print(token)
```

— Exact API calls and field names for setup `product docs`

03 **Custom OAuth credentials and scoped access for Google Drive**

Not checked yet NEW 86

Set up custom Google OAuth credentials for Google Drive by selecting Your Own Credentials in the Composio dashboard under Auth Configs, pasting your Client ID and Client secret, and setting the redirect URI to `https://backend.composio.dev/api/v1/auth-apps/add`. Control Drive access

scope by choosing narrower OAuth scopes such as `drive.metadata.readonly` (list metadata only) or `drive.file` (only files your app creates or the user opens) instead of the default full `https://www.googleapis.com/auth/drive` access.

Show your own app name on Google's consent screen and restrict Drive access to only files your app creates — useful for multi-tenant SaaS apps that need their own API quota.

> 📍 In the Composio dashboard, go to Auth Configs › Create, select Google Drive, choose OAuth 2.0 and Your Own Credentials, paste your Client ID and Client secret, set Redirect URI to `https://backend.composio.dev/api/v1/auth-apps/add`, and add the scope `https://www.googleapis.com/auth/drive.file` under Data Access.

— Exact dashboard steps and scope values given `product docs`

---

04 **Custom OAuth app support for HubSpot** `Not checked yet` NEW

85

Configure HubSpot OAuth scopes via `credentials.scopes` and `credentials.optional_scopes` fields in the Composio auth config to control which HubSpot permissions are required versus optional, manageable via the Create Auth Config API, Get Auth Config API, and Update Auth Config API. Bring your own HubSpot OAuth app by setting authentication type to OAuth2 and choosing Use your own developer authentication in the Composio dashboard under Auth Configs.

Create a custom HubSpot auth config via the API with minimal required scopes and tool-specific permissions as optional, so adding permissions later does not break existing connections.

```json
{
  "credentials": {
    "scopes": "oauth",
    "optional_scopes": "crm.objects.contacts.read
crm.objects.contacts.write crm.objects.companies.read
crm.objects.deals.read crm.objects.deals.write tickets timeline"
  }
}
```

— Named config fields and API endpoints for scope control `product docs`

---

05 **Gmail webhook ingress setup via Pub/Sub** `Not checked yet` NEW

81

Register your Pub/Sub webhook details with Composio so new Gmail triggers use webhooks instead of polling, by calling composio.webhooks.endpoints.update({ topic_name, push_service_account_email, push_audience }).

Register your Pub/Sub webhook details with Composio so new Gmail triggers use webhooks instead of polling.

```
$ composio.webhooks.endpoints.update({
  topic_name: 'projects/<project-id>/topics/<topic-id>',
  push_service_account_email: 'composio-gmail-push@<project-id>.iam.gserviceaccount.com',
  push_audience: '<webhook_url>'
})
```

— Exact API call with required fields named `product docs`

---

06 **Custom OAuth credentials and scoped access for Gmail**

`Not checked yet` NEW 75

Obtain a dedicated Google API quota by supplying your own OAuth app credentials instead of sharing Composio's managed app quota, configured via Auth Configs > Create > OAuth 2.0 > Your Own Credentials in the Composio dashboard, pasting Client ID and Client Secret with redirect URI `https://backend.composio.dev/api/v1/auth-apps/add`. Use the `gmail.readonly` scope to read messages and settings without granting send or modify access.

Use Auth Configs in the Composio dashboard to supply your own Google OAuth credentials so your app name appears on the consent screen and you hold a dedicated API quota.

> 📍 In the Composio dashboard: 1. Go to Auth Configs › Create and select Gmail. 2. Enter a config name and click Next. 3. Choose OAuth 2.0 › Your Own Credentials. 4. Paste your Client ID and Client Secret. 5. Leave Redirect URI as `https://backend.composio.dev/api/v1/auth-apps/add`. 6. Select the scopes your tools need, then click Create Auth Config.

— Dashboard steps and scope name given `product docs`

---

07 **Session-based email attachment sending in Gmail** `Not checked yet`

NEW 73

Send email attachments through a session by uploading a file with the experimental `session.experimental.files` API, then instructing the agent to pass the sandbox path using get_mount_file_s3_key() with `mimetype` and `s3key` fields to the discovered email tool.

— Named experimental API and fields, no full example `product docs`

---

08 **API key auth and connection management for Notion** `Not checked yet`

NEW 67

Connect Notion with an API key (internal integration secret starting with `ntn_` or `secret_`) by creating an auth config with the API Key auth type, skipping the OAuth redirect flow entirely. To avoid token invalidation in production, keep one active Notion connection per user per Notion app, since reconnecting the same user to the same Notion app causes Notion to issue a new `access_token` and `refresh_token` pair and invalidate the older `refresh_token`.

Connect a Notion internal integration using an API key (no OAuth redirect) when you only need to access your own workspace.

> 📍 In the Composio dashboard: 1. Go to Auth Configs › Create Auth Config › Notion 2. Set authentication type to API Key 3. Click Connect Account, paste your integration secret (starting with `ntn_` or `secret_`) 4. Confirm the account shows as Active

— Named auth type and token behaviour with concrete steps `product docs`

---

09 **BYOC deployment in your own AWS account** `Not checked yet` NEW

65

Composio now offers a BYOC (Bring Your Own Cloud) deployment mode that runs Composio's services — API, SDKs, tool execution, connections, sessions, MCP, semantic tool search, and Triggers — inside your own AWS account (Amazon EKS with AWS Fargate, in a dedicated AWS sub-account), so credentials and operational data never leave your environment.

— Describes architecture and scope but no setup steps `launch-20261123-12a2c05d`

---

10 **Approval-gated release pipeline for BYOC** `Not checked yet` NEW

60

Every release in the BYOC model goes through an approval pipeline in your account: Composio requests a specific release, your team reviews a preview of infrastructure changes, and a team member must approve before the pipeline applies changes, runs database migrations, and rolls out services.

— Clear process described, no commands given `launch-20261123-12a2c05d`

THINNER COVERAGE BELOW

---

11 **Gmail trigger diagnostics and polling control** `Not checked yet`

NEW

55

List a Gmail trigger and read its `type` and `fallback_reason` fields to diagnose whether it is polling or using webhooks, and why. Pass `type: 'poll'` when creating a Gmail trigger to explicitly request polling instead of webhooks.

— Named fields and flag but no full workflow shown `product docs`

---

12 **BYOC privacy and logging controls** Not checked yet NEW

55

Workflow inputs, results, and error details sent to Temporal Cloud are encrypted before leaving your environment, with encryption keys staying inside your account. Each project can keep tool request and response bodies out of its logs, and license/instance status reporting to Composio can be turned off. Tool search in BYOC can run its required LLM and embedding model inside your cloud.

— Lists controls but no config keys or toggle names launch-20261123-12a2c05d

13 **HubSpot tool limits and webhook requirements documented**

Not checked yet IMPROVED

45

`HUBSPOT_SEARCH_CONTACTS_BY_CRITERIA` and `HUBSPOT_LIST_CONTACTS_PAGE` now document a maximum `limit` of 100 results per request, and HubSpot webhook triggers now document that `app_id` and a developer API key are required fields for each user's own HubSpot app to receive webhook delivery.

— Named tools and fields but only a thin doc note product docs

WAS THIS USEFUL? 👍 👎

◆ AI Agent Frameworks

CopilotKit OpenBot

Feedback ⓘ v1.2.2 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's two releases make LangGraph the default agent engine with an optional embedded in-container deployment, remove step and hand-back limits on long-running turns in favor of token-window context management, and add multi-channel human-input notifications, a self-service account deletion API, and tunable database connection pools.

└──▷ **WHAT SHIPPED · 16 FEATURES** most completely described first ? what's the number?

01 **LangGraph engine default with embedded container deployment**

Not checked yet 95

BREAKING

The LangGraph TypeScript engine is now enabled by default for all new Bots (revert with `OPENBOT_DEFAULT_BOT_ENGINE=built-in` in `config.extraEnv` or `engine.langgraphTs.enabled=false` on Kubernetes). It can run embedded beside the server in one container on port 4215 (loopback only), disabled with `EMBEDDED_ENGINE=off` and pointed elsewhere via `LANGGRAPH_TS_AG_UI_URL`. New Helm values `engine.langgraphTs.enabled`, `managed-agent-token`, and `model-gateway-token` (each generated with `openssl rand -hex 32`) are required on upgrade — `helm upgrade` names any missing keys before changing anything — while the chart no longer makes or reads `agent-tool-token`, so `secrets.agentToolToken` is ignored.

Disable the embedded LangGraph engine and fall back to the built-in agent when the container environment cannot run Node alongside the server.

```text
EMBEDDED_ENGINE=off
LANGGRAPH_TS_AG_UI_URL=http://langgraph-engine:4215
OPENBOT_DEFAULT_BOT_ENGINE=built-in
```

Upgrade a Kubernetes install that uses an external secret store by adding the two required engine tokens before running helm upgrade.

```
$ kubectl create secret generic openbot-engine-tokens \
  --from-literal=managed-agent-token=$(openssl rand -hex 32) \
  --from-literal=model-gateway-token=$(openssl rand -hex 32)
helm upgrade openbot openbot/openbot \
  --reuse-values \
  --set engine.langgraphTs.enabled=true
```

— Names env vars, Helm keys, port and exact upgrade commands v1.2.1

02 **Self-service account deletion API** Not checked yet NEW

90

New `getAccountDeletion`, `requestAccountDeletion`, and `cancelAccountDeletion` endpoints under `/api/v2/people/me/account-deletion` let a user request, check, or cancel deletion of their own account, surfacing problem types `account-deletion-blocked`, `reauthenticate`, `account-deletion-not-confirmed`, `account-deletion-under-way`, and `account-pending-deletion`. Only session or app sign-in works; API keys are refused.

Request account deletion programmatically when automating offboarding workflows — returns current deletion status or surfaces the reason it is blocked.

```
$ curl -X POST
https://openbot.example.com/api/v2/people/me/account-deletion \
  -H 'Authorization: Bearer <session-token>' \
  -H 'Content-Type: application/json' \
  -d '{"confirmation": "delete my account"}'
```

— Named endpoints, path and problem types with a runnable example v1.2.1

03 **Tunable database connection pool settings** Not checked yet NEW

90

Pool size and idle timeout can now be controlled with `database.pool.max`, `database.pool.systemMax`, and `database.pool.idleTimeoutSeconds` on Kubernetes, or `OPENBOT_DATABASE_POOL_MAX`, `OPENBOT_DATABASE_SYSTEM_POOL_MAX`, and `OPENBOT_DATABASE_IDLE_SECONDS` elsewhere; a value of `0` keeps an idle connection open indefinitely.

Cap connections to a shared database and reclaim idle ones quickly when running OpenBot alongside other services on the same host.

```yaml
database:
  pool:
    max: 5
    systemMax: 2
    idleTimeoutSeconds: 30
```

— Exact config keys, env vars and a runnable example v1.2.2

04 **Unlimited long-running turns with token-window context management**

Not checked yet IMPROVED 85

Long turns now run without a step or hand-back ceiling — the previous 48 hand-back limit, 40-step LangGraph (TypeScript) budget, and 25-step LangGraph (Python) limit are gone. Context across a long run is instead kept inside the model's token window via the new `recall_earlier_results` tool, automatic clearing of old results past 40,000 tokens, Google ADK summarisation past 100,000 prompt tokens (keeping the newest 20 events), and Claude Agent SDK trimming (keeping the newest 10 results).

— Names the tool and exact token/event thresholds per engine v1.2.2

05 **Mid-turn human input notifications across channels**

Not checked yet NEW 80

Users are notified on phone, browser, Slack, Teams, and SMS when a Bot needs human input mid-turn — sign-in, secret entry, or takeover — and the server carries the turn on once they respond. Browser push now reaches Google (`fcm.googleapis.com`), Mozilla (`*.push.services.mozilla.com`), Apple (`*.push.apple.com`), and Microsoft (`*.notify.windows.com`) with no required configuration, since the server auto-generates its VAPID key pair on first request (a fixed pair can be supplied via `webPushVapidPublicKey` / `webPushVapidPrivateKey` and `webPushSubject`). The carried turn is surfaced as `turn` in a conversation's work stream for apps sending `OpenBot-Capabilities: conversation-work-turn` .

Enable browser push notifications across multiple database replicas by supplying a fixed VAPID key pair and a contact address.

```yaml
secrets:
  webPushVapidPublicKey: "<your-public-key>"
  webPushVapidPrivateKey: "<your-private-key>"
config:
  webPushSubject: "mailto:ops@example.com"
```

— Names push services, VAPID config and capability header v1.2.2

06 **Per-member General Assistants by default** Not checked yet BREAKING

75

Every member now gets their own General Assistant by default instead of one shared per organization; the organization's existing shared assistant is renamed or reassigned on first open after upgrade. Set `config.ownGeneralAssistant: false` on Kubernetes or `OPENBOT_OWN_GENERAL_ASSISTANT=false` elsewhere before upgrading to keep the previous shared-only behaviour.

Opt out of per-member General Assistants before upgrading to keep a single shared assistant for the whole organization.

```yaml
config:
  ownGeneralAssistant: false
```

— Named config key/env var with clear pre-upgrade action v1.2.2

07 **Deduplicated computer step execution** Not checked yet IMPROVED

75

Concurrent computer step execution is now deduplicated so each step runs exactly once. Apps sending `OpenBot-Capabilities: computer-step-taken` receive `taken`, `COMPUTER_STEP_RUNNING`, or `COMPUTER_STEP_OUTCOME_UNKNOWN` statuses, and a direct computer call can name its `threadId` and `toolCallId`.

— Names header, status values and call parameters v1.2.2

THINNER COVERAGE BELOW

08 **Verified managed-update chart signatures** Not checked yet NEW

55

Managed-update chart signatures are now verified by default using OpenBot's release key shipped in the `openbot-agent` chart. Set `product.verify.enabled: false` on a mirror whose charts carry no signature.

— Names the config key and exact opt-out condition v1.2.1

09 **Conversation run status and approval listing APIs** Not checked yet

NEW

50

The new `getConversationRun` API retrieves how a Bot run ended, so a finished answer shows complete even when its end never reached the client, and the new `listConversationApprovals` API lists answered approvals in any conversation.

— Names both endpoints but no usage example given v1.2.1

10 **Gemini requires API key, OAuth sign-in removed** Not checked yet

50

BREAKING

Gemini now connects with a Gemini API key only; Google sign-in for Gemini is removed and every saved Google sign-in for Gemini is deleted at upgrade. `config.googleModelSignIn` and `secrets.googleModelOAuthClientSecret` are no longer read on Kubernetes.

— Names removed config keys, no reconfiguration steps given v1.2.1

11 **AG-UI app tool naming change** Not checked yet BREAKING

45

AG-UI app tools are now exposed as `<app>__<tool>` instead of `mcp__<app>__<tool>` for providers that reject the old prefix; calls under the old name still work, though old-name storage and display are unchanged.

— Names old and new tool naming scheme, no migration steps v1.2.1

12 **System prompt delivered as quoted conversation** Not checked yet

45

BREAKING

Connected Bots no longer receive `forwardedProps.openbotSystemPrompt`; `system` or `developer` messages are now delivered as quoted conversation rather than as a role-level system message.

— Names the removed field but no migration guidance v1.2.1

13 **Bot instructions up to 5000 characters** Not checked yet IMPROVED

40

Bot instructions now accept up to 5000 characters, up from 1000, across the New Bot dialog, the Setup page, the starter Bot, and the API.

— Gives exact before/after limit and affected surfaces v1.2.1

14 **Unbiased AI connections stop running** Not checked yet BREAKING

35

An AI connection saved for Unbiased AI stops running after upgrade — where it is the organization's or a person's default AI, Bots on it stop before they start. Connect another AI provider before upgrading.

— States the impact and pre-upgrade action, no mechanism detail v1.2.2

---

15 **ChatGPT plan sign-in requires re-authentication** Not checked yet

30

BREAKING

A ChatGPT plan sign-in saved by an earlier version of OpenBot Desktop is no longer offered; users must sign in again with OpenAI's Sign in with ChatGPT.

— States the break but no detail on mechanism v1.2.1

---

16 **Model selection for recipient-less conversations** Not checked yet

IMPROVED

20

A new conversation with nobody in To: can now pick its model, as Home does.

— Single-sentence description with no further detail v1.2.2

└──▷ BREAKING ON UPGRADE

! An AI connection saved for Unbiased AI stops running — where it is the organization's or a person's default AI, Bots on it stop before they start. Connect another AI provider before upgrading.

! By default, every member now gets their own General Assistant; the organization's shared one is renamed or reassigned on first open after upgrade. Set `config.ownGeneralAssistant: false` or `OPENBOT_OWN_GENERAL_ASSISTANT=false` before upgrading to preserve the previous shared-only behaviour.

! A VAPID key pair that doesn't match, one key set without the other, or a `config.webPushSubject` / `WEB_PUSH_SUBJECT` value that is not an `https:` URL or `mailto:` address stops the server at start.

! A pool value that is not a whole number, or a pool under 1, stops the server at start.

! A value other than `true` or `false` for `config.ownGeneralAssistant` / `OPENBOT_OWN_GENERAL_ASSISTANT` stops the server at start.

! Google sign-in for Gemini is removed; every saved Google sign-in for Gemini is deleted at upgrade, and Gemini connections must be reconfigured with a Gemini API key.

! `config.googleModelSignIn` and `secrets.googleModelOAuthClientSecret` are no longer read on Kubernetes.

! `engine.langgraphTs.enabled` is on by default (including for `--reuse-values` upgrades); Kubernetes installs using an external secret store (`externalSecrets`) or `secrets.existingSecret` must add `managed-agent-token` and `model-gateway-token` (each `openssl rand -hex 32`) before upgrading, or `helm upgrade` refuses.

! The Helm chart no longer makes or reads `agent-tool-token` ; `secrets.agentToolToken` is ignored.

- ! AG-UI app tools are now named `<app>__<tool>` instead of `mcp__<app>__<tool>`; providers that stored the old name will see the new name going forward (old-name calls still work, but old-name storage and display are unchanged).
- ! Connected Bots no longer receive `forwardedProps.openbotSystemPrompt`; `system` or `developer` messages are now delivered as quoted conversation rather than as a role-level system message.
- ! A ChatGPT plan sign-in saved by an earlier version of OpenBot Desktop is no longer offered; users must sign in again with OpenAI's Sign in with ChatGPT.

WAS THIS USEFUL? 👍 👎

AWS Strands Agents

Feedback ⓘ harness-cli/v0.6.0 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 automatic prompt-cache placement for GPT-5.6+ models on the Bedrock Converse API, model-side media filtering via `acceptedMimeTypes`, an expanded `strands dev` auto-optimization workflow, and a cluster of breaking harness changes spanning both Python and TypeScript — a `stash` → `archive` rename, default `manage_background_task` behaviour, and a switch to the SDK's vended subagent tool.

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

01 Auto-optimize workflow improvements in `strands dev`

Not checked yet IMPROVED 85

`strands dev` numbers the two stages of auto-optimize in its output, accepts any whole number for the auto-optimize cadence, supports a `--max-rollouts` flag to cap how many optimization runs execute (e.g. `strands dev --max-rollouts 5`), and can run the auto-optimizer in an isolated dev workspace for faster CI feedback.

Run auto-optimization in `strands dev` with a specific rollout budget to cap how many optimization runs execute.

```
$ strands dev --max-rollouts 5
```

Run auto-optimization in `strands dev` with a capped number of rollouts, useful for bounding CI time or cost on a prompt-tuning pass.

```
$ strands dev --max-rollouts 10
```

Run the auto-optimizer in an isolated dev workspace, capping the number of rollouts for a faster feedback loop in CI

```
$ strands dev --max-rollouts 10
```

Run evaluation and optimization experiments in an isolated dev workspace, capping the number of rollouts for a quick feedback loop.

```
$ strands dev --max-rollouts 5
```

— Runnable flag with examples and named stages `harness-typescript/v0.6.0` `typescript/v1.27.0` `python/v1.68.0`

02 **Opt-in background task cancellation on cancel()** Not checked yet

NEW

80

Agents can cancel their spawned background tasks on demand by opting in to `cancelBackgroundTasks` on agent.cancel(), e.g. await agent.cancel({ cancelBackgroundTasks: true }), so background tasks are torn down rather than left running.

Cancel an agent and also stop any background tasks it spawned — useful when tearing down a session early in a test harness or server shutdown handler.

typescript

```
await agent.cancel({ cancelBackgroundTasks: true })
```

Let the agent clean up its own background tasks when cancelled, rather than leaving them running

typescript

```
import { createHarness } from '@strands-agents/harness'

const agent = await createHarness()
const run = agent.invoke('Run a long background job')
// cancel and also tear down any spawned background tasks
await agent.cancel({ cancelBackgroundTasks: true })
```

— Exact API call and runnable code example given `harness-typescript/v0.6.0` `typescript/v1.27.0`

03 **Float values for `context_utilization`** Not checked yet IMPROVED

80

Context management in Python now accepts float values like `0.7` for `context_utilization`, so the context manager trims at a percentage of capacity, e.g. create_harness(context_utilization=0.7), rather than a fixed token count.

Use a float context utilization budget so the context manager trims at 70% capacity rather than a fixed token count.

```python
from strands_harness import create_harness

agent = create_harness(context_utilization=0.7)
```

— Runnable code example with exact parameter and value `harness-python/v0.6.0`

---

04 **Prompt-cache points for GPT-5.6+ on Bedrock Converse**

`Not checked yet` **NEW** 60

Strands Agents now places prompt-cache points for GPT-5.6 and later on the Bedrock Converse API, falling back to implicit caching when explicit placement is not supported.

— Names model, API and fallback behaviour but no config example `harness-cli/v0.6.0` `python/v1.68.0`

---

05 **Automatic media filtering via `acceptedMimeTypes`** `Not checked yet`

NEW 60

A new `acceptedMimeTypes` field on model info causes `ModelProxy` to automatically omit media types the model does not accept from requests.

— Names the field and component but no usage example `harness-cli/v0.6.0`

THINNER COVERAGE BELOW

06 **HttpDecisionModel and dynamic tool loading** `Not checked yet` **NEW** 30

The TypeScript harness adds an `HttpDecisionModel` and dynamic tool loading.

— Named only in summary with no mechanism or example `harness-typescript/v0.6.0`

---

07 **Dogwood authorization handler for interventions** `Not checked yet`

NEW 25

Adds a Dogwood authorization handler for the interventions subsystem in Python.

— Bare name with no mechanism or usage detail `harness-cli/v0.6.0`

└──▷ BREAKING ON UPGRADE

! The harness now uses `manage_background_task` by default; existing setups that relied on the previous background-task behaviour will behave differently on upgrade.

! The context-manager's `stash` method is renamed to `archive` in TypeScript (graduated out of experimental).

! The harness now uses `manage_background_task` by default — existing setups that relied on the previous default background-task behaviour will behave differently on upgrade.

! The harness now uses the SDK's vended subagent tool instead of its own implementation (refactor marked breaking).

! The harness now uses `manage_background_task` by default (previously opt-in); agents relying on the old default background-task behaviour will behave differently on upgrade.

! The harness now uses the SDK's vended subagent tool instead of its own implementation; custom setups that depended on the previous subagent tool wiring will need to update.

! The TypeScript context manager's `stash` API is renamed to `archive`; any code calling `stash` will break.

! The context-manager's `stash` method is renamed to `archive` in Python; any code calling .stash() will break.

! The harness now uses `manage_background_task` by default; existing harnesses relying on the previous background-task default will behave differently.

! The harness now uses the SDK's vended subagent tool by default; custom subagent wiring that depended on the previous implementation may break.

WAS THIS USEFUL? 👍 👎

PydanticAI

Feedback ⓘ v2.77.0 NOTES ↗ RANK

PydanticAI is a framework that builds type-safe Python agents with dependency injection, model integrations, tools, and structured outputs.

> PydanticAI shipped a new Pax terminal coding agent with a pluggable sandbox and plugin system, launched the pydantic-ai-harness library of 100+ reusable agent capabilities, and added SQLite/Postgres/MongoDB conversation persistence alongside programmatic history compaction, unified cross-provider thinking summaries, and realtime mid-session tool updates — plus a long list of breaking changes around usage limits, sub-agent authority, and streaming state.

└──▷ **WHAT SHIPPED · 24 FEATURES** most completely described first ? what's the number?

01 **pydantic-ai-harness capability library** Not checked yet NEW

95

Adds `pydantic-ai-harness` (Pydantic AI Harness) as a library of over 100 production capabilities — including Coder(), ComputerUse(), E2BSandbox(), DockerContainer(), model fallback, memory, sub-agents, guardrails, and Agent Skills — to assemble a reusable agent harness. ComputerUse() alongside a sandbox capability such as DockerContainer(desktop=True) gives any agent a full desktop with OpenAI, Anthropic, or Google native computer-use tooling; swapping LocalWorkspace('.') for a cloud sandbox capability such as E2BSandbox() lets Coder()'s file and shell tools run in the cloud without changing any tool code; and Agent Skills load automatically from `.agents/skills` and `.claude/skills` directories via the default `coder` plugin, or via Skills() alongside Coder() in standalone Python usage.

Give any agent a full desktop environment inside a Docker container so it can click, type, and screenshot its way through GUI tasks.

python

```python
from pydantic_ai import Agent
from pydantic_ai_harness import ComputerUse, DockerContainer

agent = Agent(
    'anthropic:claude-sonnet-5-5',
    capabilities=[DockerContainer(desktop=True), ComputerUse()],
)
result = agent.run_sync('Open Firefox and find the latest Python release on python.org.')
```

Run a coding agent in a cloud E2B sandbox so all file edits and shell commands execute remotely, not on your local machine.

```python
from pydantic_ai import Agent
from pydantic_ai_harness import Coder, E2BSandbox

agent = Agent(
    'anthropic:claude-sonnet-5-5',
    capabilities=[E2BSandbox(), Coder()],
)
```

— Named capabilities with two runnable code examples `launch-20261125-ac6c5a74`

02 **Conversation history persistence backends** `Not checked yet` NEW

90

Store conversation history in SQLite, Postgres, or MongoDB using the new `SqliteConversationStore`, `PostgresConversationStore`, or `MongoDBConversationStore` backends in `pydantic_ai_harness.step_persistence.conversations`. `SqliteConversationStore` and `ConversationConflict` are renamed `SqliteSessionStore` and `SessionConflict` in the same release; the old names still import with a `HarnessDeprecationWarning`.

Persist conversation history to Postgres so long-running agents survive restarts and scale across workers.

```python
from pydantic_ai_harness.step_persistence.conversations import
PostgresConversationStore

store =
PostgresConversationStore(dsn='postgresql://user:pass@localhost/m
ydb')
```

— Named classes, module path, and a runnable example `v2.77.0`

03 **Programmatic history compaction controls** `Not checked yet` IMPROVED

90

Control history compaction programmatically with the new compact_history() capability hook and RunContext.request_compaction() method, which emit `CompactionStartEvent` and `CompactionEndEvent` on `AgentStreamEvent`; models now see compaction receipts as a typed `CompactionReceiptPart` instead of a `UserPromptPart`, for more precise history parsing. Default `FallbackCompaction` chains now recover from `UnexpectedModelBehavior` instead of propagating it immediately, and apps carrying `result.conversation` forward with a summarizing

capability (`SummarizingCompaction` , `RelevanceCompaction`) now get the full history re-summarized each run; pass result.all_messages() as `message_history=` to carry the compacted history instead.

— Names hooks, events, part types and a migration path `v2.77.0` `v2.76.0`

---

04 **Pax CLI agent selection and session import flags** Not checked yet

NEW

80

Adds `--agent` / `-a` to point Pax at any PydanticAI `Agent` from your codebase instead of the default agent, running it with its own model, tools, and instructions while keeping the full terminal; and `pax --resume-claude` or `pax --resume-codex` to import existing sessions from Claude Code or Codex and continue them on any model.

Run your own PydanticAI agent in a real terminal session, using its own model, tools, and instructions, without loading any default Pax plugins.

```
$ pax --agent my_agents:reviewer -p "Review the diff on this branch against our guidelines"
```

— Exact flags and a runnable command given `launch-20261125-ac6c5a74`

---

05 **Unified thinking summary setting** Not checked yet NEW

80

Get readable thinking output across Anthropic and Bedrock providers via the new `thinking_summary` unified setting, alongside provider-specific `anthropic_thinking_display` and `bedrock_thinking_display` settings. On Anthropic and Claude on Bedrock, a request using the unified `thinking` setting now sends `thinking.display: 'summarized'` , which can add latency before the first text token.

— Names settings and the exact wire value sent `v2.77.0`

---

06 **Preflight validation errors for OpenAIChatModel** Not checked yet

80

BREAKING

With `OpenAIChatModel` on the OpenAI or Azure provider, PydanticAI now raises `UserError` before sending image-output requests (`output_type=BinaryImage` or `output_type=[str, BinaryImage]`); and raises `ModelRequestRejectedError` before sending tools to a GPT-5.4 or GPT-5.6 model with reasoning on, replacing a later provider 400 `ModelHTTPError` , so `except ModelHTTPError` , `status_code` / `body` checks, and `fallback_on` entries naming `ModelHTTPError` no longer match.

— Exact error types, triggers and code that stops matching `v2.77.0`

07 **Pax sandbox switching and computer-use commands** Not checked yet

NEW

75

Swap the sandbox a Pax session runs in before the first prompt with `/sandbox e2b`, `/sandbox boxd`, `/sandbox daytona`, `/sandbox opensandbox`, `/sandbox modal`, `/sandbox render`, `/sandbox sprites`, `/sandbox kubernetes`, `/sandbox local smol`, or `/sandbox local boxlite`; also adds `/computer-use` desktop sandboxes to Pax for GUI-driving tasks.

— Names every sandbox target as an exact command v2.76.0 launch-20261125-ac6c5a74

08 **Root conversation span tracking** Not checked yet NEW

75

Record `pydantic_ai.root_conversation_id` on spans of nested agent runs so sub-agent token cost rolls up to the outer conversation.

Track token cost across nested agent runs by rolling sub-agent spans up to the root conversation in your observability backend.

python

```
import opentelemetry.sdk.trace as sdk
# pydantic_ai.root_conversation_id is set automatically on
nested-run spans;
# just read it from the span attributes in your exporter:
for span in finished_spans:
    root_id =
span.attributes.get('pydantic_ai.root_conversation_id')
    if root_id:
        bill_to_conversation(root_id,
span.attributes.get('gen_ai.usage.total_tokens', 0))
```

— Named span attribute with a runnable rollup example v2.76.0

09 **ConversationSearch scan and query limits** Not checked yet IMPROVED

75

`ConversationSearch`'s `search_conversation_history` now scores only the newest `max_scanned_messages` messages (default `2000`); raise the setting to search further back. A query over 1,000 characters or 32 unique terms raises `ModelRetry`.

— Names the setting, its default and retry trigger limits v2.77.0

10 **FileSystemToolset read_file offset fix** Not checked yet BREAKING

75

FileSystemToolset.read_file(..., offset=N) and `ReadFileArgs` / `ReadFileCallPart` built with an explicit `offset` now start one line earlier, and `offset=0` raises; replace

every `offset=N` call with `offset=N+1` (find with `grep -rn 'read_file(.*offset' .`).

— Exact migration command and call signature given v2.76.0

---

11 **Run-start context placement before prompt** Not checked yet

IMPROVED

70

Place Memory and Planning run-start context before the user prompt by passing enqueue(placement='before_prompt'); new runs now record this context inside the prompt's `ModelRequest` before the user prompt rather than after, so code reading the snapshot at `all_messages()[1]` or treating the prompt as the request's last part sees a different shape.

— Names the exact call and the shape change it causes v2.77.0

---

12 **Realtime tool updates mid-session** Not checked yet NEW

70

Reveal tools mid-session in realtime sessions by setting `RealtimeModelProfile.supports_tool_updates` on a model profile and sending `UpdateTools` through `RealtimeConnection.send`; `RealtimeInput` gains `UpdateTools` and `DEFAULT_REALTIME_PROFILE` gains `supports_tool_updates: False`.

— Names the profile flag, input type and send path v2.76.0

---

13 **GoogleInteractionsModel stateless support** Not checked yet NEW

70

Add stateless Google Interactions support with the new `GoogleInteractionsModel` class. Requests to `gemini-2.5-pro` or `gemini-2.5-flash` through `GoogleModel` or `GoogleInteractionsModel` now emit `DeprecatedModelWarning`, so code using `filterwarnings = error` fails on those requests unless moved to a replacement model, the warning is filtered, or `profile={'deprecation': None}` is set.

— Names the class and the migration path for the warning v2.76.0

---

14 **Fallback subagent usage counted against run limits**

Not checked yet

BREAKING

65

A fallback subagent's requests and tokens now count against the run's `UsageLimits`, including the default `request_limit=50`, so a run near a limit that used to finish can now raise `UsageLimitExceeded`.

— Names the limit, its default and the new exception v2.77.0

---

15 **Stream response terminal state flag** Not checked yet

BREAKING

65

The final stream_response() yield now has `state='incomplete'` when an Anthropic, Google, or OpenAI Responses stream ends before its terminal event; code reading `response.state != 'incomplete'` or `state == 'complete'` as the stream ended now sees the difference.

— Names the exact field and value to check v2.77.0

16 **Skills missing library path handling** Not checked yet BREAKING

65

Skills(...) without `missing_directories` now skips a missing or unreadable library path instead of raising; pass `missing_directories='error'` to keep the old behavior. `SkipReason` gains a value, so an exhaustive `match` on it stops type-checking.

— Names the parameter, its values and the type impact v2.77.0

17 **Pax plugin toggle commands** Not checked yet NEW

60

Toggle Pax plugins — including `coder`, `sandbox`, `observability`, `mcp`, and others — at runtime with `/plugins disable <name>` or `/plugins enable <name>`.

— Exact commands named, little mechanism beyond toggling launch-20261125-ac6c5a74

18 **CodeMode final output submission** Not checked yet NEW

60

Use CodeMode(allow_final_output=True) to let sandboxed model-written code submit the run's final output directly.

— Exact flag named and what it unlocks v2.76.0

19 **ModelSchemaTooComplexError retry for Anthropic** Not checked yet

IMPROVED

60

Raise `ModelSchemaTooComplexError` for Anthropic's Schema is too complex 400 error, then automatically retry with an inferred Native Output schema in Tool Output mode.

— Names the error type and the automatic retry mechanism v2.76.0

THINNER COVERAGE BELOW

20 **Pax speculative execution for file reads** Not checked yet NEW

55

Enable speculative execution in Pax so safe file reads begin while the model is still streaming, by pressing Ctrl+X, then Ctrl+S.

— Exact keystrokes given, mechanism described thinly launch-20261125-ac6c5a74

21 **OpenSandboxBackend volume mounting** Not checked yet IMPROVED

50

Forward `volumes` through `OpenSandboxBackend` to `Sandbox.create` for volume-mounted sandboxes.

— Single parameter named, no further detail given v2.76.0

22 **Pax footer and status segment enhancements** Not checked yet NEW

45

Show the `/permissions` mode in the Pax footer via the new `'permissions'` value on the `FooterField` literal, and let a Pax plugin status segment return a `StatusLink` (in addition to `str`) for a clickable link in the status row.

— Thin UI additions, no navigation path given v2.77.0 v2.76.0

23 **Per-tool UI visibility control** Not checked yet NEW 40

Add per-tool `ui_visibility` to keep specific tool results from reaching the client in UI adapters.

— Single flag named, no further mechanism given v2.76.0

24 **Pax extras install for every model provider** Not checked yet

IMPROVED 35

Install `pydantic-ai-pax` extras for every model provider plus `all`, and include the Google SDK by default in Pax.

— Bare statement of a dependency bundling change v2.76.0

└──▷ **BREAKING ON UPGRADE**

! `SqliteConversationStore` and `ConversationConflict` in `pydantic_ai_harness.step_persistence.conversations` are renamed `SqliteSessionStore` and `SessionConflict`; old names still import with a `HarnessDeprecationWarning`.

! Skills(...) without `missing_directories` now skips a missing or unreadable library path instead of raising; pass `missing_directories='error'` to keep the old behavior. `SkipReason` gains a value, so an exhaustive `match` on it stops type-checking.

! On Anthropic and Claude on Bedrock, a request with the unified `thinking` setting now sends `thinking.display: 'summarized'`, which can add latency before the first text token.

! `AgentStreamEvent` gains `CompactionStartEvent` and `CompactionEndEvent`; an exhaustive `match` over the union (`assert_never`) stops type-checking.

! `SubAgents` no longer loads a sub-agent definition file below `'operator'` authority; durable runs using `SubAgents` with `agent_folders` now record one `realpath` per agent definition file at run start — let in-flight runs finish, or use Temporal worker versioning or a new DBOS app version, before upgrading.

! Default `FallbackCompaction` chains now recover from `UnexpectedModelBehavior` instead of propagating it immediately.

! The final stream_response() yield now has `state='incomplete'` when an Anthropic, Google or OpenAI Responses stream ends before its terminal event; code reading `response.state != 'incomplete'` or `state == 'complete'` as the stream ended sees the difference.

! A fallback subagent's requests and tokens now count against the run's `UsageLimits`, including the default `request_limit=50`, so a run near a limit that used to finish can raise `UsageLimitExceeded`.

! New runs record `Memory` and `Planning` run-start context inside the prompt's `ModelRequest` before the user prompt; code that reads the snapshot at `all_messages()[1]` or the prompt as the last part of its request sees a different shape.

! Models now see compaction receipts as `CompactionReceiptPart` instead of `UserPromptPart`; code that reads receipts from history as `UserPromptPart`s should look for `CompactionReceiptPart`.

! On `AsyncAnthropicFoundry` with a supported Claude model, mid-conversation system prompts are sent as native `{'role': 'system'}` entries instead of `<system>`-tagged user text; a conversation already in progress misses the prompt cache once.

! With `OpenAIChatModel` on the OpenAI or Azure provider, a request with `output_type=BinaryImage` or `output_type=[str, BinaryImage]` now raises `UserError` before it is sent.

! With `OpenAIChatModel` and a GPT-5.4 or GPT-5.6 model, a request that sends tools with reasoning on now raises `ModelRequestRejectedError` instead of `ModelHTTPError`; `except ModelHTTPError`, `status_code`/`body` checks, and `fallback_on` entries naming `ModelHTTPError` no longer match.

! `FooterField` literal gains `'permissions'` and `Status.FIT_STEPS` gains a step; exhaustive matches over `FooterField` or comparisons of `FIT_STEPS` to a fixed tuple see one more value.

! `LogfireSettings.service_name` in Pax's Logfire plugin now defaults to `pydantic-ai-pax`; to keep the old name, set it in `/plugins configure logfire`.

! A gh-aw workflow that imports the engine definition from `main` and sets its own `engine: version:` to a release from 0.59.0 to 0.72.x now fails at install time; raise the override to 0.73.0 or newer, or drop it.

! The gh-aw engine's Pax run now sets `PYTHON_DOTENV_DISABLED=1`, so a `.env` file committed to the repository no longer supplies its variables.

! The `google` and `google-realtime` extras now require `google-genai>=2.34.0`.

! `pydantic-ai-slim` now requires `genai-prices>=0.2.4`.

! ModelResponse.cost(), EmbeddingResult.cost() and ImageGenerationResult.cost() now raise ValueError('Model name is required to calculate price') instead of a bare `AssertionError` when the model name is missing; code that caught `AssertionError` from cost() needs `ValueError` instead.

! `ConversationSearch`'s `search_conversation_history` now scores only the newest `max_scanned_messages` messages (default `2000`); raise the setting to search further back.

A query over 1,000 characters or 32 unique terms raises `ModelRetry`.

! Stdio MCP servers that plain agent.run() calls start now stop once the agent and toolset go out of scope; FastMCP handlers called from a kept-alive stdio connection now see get_current_run_context() return `None`.

! `RealtimeInput` (the union `RealtimeConnection.send` accepts) gains `UpdateTools`, and `DEFAULT_REALTIME_PROFILE` gains `supports_tool_updates: False`; an exhaustive match over `RealtimeInput` using `assert_never` now fails type-checking — handle `UpdateTools` there or raise for it.

! Requests to `gemini-2.5-pro` or `gemini-2.5-flash` through `GoogleModel` or `GoogleInteractionsModel` now emit `DeprecatedModelWarning` (a `UserWarning`); code using `filterwarnings = error` will fail on those requests — move to the replacement model, filter the warning, or set `profile={'deprecation': None}`.

! `Selection.feed`, `span`, `highlight` and `text` in `pydantic_ai_pax.ui.prompt.prompt_selection` take new arguments: `feed` takes `now`, `span` takes `frame` and `joins` in place of `width`, and `highlight` and `text` take `joins` — callers must be updated.

! A Pax plugin status segment may now return a `StatusLink` in addition to `str`; a segment that returns `str` works as before, but exhaustive type matches must handle `StatusLink`.

! `pydantic_ai_pax.cli.self_update.PACKAGES` now requests `pydantic-ai-slim[anthropic,google,mcp,openai]` instead of the previous package set.

! Apps carrying `result.conversation` into the next run with a summarizing capability (`SummarizingCompaction`, `RelevanceCompaction`) now have the full history re-summarized on each run; to carry the compacted history instead, pass result.all_messages() as `message_history=`.

! FileSystemToolset.read_file(..., offset=N) and `ReadFileArgs` / `ReadFileCallPart` built with an explicit `offset` now start one line earlier and `offset=0` raises; replace every `offset=N` call with `offset=N+1` (find with `grep -rn 'read_file(.*offset' .`).

WAS THIS USEFUL?  

open-multi-agent

Feedback ⓘ v1.26.0 NOTES ↗ RANK

open-multi-agent is a self-hosted TypeScript agent runtime with durable approvals and verifiable run records.

> open-multi-agent's v1.26.0 release adds a hard byte cap on process-backed agent output and allows async stream callbacks, alongside two breaking changes affecting streaming refusal handling and Gemini parameter forwarding.

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

01 Hard byte cap on process-backed agent output

Not checked yet

NEW

90

A new `maxOutputBytes` setting in the process backend config caps combined stdout and stderr from process-backed agents; omitting it keeps output unlimited, and exceeding the limit kills the process tree and fails the run with a diagnostic error. Intended to bound untrusted or noisy commands from flooding memory or storage.

Prevent a process-backed agent from flooding memory or storage when running untrusted or noisy shell commands by setting a hard output byte ceiling.

json

```
{
  "processBackend": {
    "maxOutputBytes": 1048576
  }
}
```

— Named config key with exact mechanism and runnable example v1.26.0

02 Streaming refusal handling changed for OpenAI, Azure OpenAI, Copilot adapters

Not checked yet

BREAKING

75

Handlers for streaming refusals from the OpenAI, Azure OpenAI, and Copilot adapters that previously expected empty content with an `end_turn` stop reason must now handle text events and a `content_filter` stop reason instead.

— Clear before/after migration note, no code example v1.26.0

03 Async-capable stream callbacks

Not checked yet IMPROVED

60

Stream callbacks in `OrchestratorConfig.onAgentStream` and pool stream callbacks now accept `Promise<void>` return values in addition to `void`; returned promises are observed for rejection without being awaited.

— Names exact config surface but no example given v1.26.0

THINNER COVERAGE BELOW

04 **Gemini topK parameter now forwarded to provider** Not checked yet

50

BREAKING

Callers that set Gemini `topK` and relied on it being silently dropped will now see it forwarded to the provider; callers should recheck provider behavior and model support.

— Names the field but gives no concrete remediation steps v1.26.0

└──▷ **BREAKING ON UPGRADE**

! Handlers for streaming refusals from the OpenAI, Azure OpenAI, and Copilot adapters that expected empty content with an `end_turn` stop reason must now handle text events and a `content_filter` stop reason instead.

! Callers that set Gemini `topK` and relied on it being silently dropped will now see it forwarded to the provider; recheck provider behavior and model support.

WAS THIS USEFUL?  

HeyGen HyperFrames

Feedback ⓘ v0.9.104 NOTES ↗ RANK

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

> HyperFrames expanded sealed-preview editing with ghost display, stage acts, keyboard nudges and keypress forwarding, extended frame-presented waits to a 15 s ceiling with new health metrics, and added `shownPreviewFrame` and composition-size exports plus a stage-layout measurement utility.

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

01 **`shownPreviewFrame` export from Studio** Not checked yet NEW

65

Studio now exports `shownPreviewFrame` so host applications can read which preview frame has been promoted.

– Named, directly usable export; no code sample included. v0.9.100

02 **Sealed preview editing enhancements** Not checked yet NEW

60

Sealed previews now display removed elements as ghosts so editors can see what was deleted, can hold and draw stage acts over the editing port, and can run Delete and arrow-key nudges in-frame once Studio arms them. Studio also now passes keypress events to the host during a sealed text edit, including press-location information.

– UI behaviors clearly described but no API or flag names given. v0.9.103

THINNER COVERAGE BELOW

03 **Composition size export via Parsers** Not checked yet NEW

55

The size the renderer compiles a composition at can now be read via the Parsers export, making rendered dimensions available to downstream tooling.

– Named export surface but no usage sample provided. v0.9.104

04 **Configurable wait-time ceiling for frame-presented waits**

Not checked yet IMPROVED

45

A frame-presented wait can now run longer, up to 15 s, giving long-running renders more time before timing out.

– Concrete 15 s limit named but no config key given. v0.9.104

05 **Lint warning for oversized 3D-transformed elements**

Not checked yet NEW

40

Lint now warns when a 3D-transformed element is far larger than the stage.

— Behavior named but no threshold or rule ID given. v0.9.103

06 **Frame-presented request and preview health metrics**

Not checked yet NEW 30

The runtime now exposes frame-presented request and preview health metrics.

— No metric names, format, or endpoint specified. v0.9.103

07 **Preview stage layout and style measurement in Core**

Not checked yet NEW 25

Core can now measure preview stage layout and styles over the editing port.

— Single thin line with no mechanism or API detail. v0.9.101

WAS THIS USEFUL? 👍 👎

openexecutive

Feedback ⓘ v0.8.0 NOTES ↗ RANK

Openexecutive is an ai-powered virtual executive team — a single coherent executive persona backed by 8 specialist agents (FastAPI + Next.js).

> OpenExecutive v0.8.0 adds Google Messages access, voice input, and self-hosted/BYO-key routing for Claude, Azure models, Gemini and Grok, alongside a reorganised home screen and a breaking change to delegation gating.

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

01 **Self-hosted and BYO-key model routing** Not checked yet NEW

75

Run Claude on a company's own Amazon Bedrock or Google Vertex AI instance instead of Anthropic's hosted endpoint, run models deployed in a company's own Azure account, and send Gemini and Grok requests using Google's and xAI's own API keys.

– Names all providers but no config keys or setup steps given v0.8.0

02 **Delegation now routes through single gate** Not checked yet

60

BREAKING

Delegation send as you now routes through a single gate with a single In training state; existing delegation flows that relied on the previous multi-gate behaviour will break on upgrade.

– Explains before/after but no migration command given v0.8.0

THINNER COVERAGE BELOW

03 **Google Messages read/send via browser extension** Not checked yet

NEW

40

Read and send texts through Google Messages via the browser extension.

– Clear capability but no navigation or config detail v0.8.0

04 **Voice input in chat** Not checked yet NEW

40

Use voice input in chat with a Gemini key or the browser's own speech recognition.

– States mechanism options but no setup steps v0.8.0

05 **Home screen reorganised into sections** Not checked yet IMPROVED

25

Home screen now reorganised into For you , What I'm doing , and I could also sections.

— Names sections but no detail on behavior change v0.8.0

└──▷ **BREAKING ON UPGRADE**

! Delegation send as you now routes through a single gate with a single In training state; existing delegation flows that relied on the previous multi-gate behaviour will break on upgrade.

WAS THIS USEFUL? 👍 👎

Mastra

Feedback ⓘ announced · Nov 27, 2026 BLOG ↗ RANK

Mastra is a TypeScript framework for AI applications and agents, providing workflows, agent memory, RAG, tool calling, and evals.

> Mastra shipped encrypted secrets management for Mastra Platform, keeping LLM API keys in a KMS and write-only everywhere, alongside generative UI cards that let Slack-connected agents render Block Kit fields, tables, charts and images at runtime.

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

01 Generative UI cards for Slack agents

Not checked yet NEW

88

Enable `generativeUI: true` on the Slack provider inside channels() and add `@json-render/chat` to `bundler.dynamicPackages` (requires `@mastra/core@1.83.0` or later) to give agents a `render_card` tool. Agents pass a `json-render` spec object to `render_card` to post native Slack Block Kit elements — fields, tables, charts, images, and link buttons — chosen dynamically at runtime rather than being predefined in code.

Turn on generative UI so a Slack-connected agent can compose and post rich cards — fields, tables, charts — without predefining card structure in code.

typescript

```typescript
import { Mastra } from "@mastra/core/mastra";
import { channels } from "@mastra/connect";
import { deployAgent } from "./agents/deploy-agent";

export const mastra = new Mastra({
  agents: { deployAgent },
  bundler: {
    dynamicPackages: ["@json-render/chat"]
  },
  channels: await channels({
    providers: {
      slack: {
        providerOptions: {
          generativeUI: true
        }
      }
    }
  })
});
```

— Exact config keys, package name, version requirement, and runnable code sample.

launch-20261124-4a7b84f9

02 **Encrypted secrets management in Mastra Platform** Not checked yet

87

BREAKING

Secrets are now stored encrypted via a key management service (KMS) rather than in plaintext variable stores, so values are never written to disk, container images, logs, deploy artifacts, or a hosting provider's variable store. Values are injected into the running server's process memory at startup (not build time) after a split-token double-key match, every decryption is audited for a full access trail, and saved secrets are stored as an `msec:` value in the database and are strictly write-only — never returned by the dashboard or API once saved. Use the Secret toggle in the dashboard when saving a variable to mark it encrypted (unmarked variables still pass through as plaintext at build time), and run `mastra env vars pull` to list secret names locally without writing their values into the pulled file.

Pull environment variable names for a project to a local file without exposing secret values — useful for auditing what secrets exist without risk of leaking them.

```
$ mastra env vars pull
```

— Names KMS mechanism, CLI command, DB prefix and toggle precisely.

launch-20261127-31a3f588

└──▷ **BREAKING ON UPGRADE**

! Secret values saved in the platform are no longer readable after saving — the dashboard and API no longer return stored secret values, and saved values are no longer displayed in the UI or returned to clients.

WAS THIS USEFUL? 👍 👎

EXXETA exxperts

Feedback ⓘ v0.16.2 NOTES ↗ RANK

The exxperts framework runs local-first AI agents with approval-gated memory, model-provider integrations, MCP tools, and web search.

> exxperts added cloud storage support for Rooms on macOS, letting folders in OneDrive, Google Drive, or Dropbox be used directly, with clearer feedback when a folder choice is invalid.

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

01 **Cloud drive folders for Rooms on macOS** Not checked yet NEW

55

Rooms on macOS now accept folders stored in OneDrive, Google Drive, or Dropbox, selectable via `Room settings > Folder`. The room settings UI also now explains why a given folder cannot be used and suggests what to choose instead.

Assign a cloud-synced folder (OneDrive, Google Drive, or Dropbox) to a room on macOS so its files are available through that provider.

> 📍 In Room settings, go to Folder and select a folder inside your OneDrive, Google Drive, or Dropbox directory.

— Names exact UI path and providers but no technical mechanism. v0.16.2

WAS THIS USEFUL? 👍 👎

Instructor

Feedback ⓘ v1.18.0 NOTES ↗ RANK

Instructor returns validated structured outputs from language models by defining response shapes as Pydantic models with automatic retries.

> Instructor v1.18.0 expands provider support with Azure Foundry and OpenAI-compatible gateway clients, adds Mistral batch job handling, OpenAI Decisions API classification, and bounded schema analysis for mode suggestions.

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

01 **Azure Foundry and OpenAI-compatible provider support via `from_provider`**

Not checked yet NEW 65

Instructor supports Azure Foundry clients and Azure provider aliases via `from_provider`, and shares OpenAI-compatible gateway clients and mode contracts so any OpenAI-compatible provider can be used through the same `from_provider` interface.

– Names config interface and two provider targets, no code sample v1.18.0

THINNER COVERAGE BELOW

02 **Classification and scoring via OpenAI Decisions API**

Not checked yet NEW 55

Instructor adds classification and scoring through the OpenAI Decisions API using typed `Jev` decision models.

– Names API and model type but no usage example v1.18.0

03 **Mistral batch job support** Not checked yet NEW 40

Instructor adds Mistral batch job submission and result retrieval to the batch processing subsystem.

– States capability without usage detail v1.18.0

04 **Bounded schema analysis with mode suggestions** Not checked yet

NEW 35

Instructor adds bounded schema analysis with opt-in mode suggestions to help callers choose the right structured output mode.

– Describes feature in prose only, no flag or command given v1.18.0

WAS THIS USEFUL?  

lexmount Moli

Feedback ⓘ v1.1.28 NOTES ↗ RANK

Moli is a best headless browser for AI agents. Lite, Fast, High-Compatibility. Built in Rust

> Moli v1.1.28 ships a new GPUI-CE desktop browser app, HarmonyOS ARM64 support, expanded H.264/AAC video playback, a batch of new CDP automation endpoints, and a realtime incremental layout engine.

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

01 **New CDP automation and emulation capabilities** Not checked yet

NEW 80

Moli adds multiple new CDP capabilities for automation mode: the `Schema.getDomains` endpoint to expose page domains, exposing the browser command line, DevTools node and rectangle highlight rendering, vision deficiency simulation via the emulation domain, honoring operating system text scale via the emulation domain, honoring mobile device metrics and viewport directives via the CDP emulation domain, and synthesis of mouse scroll gestures via the input domain.

– Names a concrete endpoint and specific CDP domains v1.1.28

02 **GPUI-CE desktop browser via Molio project** Not checked yet NEW

60

Moli adds a GPUI-CE desktop browser application via the Molio project, including tabs, favicons, file downloads, a downloads page, native cursors, and frame pacing tied to the display clock.

– Lists concrete app features but no usage steps given v1.1.28

THINNER COVERAGE BELOW

03 **H.264 and Media Source video playback** Not checked yet NEW

45

Moli adds H.264 software video playback on Linux, plus bundled H.264 video playback and H.264 and AAC Media Source stream playback on macOS.

– Describes codecs and platforms but no enablement flag v1.1.28

04 **Realtime incremental layout engine** Not checked yet NEW

45

Moli adds realtime layout with complete layout passes, enabling incremental rendering that retains and updates layout worlds across frames.

– Explains mechanism but exposes no config or API v1.1.28

05 **HarmonyOS ARM64 support** Not checked yet NEW 40

Moli CLI now supports HarmonyOS ARM64, with a new browser engine adapter for HarmonyOS available.

— Names platform and adapter but no setup instructions v1.1.28

WAS THIS USEFUL? 👍 👎

octop-harness Feedback ⓘ v1.0.13 NOTES ↗ RANK

Octop-harness is a production-grade agent runtime engineered from the Harness Engineering philosophy.

> Octop-harness now forwards expert-configured memory retention and skip rules into its memory extraction pipeline via the octop-memory integration.

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

01 **Retention and skip rules in memory pipeline** Not checked yet NEW 45

Expert-configured retention rules and skip rules are now forwarded to the memory extraction pipeline via the `octop-memory` integration.

— Names the integration but not config keys or mechanism. v1.0.13

WAS THIS USEFUL? 👍 👎

AutoGPT

Feedback ⓘ autogpt-platform-beta-v0.12.1 NOTES ↗ RANK

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

> AutoGPT Platform's latest release focuses on backend reliability (a new readiness endpoint and configurable keep-alive timeout), new Attio/Neon integrations and mobile app support for Otto, plus smaller fixes to the External API and sign-in flow.

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

01 Configurable server keep-alive timeout

Not checked yet IMPROVED

88

Set `SERVER_KEEP_ALIVE_TIMEOUT_SECONDS` (default 620) to keep idle backend connections open longer than upstream proxies, replacing the previous 65-second nginx default.

Increase idle connection lifetime so a reverse proxy (nginx, ALB) does not close connections before the backend does, preventing mid-request resets.

env

```
SERVER_KEEP_ALIVE_TIMEOUT_SECONDS=620
```

— Exact config key, default value, and before/after given. autogpt-platform-beta-v0.12.1

02 Readiness endpoint for health checks

Not checked yet NEW

80

New `/health/ready` endpoint checks the database and returns 503 on failure, giving load balancers and orchestrators a reliable readiness signal.

Poll the new readiness endpoint in a Docker or Kubernetes health check to ensure the backend has a live database connection before routing traffic.

```
$ curl -f http://localhost:8000/health/ready || exit 1
```

— Names exact endpoint and gives a runnable health-check command.

autogpt-platform-beta-v0.12.1

03 Mobile crash reporting, OTA updates, minimum app version

Not checked yet NEW

70

Enable mobile crash reporting, over-the-air updates, and a server-enforced minimum app version via the optional `MOBILE_MIN_APP_VERSION` environment variable.

Enforce a minimum mobile app version from the server so that clients below the threshold are prompted to update before connecting.

```env
MOBILE_MIN_APP_VERSION=0.12.1
```

— Names env var and effect but not the reporting mechanism. autogpt-platform-beta-v0.12.1

THINNER COVERAGE BELOW

04 **External API run output and version lookup fixes** Not checked yet

IMPROVED 55

Run outputs in the External API now include formatted text from Agent Output blocks when a format is set, rather than the raw value, and a workflow looked up via the External API without a version now returns its active version instead of failing.

— Describes behavior change but names no endpoint or example.

autogpt-platform-beta-v0.12.1

05 **Unified Google/Microsoft sign-in for email, files, calendar**

Not checked yet IMPROVED 45

Sign in with Google or Microsoft now connects email, files, and calendar on the same screen, so Otto can use them from the first chat without an extra setup step.

— Describes UX flow but no config or technical surface. autogpt-platform-beta-v0.12.1

06 **Connect a second computer from Local PC dialog** Not checked yet

NEW 35

Connect a second computer from the Local PC folder dialog.

— Names UI path but no further mechanism or scope. autogpt-platform-beta-v0.12.1

07 **Attio and Neon integrations for Otto** Not checked yet NEW

30

New integrations with Attio and Neon are available for Otto, with each service shown under its own name and logo.

— Names integrations but no setup or usage detail. autogpt-platform-beta-v0.12.1

WAS THIS USEFUL? 👍 👎

◆ AI Agent Communication

AG-UI

Feedback ⓘ release/2026-11-27 NOTES ↗ RANK

AG-UI: the Agent-User Interaction Protocol. Bring Agents into Frontend Applications.

> AG-UI's single release this window reworked guidance and CopilotKit wiring: it added a dedicated CopilotKit 2.x entry point for the Mastra adapter, introduced per-run guidance middleware for LangGraph, and changed how the Mastra adapter manages per-run request context.

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

01 Per-run guidance control and request context in Mastra

Not checked yet 88

BREAKING

In the Mastra adapter, set `runGuidance: false` to opt out of per-run guidance, which keeps the reserved context entries `OpenBot standing role` and `OpenBot granted tools guidance` available to custom tool and prompt callbacks; those entries are no longer present when guidance is enabled. The adapter now also creates a fresh `RequestContext` per run and copies host-provided values into it, so code that relied on the adapter writing run context into the original object must instead read from the `RequestContext` passed to the callback.

— Config key and migration path given, but no code sample. release/2026-11-27

02 Per-run guidance middleware for LangGraph

Not checked yet NEW 86

Enable per-run guidance on LangGraph agents by composing `RunGuidanceMiddleware` and `RunGuidanceAgentMixin` into an existing LangChain chat graph served through an `ag-ui-langgraph` adapter. Run guidance behavior and state snapshot handling in `ag-ui-langgraph` changed, so guidance opt-out and snapshot contents should be verified after upgrading.

Add per-run guidance to an existing LangChain chat graph served via ag-ui-langgraph.

python

```python
from ag_ui_langgraph import RunGuidanceMiddleware,
RunGuidanceAgentMixin

graph = RunGuidanceMiddleware(existing_graph)
adapter = RunGuidanceAgentMixin(adapter)
```

— Names both classes and adapter with a runnable example. release/2026-11-27

03 **CopilotKit 2.x entry point for Mastra** Not checked yet BREAKING

80

Register CopilotKit 2.x with the Mastra adapter via the new `@ag-ui/mastra/copilotkit-2` entry point, which imports the runtime from `@copilotkit/runtime` instead of the removed `@copilotkit/runtime/v2`.

Wire a CopilotKit 2.x frontend to a Mastra agent using the new dedicated entry point.

typescript

```
import { registerCopilotKit } from '@ag-ui/mastra/copilotkit-2';
```

— Names old and new import paths with a usable snippet. release/2026-11-27

└──▷ **BREAKING ON UPGRADE**

! If a custom tool or prompt callback reads `OpenBot standing role` or `OpenBot granted tools guidance` from `requestContext.get('ag-ui').context`, those entries are no longer present when guidance is enabled; set `runGuidance: false` to restore them.

! The Mastra adapter now creates a fresh `RequestContext` per run and copies host-provided values into it; code that relied on the adapter writing run context into the original object must instead read from the `RequestContext` passed to the callback.

! Run guidance behavior and state snapshot handling in `ag-ui-langgraph` changed; verify guidance opt-out and snapshot contents after upgrade.

WAS THIS USEFUL? 👍 👎

◆ Agent Memory

claude-mem

Feedback ⓘ v17.0.2 NOTES ↗ RANK

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

> claude-mem v17.0.2 adds ZCode as a supported IDE with its own install path and status checks, introduces per-prompt semantic memory injection for OpenCode, and exposes four new environment variables for controlling model selection, field condensing, and Bash skip-pattern matching, plus three new OpenAI-compatible provider presets.

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

01 ZCode installation and status support

Not checked yet NEW

80

Running `npx claude-mem install --ide zcode` installs ZCode hooks and MCP recall; the `status` command now reports when a ZCode setup is disabled or incomplete.

Add claude-mem to ZCode so hooks and MCP recall are active and `status` can report on the setup.

```
$ npx claude-mem install --ide zcode
```

— Names exact install flag and status subcommand v17.0.2

02 Bash skip pattern segment matching

Not checked yet NEW

78

Setting `CLAUDE_MEM_SKIP_BASH_PATTERNS_MATCH_EVERY_SEGMENT=true` changes how `CLAUDE_MEM_SKIP_BASH_PATTERNS` is matched against compound Bash commands, so a command is skipped only when every segment matches.

— Names both env vars and the exact matching rule v17.0.2

03 Per-prompt semantic memory injection for OpenCode

Not checked yet NEW

75

Setting `CLAUDE_MEM_SEMANTIC_INJECT=true` enables per-prompt semantic memory injection for OpenCode, so each prompt receives the observations that best match it.

— Names the environment variable and matching behavior v17.0.2

04 Field condensing bypass via env var

Not checked yet NEW

75

Setting `CLAUDE_MEM_FIELD_CONDENSE_ENABLED=false` skips the model call that shortens an oversized tool field; the field goes straight to head-and-tail trimming instead.

— Names flag and describes before/after fallback behavior v17.0.2

---

05 **OpenAI-compatible observer model from session** Not checked yet

NEW

65

Setting `CLAUDE_MEM_OPENAI_COMPAT_MODEL_FROM_SESSION=true` lets the OpenAI-compatible observer use the model picked in the active OpenCode session.

— Env var named but mechanism described only briefly v17.0.2

THINNER COVERAGE BELOW

06 **New OpenAI-compatible provider presets** Not checked yet NEW

45

Adds Rapid-MLX, Straitly, and TeamoRouter as presets for the OpenAI-compatible provider.

— Names presets but gives no configuration detail v17.0.2

WAS THIS USEFUL? 👍 👎

◆ Web Search & Scraping for Agents

Exa

Feedback ⓘ announced · Nov 23, 2026 BLOG ↗

Exa is a web search API built for AI agents, with search, content retrieval and research endpoints.

Exa adds live flight data to its search API: status by flight number, airport boards, and route options, all via the existing search endpoint at no extra cost.

NEW INTEGRATION Retrieve live flight status by flight number — including gates, terminals, baggage claim, scheduled/estimated/actual times, codeshares, aircraft type, and diversions — via Exa's existing search endpoint with no new endpoint or extra cost. Not checked yet

Fetch departure and arrival boards for any airport or city, filterable by airline, time window, or status, through the standard search endpoint. Not checked yet

Search flight options between any two cities — nonstop or up to two stops, including round trips and multi-airport cities — directly from the search endpoint. Not checked yet

Track a specific flight's latest status at any time by fetching its stable, single URL via `/contents`, returning always-current gate, estimated arrival, and baggage belt data without re-querying search. Not checked yet

WAS THIS USEFUL? 👍 👎

◆ Applied AI Agents

Legora

Feedback ⓘ announced · Nov 23, 2026 BLOG ↗ RANK

Legora is a collaborative AI workspace for lawyers.

> Legora launched Fact Intelligence, a new capability that processes case documents upfront to extract and link facts, and integrates with Legal Research to connect law to case-specific facts.

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

01 **Fact Intelligence for litigation case documents** Not checked yet

NEW 55

Fact Intelligence processes uploaded case documents upfront to extract reported facts, identify people and dates, and connect related information across documents with links to original sources. It enables litigation teams to build source-linked chronologies, trace conflicting witness accounts, and test arguments against the evidence record, and now works alongside Legal Research so lawyers can assess how law applies to the specific facts extracted from their case.

— Describes mechanism and integration but no UI path or command. launch-20261123-a3091bcf

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ ML Platforms & SDKs

Google Gemini Enterprise Agent Platform

Feedback ⓘ changelog · Nov 28, 2026 CHANGELOG ↗ RANK

Gemini Enterprise Agent Platform provides Google's managed model training, inference, evaluation, and agent deployment services, formerly offered under Vertex AI.

> Gemini Enterprise Agent Platform's biggest move this window was general availability of Gemini 4 Neon with a 1,048,576-token context window and broad agentic capabilities, alongside a defensive-cybersecurity model variant, new CodeMender preview models, voice remixing, and GA supervised fine-tuning for Gemini 3.8 Flash.

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

01 **New CodeMender preview models: Gemini 4 Neon and Hydrogen**

Not checked yet NEW 76

CodeMender now supports two new preview models: pass `--model gemini-4-neon` to use Gemini 4 Neon in Preview, where cache writes begin billing January 12, 2027, or pass `--model gemini-4-hydrogen` to use Gemini 4 Hydrogen in Preview. Gemini 3.8 Flash remains the default model in CodeMender.

— Exact CLI flags and billing date given, mechanism beyond that is thin

changelog-20261128-3378bb26

02 **Gemini 4 Neon reaches general availability** Not checked yet NEW

70

Gemini 4 Neon is now GA on the Gemini Enterprise Agent Platform, supporting a 1,048,576-token input and output limit with long decodes up to 1,048,576 tokens, function calling, code execution, computer use, URL context, and agentic video understanding. It supports grounding via Google Search, Google Maps, Web Grounding for Enterprise, Parallel and Exa Web Search.

— Rich spec list but no exact selection command or endpoint given

changelog-20261128-3378bb26

03 **Voice remixing via prompted.base_voice in Voices API**

Not checked yet NEW 69

Users can remix an existing voice into a new custom voice by setting `prompted.base_voice` in a Voices API `create` request, supplying a natural-language description of desired changes such as age, pitch, energy, or accent.

— Exact API field and request type named, ready to call changelog-20261128-3378bb26

04 **Cybersecurity-approved Gemini 4 Neon model GA** Not checked yet

NEW 67

The new model `gemini-4-neon-cyber-permissive` is now GA for defensive cybersecurity workloads, available on the OpenAI compatibility Chat Completions API, with Provisioned Throughput at 345 tokens per second per GSU and a 1-GSU minimum purchase increment.

— Model name, API, and throughput numbers given but no sample call

changelog-20261128-3378bb26

THINNER COVERAGE BELOW

05 **Supervised fine-tuning GA for Gemini 3.8 Flash** Not checked yet

NEW 35

Supervised fine-tuning for Gemini 3.8 Flash, including continuous tuning and tuning checkpoints, is now generally available.

— Names the capability but no steps, config, or endpoint changelog-20261128-3378bb26

WAS THIS USEFUL? 👍 👎

Hugging Face

Feedback ⓘ changelog · Nov 28, 2026 CHANGELOG ↗

RANK

Hugging Face is an open platform for building, sharing, and deploying machine-learning models, datasets, and applications.

> Hugging Face introduced a new Home feed on the logged-in homepage that blends personal activity, followed accounts, and trending content.

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

01 **Home feed on logged-in homepage** Not checked yet NEW 60

The logged-in homepage now defaults to a Home feed combining your own activity, activity from people and organisations you follow, trending model releases, and trending contributors, with each release grouped into a single card. The previous Following feed remains accessible via a dropdown at the top of the feed.

View your personalised Home feed — combining activity from people you follow, trending models, and trending contributors — after logging in.

📍 1. Log in to huggingface.co. 2. On your homepage, the default tab is now Home. 3. To switch back to the original view, open the dropdown at the top of the feed and select Following.

— Explains behaviour and exact UI navigation, no technical internals.

changelog-20261128-9e2b3bcb

WAS THIS USEFUL? 👍 👎

SAM (Agent Mesh) Feedback ⓘ v0.3.0 NOTES ↗ RANK

SAM is a private network for AI agents that lets each node publish, discover, and call tools and models across machines under identity-bound, deny-by-default policy.

> SAM v0.3.0 adds OIDC claim-based policy bindings with claim attestation into mesh credentials, ships the Agent Mesh Connect browser web app with a local-agent WebMCP mode, and enables agentmesh-node as the enforcement point for Google Cloud Agent Gateway via ext_proc, alongside a run of breaking changes tightening task-scoped credential enforcement, routing precedence, border-JWT audiences, and SDK compatibility.

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

01 Claim-based policy bindings and attestation Not checked yet NEW

90

`PolicyBinding.claims` in `api/agentmesh.proto` lets a policy binding apply only to identities whose verified OIDC token contains specified claim values, enabling per-issuer, device-posture, and `amr` `-based` access rules; `POST /policies` enforces a limit of 16 claim paths. `PolicyConfig.attested_claims` lists additional OIDC claim paths the control plane writes as claim(key, value) facts into member and delegated session Biscuits at `/register`, `/refresh`, `/token/exchange`, and `/oauth/token`.

Restrict a policy binding to identities from a specific OIDC issuer with a verified device-posture claim, without relying on node-declared labels.

```
$ POST /policies
{
  "bindings": [
    {
      "members": ["group:eng"],
      "claims": {
        "iss": "https://accounts.example.com",
        "device_managed": "true"
      },
      "role": "caller"
    }
  ],
  "attested_claims": ["iss", "amr", "device_managed"]
}
```

— Named proto fields and endpoints with a working example. v0.3.0

02 **Task-scope attenuation in node config** Not checked yet NEW

85

Add `require_task_scope` and `max_task_ttl` to the `attenuation` section of `agentmesh-node.yaml` to enforce task-scoped credentials and cap their lifetime on inbound mesh, external ingress, and Envoy gateway requests.

Cap task-credential lifetime and require task scope on all inbound requests to the node, preventing long-lived or unscoped credentials from reaching services.

```yaml
attenuation:
  require_task_scope: true
  max_task_ttl: 15m
```

— Exact config keys and a runnable snippet given. v0.3.0

03 **Google Cloud Agent Gateway enforcement via ext_proc**

Not checked yet NEW

85

Enable `agentmesh-node` as the policy enforcement point for Google Cloud Agent Gateway as an `ext_proc` Service Extension (`AuthzExtension` with `EXT_PROC_GRPC` , or `AuthzPolicy` with `CONTENT_AUTHZ`), so the node verifies Biscuits and task rules while the gateway owns the data path. `ext_proc` reads an MCP body only in `BUFFERED` or `FULL_DUPLEX_STREAMED` mode (requiring `allow_mode_override: true` on the Envoy filter); in half-duplex `STREAMED` mode a body split across chunks is refused with `403` , and a body above 1 MiB receives `413` . The gateway must present the node's credential over a Unix socket (`--socket-path`) or a client certificate (`--tls-ca`); any other transport receives `403` .

— Extensive named flags and modes, no single runnable command. v0.3.0

04 **Agent Mesh Connect browser app and local agent mode**

Not checked yet NEW

75

New `agentmesh-webapp` container (`ghcr.io/google/agentmesh-webapp`) and Helm chart (`charts/agentmesh-webapp`) serve Agent Mesh Connect, a browser web app that joins the mesh as a full member with its own key, credential, and Node ID using WebSocket or WebTransport; its local agent mode on the Config tab connects agents such as Claude Code to node caller tools through WebMCP and a relay requiring only Node.js 22, with single-use token pairing and a 60-second signed-call window.

— Container image and chart named but no walkthrough. UI entry point named but no exact setup steps.

v0.3.0

05 **Mesh identity headers and border JWT claim forwarding**

Not checked yet NEW 75

The node now sets `X-Mesh-Claims`, `X-Mesh-Labels`, `X-Mesh-Actor-Node`, and `X-Mesh-Origin` headers from the verified credential, forwarding the first three to egress destinations configured with `forward_context`; border JWTs issued from `/sts/token` carry `mesh_claims` and `mesh_labels`. The audience of a border JWT now comes from policy: `/sts/token` and the node's `/oauth/token` refuse a requested audience that differs from the audience of the egress policy.

— Named headers and endpoints without a usage example. v0.3.0

---

06 **Task-scoped credential enforcement tightened** Not checked yet

75

BREAKING

Under a credential with `operation.allowed_tools`, `resources/list` and `prompts/list` on HTTP now return `403`, and a tool name with a `scheme://service/` prefix or with spaces around it is denied. A task denial on an MCP stream now returns JSON-RPC error `1001` instead of `-32600`. Envoy's `ext_authz` filter must set `with_request_body`, or the node refuses any `POST` to an MCP service from such a credential.

— Exact status and error codes named for migration. v0.3.0

THINNER COVERAGE BELOW

07 **Service selection precedence for ext_authz and ext_proc**

Not checked yet 55

BREAKING

The host of a request now selects the service before a bare path: `ext_authz` and `ext_proc` evaluate `X-Mesh-Target-Service`, then `/mesh/<scheme>/<name>/`, then the request host, and last a bare `/<scheme>/<name>/` path.

— Precedence order named, no migration example given. v0.3.0

---

08 **Control plane database schema v22** Not checked yet BREAKING

55

The control plane database schema is now version 22; do not downgrade a migrated database to v0.2.0, since the old binary ignores `claims` on a binding and a policy save from it removes the claims.

— Clear migration warning, no further mechanism. v0.3.0

---

09 **SDK compatibility breaking changes** Not checked yet BREAKING

45

The Python SDK now requires `libp2p>=0.8.1`. The JavaScript SDK's connect() error message text changed: a failed connect() now throws `cannot reach <peer>:` with

one line per path tried.

— Dependency and message changes noted, no deeper detail. v0.3.0

└──▷ BREAKING ON UPGRADE

! Envoy `ext_authz` must set `with_request_body` on the `ext_authz` filter; without it, the node refuses any `POST` to an MCP service from a credential that has `operation.allowed_tools`.

! A gateway must use the Unix socket (`--socket-path`) or a client certificate (`--tls-ca`) to receive a credential from the node; requests on any other transport receive `403`.

! `ext_proc` reads an MCP body in `BUFFERED` or `FULL_DUPLEX_STREAMED` mode only; `allow_mode_override: true` must be set on the Envoy filter. In half-duplex `STREAMED` mode, an MCP body arriving in more than one chunk is refused with `403`. A body above 1 MiB receives `413`.

! The host of a request now selects the service before a bare path: `ext_authz` and `ext_proc` evaluate `X-Mesh-Target-Service`, then `/mesh/<scheme>/<name>/`, then the request host, and last a bare `/<scheme>/<name>/` path.

! The audience of a border JWT comes from policy: `/sts/token` and the node's `/oauth/token` refuse a requested audience that differs from the audience of the egress policy.

! Under a credential with `operation.allowed_tools`, `resources/list` and `prompts/list` on HTTP now return `403`. A tool name with a `scheme://service/` prefix or with spaces around it is denied.

! A task denial on an MCP stream now returns JSON-RPC error `1001` instead of `-32600`.

! The control plane database schema is now version 22; do not downgrade a migrated database to v0.2.0 as the old binary ignores `claims` on a binding and a policy save from it removes the claims.

! The Python SDK now requires `libp2p>=0.8.1`.

! The JavaScript SDK's connect() error message text changed: a failed connect() now throws `cannot reach <peer>:` with one line per path tried.

WAS THIS USEFUL? 👍 👎

MotleyAI SLayer

Feedback ⓘ v2.1.0 NOTES ↗ RANK

SLayer is an embeddable semantic layer that lets AI agents and people query warehouse data through shared metric definitions.

> SLayer v2.1.0 adds Ossie ontology import and validation, time-zone-aware querying with a new time-of-day type, window functions, several new aggregations and scalar functions, alongside a set of breaking changes to time, window-function and metric-dependency behavior.

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

01 Ossie ontology import, search and validation

Not checked yet

NEW

90

`slayer import-osi` imports Ossie ontology and mapping documents instead of refusing them; ontologies are checked, stored, exportable, and manageable through Python, REST, CLI, and MCP. Datasets, fields, and relationships can be searched by ontology concept using `concept:<Concept>` in the `search` command, and `validate_datasets` reports where your model or data disagrees with the ontology.

Import an Ossie ontology document so concepts and mappings are searchable and validatable across your model.

```
$ slayer import-osi path/to/ontology.osi.yaml
```

— Names command, search syntax and validation tool with a runnable example. v2.1.0

02 Time-of-day type and current_time keyword change

Not checked yet

88

BREAKING

A new `Time` type supports grouping, filtering, ordering, `min` / `max`, extraction of `hour` / `minute` / `second`, and literals `TIME '08:30:00'`, current_time(), and CAST(x AS TIME) on every supported database (ClickHouse requires server 25.6 or later); database `TIME` columns now ingest as `Time`. Breaking: `Time` fields and casts to `TIME` now return time-of-day values instead of the driver's raw value, so a re-ingest re-types stored `DateTime` / `String` fields over `TIME` columns to `Time`; `current_time` is now a keyword, so a field with that name must be quoted, and a stored custom granularity named `current_time` is renamed `current_time_granularity` on upgrade, which stores the datasource at a new version that SLayer 2.0.x cannot read.

— Full mechanism, named literals and casts, explicit migration impact. v2.1.0

03 Time-zone-aware queries

Not checked yet NEW

82

Set `time_zone` on a query, datasource, or individual field to bucket, compare, and return results in that zone, with now(), current_date(), and relative dates all reading the clock in that zone; use `civil` for wall-clock values.

Run a query in a specific time zone so bucketing, relative dates, and now() all reflect that zone rather than the server clock.

```
$ slayer query '{"population": "orders", "metrics":
["sum(orders.order_total)"], "dimensions": ["date_trunc(month,
orders.created_at)"], "time_zone": "America/New_York"}'
```

— Mechanism and runnable query example given, minor gaps on edge cases. v2.1.0

---

04 **New aggregation functions and exact ClickHouse percentiles**

Not checked yet NEW 65

New aggregations `sum_distinct`, `avg_distinct`, `percentile_disc`, and `percentile_approx` are now available, also expressible as SUM(DISTINCT x), PERCENTILE_DISC(p) WITHIN GROUP (ORDER BY x), and APPROX_PERCENTILE(x, p); `percentile` and `median` are now exact on ClickHouse.

— Names every function and syntax form but no usage example. v2.1.0

---

05 **Window functions in fields, metrics and queries** Not checked yet

65

BREAKING

Window functions with OVER (PARTITION BY ... ORDER BY ... ROWS/RANGE ...) now work in field definitions, metrics, and queries, including portable Ossie fields and metrics. Breaking: a field defined by a window function is now computed over its dataset's rows, so raw-row queries that filter on other columns may return different window values than before.

— Explains syntax and a concrete behavior change, no example query. v2.1.0

---

06 **New scalar functions across types** Not checked yet BREAKING

63

New scalar functions `IF`, `NVL`, `NVL2`, `STARTSWITH`, `ENDSWITH`, `CHARINDEX`, `TRUNCATE`, `TO_DATE`, `TO_TIMESTAMP`, `zeroifnull`, `nullifzero`, trunc(x, d), `radians`, `degrees`, `regexp_count`, `TIMESTAMP_NTZ` literals, and the `millisecond` date part are now supported. Breaking: round(x, d) and trunc(x, d) now require an integer `d`.

— Every function named verbatim but no mechanism beyond listing. v2.1.0

THINNER COVERAGE BELOW

07 **Dependency protection for model-scoped metrics** Not checked yet

48

BREAKING

Deleting or changing a dataset, field, or metric that a stored model-scoped metric reads is now refused, listing what it would break; such broken metrics already in storage load as unloadable.

— Describes behavior change but no named flag or command. v2.1.0

---

08 **first/last ranking key behavior change** Not checked yet BREAKING

37

A `first` / `last` whose ranking key is also a query dimension now returns one value per coarser cell, and an associated `first` / `last` supplies rows like `sum`.

— States the behavior change but offers no example or migration path. v2.1.0

└──▷ **BREAKING ON UPGRADE**

! round(x, d) and trunc(x, d) now require an integer `d`.

! `current_time` is now a keyword; a field with that name must be written quoted. A stored custom granularity named `current_time` is renamed `current_time_granularity` on upgrade.

! `Time` fields and casts to `TIME` now return time-of-day values instead of the driver's raw value; a re-ingest re-types stored `DateTime` / `String` fields over `TIME` columns to `Time`.

! A `first` / `last` whose ranking key is also a query dimension now returns one value per coarser cell, and an associated `first` / `last` supplies rows like `sum`.

! Deleting or changing a dataset, field, or metric that a stored model-scoped metric reads is now refused, listing what it would break; such broken metrics already in storage load as unloadable.

! A field defined by a window function is now computed over its dataset's rows, so raw-row queries that filter on other columns may return different window values than before.

! Datasources with a renamed custom granularity are stored at a new version that SLayer 2.0.x cannot read.

WAS THIS USEFUL?  

TileLang

Feedback ⓘ v0.1.16 NOTES ↗

TileLang is a Python-based domain-specific language and compiler for writing optimized GPU and CPU kernels.

TileLang adds Ascend 950 PTO backend, process-based parallel JIT compilation, NVRTC device compilation, OpenMP CPU parallelization, and new CUDA L1/L2 prefetch and clamp intrinsics.

└──▷ GET THIS VERSION ▾

pip

```
$ pip install tilelang==0.1.16
# already installed? upgrade to this version:
$ pip install --upgrade tilelang==0.1.16
```

git

```
$ git clone --branch v0.1.16 https://github.com/tile-ai/tilelang.git
# already have the repo? check out this version:
$ git checkout v0.1.16
```

PyPI attestation · checked 28 Nov Wrong command?

Add zero-copy host-mapped tensor parameters on CUDA and Ascend with `T.annotate_host_mapped_buffers`. Not checked yet

Prefetch data to L1 or L2 cache on CUDA using `T.prefetch_l1`, `T.prefetchu_l1`, and `T.prefetch_l2` with optional eviction policies. Not checked yet

Clamp values symmetrically on SM86+ with `T.clamp_symmetric` (using `min.NaN.xorsign.abs` semantics). Not checked yet

Cast to TF32 with round-to-nearest-even rounding using T.cast(..., round="rn"). Not checked yet

Compile kernels in parallel using separate processes with tilelang.par_compile(..., executor="process"). Not checked yet

+6 more

NEW SUPPORTED PLATFORM gfx1200 FP16/BF16 WMMA linear shared layouts and ROCm 10 support

NEW SUPPORTED PLATFORM Windows ARM64 build and runtime support

NEW SUPPORTED PLATFORM SM120/SM121 block-scaled MMA and sm_103a/sm_107a targets in CuTeDSL: GB10

WAS THIS USEFUL? 👍 👎

Snowflake Cortex AI

Feedback ⓘ announced · Nov 24, 2026

BLOG ↗

Snowflake Cortex AI provides managed language-model functions, search, and agents that work with data governed in Snowflake.

Snowflake Horizon Catalog adds Access History and IRC telemetry so external-engine reads, writes, and failures appear in existing Snowflake audit views with no extra infrastructure.

NEW INTEGRATION Query external-engine Iceberg activity alongside Snowflake SQL in `SNOWFLAKE.ACCOUNT_USAGE.ACCESS_HISTORY` by filtering on `event_source = 'horizon_irc'`; reads appear in `direct_objects_accessed` and writes in `object_modified_by_ddl`, with 365-day retention on Enterprise Edition and higher. Not checked yet

Distinguish IRC traffic from Snowflake SQL using the `event_source` column (`snowflake_sql` vs `horizon_irc`), with Iceberg operation details such as `LoadTable`, `CreateTable`, or `UpdateTable` recorded in `additional_properties`. Not checked yet

Track write operations at snapshot level: each write entry in Access History records the inferred change kind (`append`, `overwrite`, `delete`, or `truncate`) and the parent and resulting snapshot IDs. Not checked yet

Audit governance enforcement for external engines: when an external engine reads a policy-protected table via the Scan Plan API PUBLIC PREVIEW, the `policies_referenced` column records which masking and row access policies Snowflake evaluated. Not checked yet

NEW INTEGRATION Enable Horizon Catalog telemetry to record every HTTP request — including failures — made by external engines to Horizon Catalog IRC APIs in your account's event table; set `LOG_LEVEL` to at least `INFO` to activate. Not checked yet

+2 more

WAS THIS USEFUL? 👍 👎

◆ Inference Providers & GPU Clouds

AWS Amazon SageMaker AI

Feedback ⓘ

changelog · Nov 28, 2026 CHANGELOG ↗ RANK

Amazon SageMaker AI provides managed tools and infrastructure for building, training, fine-tuning, and deploying machine-learning models.

> Amazon SageMaker AI expanded HyperPod with new managed IAM policies for joining global inference gateways and increased pod density on EKS-orchestrated clusters via dynamic ENI attachment.

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

01 **Managed policies for HyperPod inference gateways** Not checked yet

NEW 64

Three new AWS managed policies support HyperPod inference gateways joining a global inference gateway: `AmazonSageMakerHyperPodInferenceGatewayGlobalRoutingAccess`, `AmazonSageMakerHyperPodInferenceGatewayMetricsPublisherAccess`, and `AmazonSageMakerGlobalInferenceGatewayExecutionAccess`.

— Names exact policy ARNs but not usage mechanism. changelog-20261128-fad5de5b

THINNER COVERAGE BELOW

02 **More ENIs per node in EKS-orchestrated HyperPod clusters**

Not checked yet IMPROVED 56

HyperPod clusters orchestrated by Amazon EKS can now attach more elastic network interfaces (ENIs) to nodes, increasing the number of pods per node up to the instance type maximum via dynamic ENI attachment.

— Explains mechanism and benefit but no config step given. changelog-20261128-fad5de5b

WAS THIS USEFUL? 👍 👎

AWS Amazon Bedrock

Feedback ⓘ changelog · Nov 28, 2026

CHANGELOG ↗

Amazon Bedrock provides managed foundation-model inference, customization, knowledge bases, and APIs for generative AI applications.

Amazon Bedrock adds Trusted Access Programs with IAM/SCP enrollment controls and expands EFS/EDC to Anthropic models, plus a new automated and human review framework for retained content.

Control enrollment in Trusted Access Programs at the project, account, or organization level using IAM and service control policies, via the new Trusted Access Programs feature. Not checked yet

Store retained content in your own Amazon S3 bucket or in Amazon-managed storage under Enterprise Frontier Safeguards (EFS) and Enterprise Data Controls (EDC) for Anthropic models, now in public preview. Not checked yet

Amazon Bedrock now runs automated review and human review on retained inputs and outputs for models whose provider requires review, with a new topic explaining who reviews cases and how trusted access programs affect retention. Not checked yet

NEW SUPPORTED PLATFORM Watermarking extended to additional Claude models: Claude Sonnet 4.6, Claude Opus 4.6, Claude Opus 4.5, Claude Sonnet 4.5, Claude Haiku 4.5 and 2 more

WAS THIS USEFUL? 👍 👎

◆ Local LLM Runtimes

LM Studio

Feedback ⓘ changelog · Nov 28, 2026 CHANGELOG ↗ RANK

LM Studio is a desktop application that runs local LLMs and provides model discovery, downloads, chat, and a local API server.

> LM Studio's latest release adds per-model engine overrides, Mac-only Splash acceleration, Inkling reasoning/tool-call parsing, and redesigns both the Model Search and LM Link pages.

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

01 **Redesigned Model Search page** Not checked yet IMPROVED

55

Model Search now opens as a full page instead of a modal, with format, hardware fit, and sort filters moved to the left sidebar.

– Describes before/after layout and named filter types changelog-20261128-412e0486

02 **Per-model engine and version overrides** Not checked yet NEW

53

Load settings now let you set per-model engine and version overrides, controlling which inference engine and which version of it each individual model uses.

– Names the settings surface but no exact UI path or options changelog-20261128-412e0486

03 **Redesigned LM Link page** Not checked yet IMPROVED

50

The LM Link page is redesigned so the sidebar lists your networks and the main area shows the selected network's devices alongside the selected device's details and models.

– Describes new layout structure but no config or command changelog-20261128-412e0486

04 **Splash acceleration for GGUF and MLX on Mac** Not checked yet NEW

45

LM Studio can now enable Splash acceleration for compatible GGUF and MLX models on Mac.

– Names formats and platform but no mechanism or how to enable changelog-20261128-412e0486

05 **Inkling reasoning and tool call parsing** Not checked yet NEW

25

LM Studio now parses Inkling reasoning and tool calls.

— Bare statement with no detail on behavior or scope changelog-20261128-412e0486

WAS THIS USEFUL? 👍 👎

vMLX

Feedback ⓘ v1.6.87 NOTES ↗ RANK

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

> vMLX's biggest work this window was deep ERNIE 5.5 MoE support — native model wiring, MXFP8/MXFP4 quantization and GEMV kernels, and chunked MLA prefill with a dedicated prefix cache — alongside Hy4 runtime upgrades (RAM/SSD caching, streaming, vision-language) and new support for Qwen3.5-MoE, bonsai2.5, and holo4 models.

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

01 **ERNIE 5.5 model support and performance** Not checked yet NEW 75

Adds native `ernie5_5_moe` model support with the `JANGHT2.4` and DSpark runtime, including fused JANGH/JANGT SiTU kernels, parsers, and registry wiring, plus barrier-free multi-row MXFP8 2D/3D GEMV at `T=1..8`, `T<=8` parallel-slot JT/JH weighted-down, an ARFallback cold-compile outlier guard, new `ERNIE-5.5 MXFP4` quantization support, 512-strip chunked MLA prefill, and an exact typed `ErnieLayerCache` RAM and SSD prefix cache for ERNIE 5.5.

— Names model id and runtime but no usage command Dense kernel mechanism and named formats, but no user-facing control Names a concrete cache type and strip size, no usage path given

v1.6.86 v1.6.87

THINNER COVERAGE BELOW

02 **jangt model kernel improvements** Not checked yet NEW 55

Adds native NVFP4 and MXFP8 mixed-group support in `JTMixedSwitchGLU` for jangt models, and a soft-capped Inkling activation variant for the trellis kernels in jangt.

— Names two internal kernel additions, no runnable surface v1.6.86

03 **Hy4 runtime caching, streaming, and vision-language support**

Not checked yet NEW 55

The Hy4 runtime gains RAM and SSD cache support enabling prompt-cache persistence across server restarts, per-tool-call streaming, reasoning effort controls, full vision-language support with DFly2 integration, Panel settings parity, and a new Hy4-flash runtime variant.

— Explains persistence mechanism but no config keys or flags v1.6.87

04 **Fused Hadamard/ternary kernels for bonsai2.5 and holo4**

Not checked yet NEW 50

Adds fused Hadamard/ternary kernels, an eval mode, and adaptive native MTP for bonsai2.5 and holo4 models.

— Names models and kernel types but no mechanism detail v1.6.86

05 **Qwen3.5-MoE model support** Not checked yet NEW 40

Now supports Qwen3.5-MoE, with a sanitize fix applied during model loading.

— New model name given, fix described only as a sanitize patch v1.6.86

06 **Inkling-2 bundling and model loading** Not checked yet NEW 40

Inkling-2 is now bundled with DFlash2, and inkling2 adds support for loading JANGHT bundles with per-unit routed formats.

— Names the bundle and format but mechanism stays thin v1.6.87 v1.6.86

07 **Phase 3 long-context kernels** Not checked yet NEW 15

Adds Phase 3 long-context kernels.

— Bare name with no mechanism or scope given v1.6.87

WAS THIS USEFUL? 👍 👎

MLX Serve

Feedback ⓘ v26.11.4 NOTES ↗ RANK

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

> mlx-serve adds support for the Inkling 2 Mini model with multimodal input and speculative decoding, extends Thunderbolt tensor parallel for DeepSeek-V4.1 beyond two Macs, and speeds up long agent sessions on sliding-window models along with several performance and security fixes.

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

01 **Inkling 2 Mini model support** Not checked yet NEW 75

Adds support for Inkling 2 Mini, a 26B looped model that runs from its released checkpoint, an mlx-lm pack, or an oMLX oQ pack, with image input and multi-chat GPU batching. Speculative decoding is provided via a DFlash2 drafter bundled in the model pack, which checks a tree of candidate continuations in one pass per round. On M1–M4 Macs it decodes 10–13% faster and reads prompts ~20% faster; on M5 it decodes long contexts up to 14% faster with an 8-bit KV cache and serves 4–8 concurrent chats 23–31% faster using its bundled draft head.

— Names pack formats and gives concrete per-chip speed figures v26.11.4

02 **Thunderbolt tensor parallel scaling for DeepSeek-V4.1**

Not checked yet IMPROVED 65

Thunderbolt tensor parallel for DeepSeek-V4.1 now spans more than two Macs, with the LM head and DSpark draft split across them, plus a built-in plugin for directional ablation. DeepSeek-V4.1 also reads prompts faster and decodes ~11% faster on M5 Macs.

— Names mechanism and model but no setup command shown v26.11.4

THINNER COVERAGE BELOW

03 **Prompt-position resume for sliding-window agent sessions**

Not checked yet IMPROVED 58

Long agent sessions on sliding-window models (Gemma 4, Inkling 2) now resume from the last prompt position on follow-up turns instead of re-reading the whole conversation, dropping Gemma 4 26B at 80K tokens from 20s to 2s to first token.

— Concrete before/after timing but no config surface named v26.11.4

04 **Fix for `--os-reserve-gib` memory reservation flag**

Not checked yet IMPROVED 45

Re-enables `--os-reserve-gib` to take effect for memory reservation.

— Names exact flag, a runnable fix, but no further mechanism v26.11.4

05 **API key required before request body is read** Not checked yet

IMPROVED 25

Requests without an API key are now refused before their body is read.

— States the behavior change but no endpoint or config detail v26.11.4

WAS THIS USEFUL? 👍 👎

llama.cpp

Feedback ⓘ v1.1.0 NOTES ↗ CODE ↗ RANK

Llama CPP is a C/C++ library and set of command-line tools that runs large language models locally.

> llama.cpp v1.1.0 adds budget-based quantization with resumable state, splits DFlash speculative decoding into four selectable draft methods, expands TTS/audio model and generation support, and adds OpenAI-style named tool_choice plus a new SSE model-listing endpoint to the server.

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

01 **Budget-based quantization with resumable state** Not checked yet

NEW 90

Quantize models to a target bits-per-weight or byte-size budget using `--target-bpw` or `--target-size`, with a resumable `state_file` so interrupted runs can continue. The `llama_model_quantize_params` struct in `llama.h` gains `target_bpw`, `target_size`, `state_file`, and an abort callback.

— Names flags and struct fields but no sample invocation v1.1.0

02 **SSE model listing endpoint** Not checked yet NEW 85

Add `GET /models/sse` endpoint to `llama-server` for server-sent-event model listing in single-model mode, e.g. `curl -N http://localhost:8080/models/sse` to stream model availability events.

Stream model availability events in single-model mode — useful for tooling that needs to react when the server is ready.

```
$ curl -N http://localhost:8080/models/sse
```

— Named endpoint with runnable curl example v1.1.0

03 **DFlash speculative decoding split into four draft methods**

Not checked yet IMPROVED 74

Speculative decoding via DFlash is split into four separate draft methods — `draft-dflash`, `draft-dflash2`, `draft-dspark`, and `draft-lilicorr` — selectable programmatically via new `llama_dflash_variant` and llama_model_dflash_variant() additions to `llama.h`.

— Names all four methods and API but not their behavioural differences v1.1.0

04 **Expanded TTS and audio model support** Not checked yet NEW

70

Add audio types for Zonos3, YuE2, MiniMax-Music3, Fish S2, Moshi, Kokoro, tiny-voca, and VoxCPM2 to `mtmd.h`, along with `n_channels`, mtmd_gen_audio_g2p(), and mtmd_gen_state_get_ref_codes() for audio generation control.

— Names all new model types and functions, lacks usage example v1.1.0

---

05 **`llama_batch` API deprecated for `llama_batch_ext`** Not checked yet

DEPRECATED

63

The `llama_batch` API in `llama.h` is deprecated in favour of `llama_batch_ext` and `llama_process`; callers using the old API should migrate.

— Names old and new APIs but no migration steps v1.1.0

---

06 **GGUF metadata array reading functions** Not checked yet NEW

60

Add llama_model_meta_arr_n() and llama_model_meta_arr_str() to `llama.h` to read GGUF string arrays from model metadata.

— Names the functions but no usage walkthrough v1.1.0

---

07 **Named `tool_choice` support in server** Not checked yet NEW

60

Accept OpenAI-style named `tool_choice` in the server and implement it end-to-end in the chat layer, including passing `reasoning_effort=none` when supported.

— Describes behaviour but no request example v1.1.0

THINNER COVERAGE BELOW

---

08 **UI agent and MCP proxy flags** Not checked yet NEW

50

Add `--ui-agent` and `--ui-mcp-proxy-allow-private-urls` flags to `llama-server`.

— Bare flag names with no explanation of behaviour v1.1.0

└──▷ BREAKING ON UPGRADE

! The `llama_batch` API in `llama.h` is deprecated in favour of `llama_batch_ext` and `llama_process`; callers using the old API should migrate.

WAS THIS USEFUL? 👍 👎

◆ AI / LLM Gateways

OpenRouter

Feedback ⓘ updated · Nov 28, 2026 NOTES ↗ RANK

OpenRouter is a routing gateway that exposes hundreds of models from many providers behind one OpenAI-compatible API with failover and unified billing.

> OpenRouter tightened management key permissions, added Hugging Face inference routing with trace storage, published new RAG reranking guidance, and localized its docs into 8 languages.

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

01 **Reranking guidance for RAG pipelines** Not checked yet NEW 55

New documentation explains reranking in RAG: pass vector search's top 30 candidates through a cross-encoder model to re-score each chunk directly against the query.

— Concrete numbers and technique given as actionable guidance snapshot-20261128

02 **Management key access controls** Not checked yet NEW 45

Management keys can now be restricted to read-only access, limited to a single workspace, or both, so each key carries only the permissions it needs.

— Describes scope but no exact flag or endpoint named snapshot-20261128

03 **Hugging Face inference routing and trace storage** Not checked yet NEW 45

OpenRouter now powers inference routing for Hugging Face Chat models, and can send traces to a Hugging Face Storage Bucket.

— Names integration targets but no setup steps given snapshot-20261128

04 **8-language documentation localization** Not checked yet NEW 20

OpenRouter's documentation is now available in 8 languages.

— Only mentioned in summary, no further detail provided snapshot-20261128

WAS THIS USEFUL? 👍 👎

BerriAI LiteLLM

Feedback ⓘ v1.108.19 NOTES ↗ RANK

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

> LiteLLM's single release this window adds per-deployment deferred tool handling for Claude Code, new Vertex AI Meta Muse models, and ElevenLabs real-time transcription with spend tracking.

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

01 **ElevenLabs real-time transcription support** Not checked yet NEW 65

Adds ElevenLabs `scribe_v2_realtime` transcription on `/v1/realtime` with spend tracking.

– Names endpoint and model but no example call v1.108.18

02 **Deferred tool handling for Claude Code prompt caching**

Not checked yet NEW 60

Adds `deferred_tools` handling per deployment on the Messages API so Claude Code tool loads keep the prompt cache intact on non-Anthropic models.

– Names config surface but lacks setup steps v1.108.18

THINNER COVERAGE BELOW

03 **Meta Muse models on Vertex AI** Not checked yet NEW 35

Adds Meta Muse models on Vertex AI with native Responses API routing.

– Names models and routing but no usage detail v1.108.18

NEW INTEGRATION Volcengine Ark Seedream image generation and editing

v1.108.19 NOTES ↗ LATEST STABLE

└──▷ GET THIS VERSION ▾

Docker

```
$ docker pull ghcr.io/berriai/litellm:v1.108.19
```

```git
$ git clone --branch v1.108.19
https://github.com/BerriAI/litellm.git
# already have the repo? check out this version:
$ git checkout v1.108.19
```

published under the project's GitHub account · checked 28 Nov Wrong command?

WAS THIS USEFUL? 👍 👎

Mozilla AI Otari

Feedback ⓘ v0.58.0 NOTES ↗ RANK

Mozilla AI Otari is an open-source AI gateway you run yourself. One endpoint for 40+ providers, with virtual keys, budgets, and usage tracking.

> Otari v0.58.0 adds audit-change attribution with before/after values surfaced on request spans, expands the model catalog with hosted-provider traffic visibility, data-location filtering, and richer metadata, and ships a breaking consolidation of identity address-proof logic.

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

01 **Model catalog gains provider visibility, location filtering, metadata**

Not checked yet NEW 55

The dashboard now shows what hosted providers served requests apart from your own keys, the catalog can filter and list models by hosted data location, and model listings now include each model's description, release date, and knowledge cutoff.

— Three concrete catalog/dashboard additions, no config keys named. v0.58.0

02 **Consolidated identity address-proof logic** Not checked yet

55

BREAKING

Address-proof logic is now decided in one place (PR #3595); any code or integration relying on the previous distributed address-verification behaviour may break on upgrade.

— Names the PR and gives a clear migration warning. v0.58.0

03 **Audit-change attribution on request spans** Not checked yet NEW

45

Otari now records who changed an audited setting and what the previous value was, surfacing this information on the request span.

— Describes mechanism but no named config key or API. v0.58.0

04 **Role-based onboarding steps** Not checked yet IMPROVED 30

Onboarding now shows operators and members only the steps relevant to their role.

— Thin one-line description, no specifics on roles affected. v0.58.0

05 **Configurable web app manifest description** Not checked yet NEW

30

Builds can now set the web app manifest's description.

— Minimal detail, no build flag or file path named. v0.58.0

06 **Settled invitation limits for organizations** Not checked yet NEW

25

Organizations now have settled invitation limits with supporting services.

— Vague summary, no limit values or service names given. v0.58.0

└──▷ **BREAKING ON UPGRADE**

! Identity: address-proof logic is now decided in one place (PR #3595); any code or integration relying on the previous distributed address-verification behaviour may break on upgrade.

WAS THIS USEFUL? 👍 👎

◆ Inference Serving & Optimization

Niko1221 Strata

Feedback ⓘ v0.2.23 NOTES ↗

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 adds custom local model loading, a DFlash 2 speculative drafter replacing DSpark, fast mode on by default for more GPU families, and new CLI flags for fine-grained inference control.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v0.2.23 https://github.com/Niko1221/Strata.git
# already have the repo? check out this version:
$ git checkout v0.2.23
```

Load your own fine-tunes, merges, or third-party quantizations from the app's local-files screens or with `strata launch --model-path` and `--custom-label`; supports `.gguf` for Qwen3.8-Flash-Next and Qwen3.6-35B-A3B, `.strata` for HY-4 Flash, and both formats for Inkling-2-Mini. Not checked yet

NEW SUPPORTED PLATFORM Strata adds DFlash 2 as a new speculative drafter (replacing DSpark), enabled by default on Radeon RDNA 2/3/4, R9700, Strix Halo, Apple M5 Max with the SQ-20 v2 file, and NVIDIA cards by VRAM size. Not checked yet

Control DFlash 2 speculative drafting per-run with `--draft on` or `--draft off`; opt into turbo decode with `--turbo` or `--turbo-max` to skip weakest expert picks in late layers. Not checked yet

Fast mode is now on by default on the RTX 5090, RTX 5060 Ti 16 GB, RTX 5070 12 GB, RTX PRO cards, DGX Spark, and Radeon R9700. Not checked yet

Turn fast mode off with `STRATA_INK_SM120=0`, or keep fast mode while preserving exact output words with `STRATA_INK_SM120_EXACT=1`; opt in to compact hybrid fast path on RTX 5060 8 GB with `STRATA_INK_SM120_COMPACT_HYBRID=1`, Intel Arc Alchemist fast set with `STRATA_INK_INTELARC=1`, and Mac speed mode with `STRATA_INK_M5MAX=1`. Not checked yet

+2 more

NEW SUPPORTED PLATFORM Olares container deployment: ghcr.io/niko1221/strata:0.2.23-cuda13

NEW SUPPORTED PLATFORM No-AVX2 experimental Linux engine: strata-engine-0.1.70-linux-x86_64-cuda12-noavx2-experimental.tar.gz

WAS THIS USEFUL?  

Intel AutoRound

Feedback ⓘ v0.17.0 NOTES ↗ RANK

AutoRound quantizes language and multimodal models through weight-rounding optimization and exports them for inference engines.

> AutoRound's v0.17.0 release centers on Recurrent Residual Quantization (RRQ), which lets one checkpoint serve multiple bit-widths without re-quantizing, alongside broad NVFP4/MXFP4 format support, a new TEQ algorithm and named tuning recipes, a W4A8 kernel for XPU MoE inference, and a security-hardening change disabling `trust_remote_code` by default.

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

01 Recurrent Residual Quantization (RRQ)

Not checked yet NEW

95

RRQ is a new quantization algorithm invoked via `--backend_format auto_round:rrq`, producing a single checkpoint that serves 2/4/6/8-bit and per-layer mixed precision selectable at load time with load_rrq_model(..., active_bits=<n>), without needing to re-quantize the model.

Produce a multi-bit RRQ checkpoint for Qwen3-0.6B, then load it at 4-bit precision without re-quantizing — useful when you need to serve the same model at different precisions.

```
$ python -m auto_round.cli.main \
    --model Qwen/Qwen3-0.6B --scheme W2A16 \
    --backend_format auto_round:rrq --dump_config
```

Load an RRQ checkpoint at 4-bit active precision at inference time — no separate quantization run needed.

```python
from auto_round import load_rrq_model
model = load_rrq_model('./out/Qwen/Qwen3-0.6B/8bit',
active_bits=4)
```

— Named flag, API call and runnable before/after examples v0.17.0

02 TEQ algorithm and named tuning recipes

Not checked yet NEW

75

Adds the TEQ (Transformation-based Equivalent Quantization) algorithm, selectable at the CLI or API level, and named recipes to skip manual hyper-parameter tuning: `--`

`recipe auto-round-best` or `--recipe auto-round-light` in the CLI, or AutoRound(..., recipe='auto-round-best') in Python.

Use the `auto-round-best` recipe in the CLI to get the highest-accuracy preset without manually tuning hyper-parameters.

```
$ python -m auto_round.cli.main \
    --model Qwen/Qwen3-0.6B --recipe auto-round-best
```

— Named flags and a runnable CLI example v0.17.0

03 **NVFP4/MXFP4 quantization format support** Not checked yet NEW

75

Adds NVFP4/MXFP4 quantization format support: quantize and export SVDQuant NVFP4 checkpoints for vLLM-Omni; export MXFP4 checkpoints to SVDQuant Omni format; convert LLMC/ModelOpt/Quark-format NVFP4 weights to MXFP4; and add the `NVFP4_E5M3_GS` data-type scheme with global scale support for NVFP4 quantization.

— Names formats and schemes but no runnable command shown v0.17.0

04 **trust_remote_code disabled by default** Not checked yet BREAKING

60

`trust_remote_code` is now disabled by default, reducing the attack surface when loading untrusted model repositories; workflows that relied on it must now pass `trust_remote_code=True` (or the equivalent CLI flag) explicitly.

— Clear migration note with the exact setting to pass v0.17.0

THINNER COVERAGE BELOW

05 **W4A8 ARK kernel for XPU MoE inference** Not checked yet NEW

55

Accelerates MoE inference on Intel XPU with a new W4A8 ARK kernel (int4 weight / int8 compute) covering both prefill and decode.

— Describes mechanism but no usage example given v0.17.0

06 **Data-parallel block tuning CLI flag** Not checked yet NEW

55

Run data-parallel block tuning across multiple GPUs with the `--data_parallel` CLI flag.

— Named flag is runnable but mechanism undescribed v0.17.0

07 **Nemotron-H Omni model wrappers** Not checked yet NEW 40

Adds new model wrappers to quantize Nemotron-H Omni family models (Nemotron 3.5 Super VL, Bolt 74B-A9B).

— Names specific models but no usage details v0.17.0

---

08 **Tensor parallelism and LoRA tuning for FP4 MoE** Not checked yet

NEW

25

Supports tuning large language models with tensor parallelism and LoRA for FP4 MoE workloads.

— Thin prose description with no flags or commands v0.17.0

└──▷ **BREAKING ON UPGRADE**

! `trust_remote_code` is now disabled by default; any workflow that relied on it being enabled must now pass `trust_remote_code=True` (or the equivalent CLI flag) explicitly.

WAS THIS USEFUL? 👍 👎

FlashInfer

Feedback ⓘ v0.7.6 NOTES ↗ RANK

FlashInfer provides GPU kernels for language-model inference, including attention, sampling, and quantized operations.

> FlashInfer v0.7.6 adds NVFP4 sparse-MLA prefill for DeepSeek-V4 on Blackwell, makes attention, sampling and MoE kernels batch-invariant, and extends MoE and weight-localization support to dual-die Rubin GPUs, while stabilizing QSA sparse paged attention and requiring a newer cuDNN Frontend.

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

01 **Batch-invariant execution for attention, sampling and MoE**

Not checked yet NEW 93

Linear attention becomes batch-invariant when `batch_invariant=True` is passed to `recurrent_kda` and `chunk_gated_delta_rule`, so a token produces the same bits across prefill, chunked prefill and decode. Token sampling is made batch-invariant via `batch_invariant=True` with a per-row `seed` and `offset` passed to `sampling_from_probs`, `sampling_from_logits`, and the top-k, top-p and min-p samplers. `BatchInvariantFp8PerTensorConfig` is added to `MoELayer`, and `batch_invariant=True` is added to the expert-parallel MegaMoE for BF16, NVFP4, MXFP4, MXFP8, W4A16 and W8A16 weights on SM100, SM103 and SM107.

Reproduce identical sampled tokens across prefill and decode positions — useful for deterministic speculative decoding or audit logging — by keying each row on a fixed seed and offset.

```python
import torch
import flashinfer

# probs shape: [batch, vocab]
probs = torch.softmax(torch.randn(4, 32000, device='cuda'),
dim=-1)
seed = torch.tensor([1, 2, 3, 4], dtype=torch.int64,
device='cuda')
offset = torch.zeros(4, dtype=torch.int64, device='cuda')

tokens = flashinfer.sampling_from_probs(
    probs,
    batch_invariant=True,
    seed=seed,
    offset=offset,
)
print(tokens)  # same values regardless of batch position or
prefill/decode phase
```

— Explains mechanism, scope across three kernel types, with a runnable example. v0.7.6

02 **qsa_sparse_paged_attention stabilized, qsa_nvfp4 removed**

Not checked yet BREAKING 90

Graduates `qsa_sparse_paged_attention` (gated sparse paged attention for Qwen QSA models) to a stable API, prebuilt in the wheel for SM80 through SM90. The experimental `flashinfer.experimental.qsa_nvfp4` is removed; callers must switch to `qsa_sparse_paged_attention` (with `backend='cake'` or `FLASHINFER_ALLOW_EXPERIMENTAL_AUTO_BACKENDS=1` for NVFP4 KV cache on SM100, SM103, SM107 or SM120).

— Exact migration path with flag, env var and supported architectures. v0.7.6

03 **NVFP4 sparse-MLA prefill for DeepSeek-V4 on Blackwell**

Not checked yet NEW 86

Adds NVFP4 sparse-MLA prefill for DeepSeek-V4 on Blackwell (SM100/SM103) via trtllm_batch_decode_sparse_mla_dsv4(..., kv_cache_format='nvfp4'), with queries packed first using `cake_dsv4_nvfp4_pack_query`.

Run DeepSeek-V4 sparse MLA prefill over an NVFP4 KV cache on a Blackwell GPU to serve 256K-context compressed layers at lower memory bandwidth.

```python
import flashinfer

# pack the query into the NVFP4 layout expected by the kernel
packed_q = flashinfer.cake_dsv4_nvfp4_pack_query(q)

output = flashinfer.trtllm_batch_decode_sparse_mla_dsv4(
    packed_q,
    kv_cache,
    kv_cache_format='nvfp4',
    # ... other existing arguments
)
```

— Names exact function, args and target GPUs with a runnable example. v0.7.6

---

04 **Dual-die weight and decode-state localization on Rubin**

Not checked yet NEW 73

Splits NVFP4 MoE expert weights across dual-die Rubin dies with `localized_weights=` and `split_fp4_moe_weights_by_locality_domain`, and MXFP8 MoE with `localized_options={'fc2_split': 'k'}`. Also localizes FP8 grouped GEMM weights per die on dual-die SM107a via `cake_batch_deepgemm_fp8.localize_weights`, and Gated DeltaNet decode state via `localize_gdn_decode_state`.

— All named functions/kwargs listed but no example or numbers given. v0.7.6

---

05 **Token-sparse NVFP4 KV cache conversion** Not checked yet NEW 72

Converts only selected NVFP4 KV tokens to FP8 for token-sparse attention — performed once per request and safe inside a CUDA graph — via `gather_nvfp4_kv_cache_by_position` and `compact_nvfp4_kv_cache`.

— Names both functions and the use case but no usage example given. v0.7.6

---

06 **NVFP4 and MXFP8 MoE support on Rubin (SM107)** Not checked yet NEW 71

Adds NVFP4 MegaMoE expert-parallel MoE on Rubin (SM107) through `MoEEpLayer` using the `sm107_nvfp4_nvfp4_bf16_cake` backend, and accepts `quant_mode='mxfp8'` in the CuTe DSL fused MoE on SM107, including ClampedRelu2.

— Names classes and backends but no runnable example. v0.7.6

THINNER COVERAGE BELOW

---

07 **cuDNN Frontend ≥1.33.0 now required** Not checked yet

BREAKING

53

FlashInfer now requires cuDNN Frontend ≥1.33.0; projects that pin `nvidia-cudnn-frontend` below 1.33 must lift the cap.

— Clear dependency bump but no further detail on impact. v0.7.6

└──▷ **BREAKING ON UPGRADE**

! `flashinfer.experimental.qsa_nvfp4` is removed; call `qsa_sparse_paged_attention` instead (with `backend='cake'` or `FLASHINFER_ALLOW_EXPERIMENTAL_AUTO_BACKENDS=1` for NVFP4 KV cache on SM100, SM103, SM107 or SM120).

! cuDNN Frontend ≥1.33.0 is now required; projects that cap `nvidia-cudnn-frontend` below 1.33 must lift the cap.

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

dosco GraphJin

Feedback ⓘ v3.27.0 NOTES ↗ RANK

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

> GraphJin's recent releases rework its storage and access-control internals, add a browser-based first-run setup with Claude Code onboarding, introduce an explicit API service mode, and expand admin console controls for users and table-level permissions.

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

01 Sources config and access control changes

Not checked yet

90

BREAKING

Databases are now declared as sources using a `sources:` block, with fields `name`, `kind`, `default`, `type`, `host`, and `dbname`, replacing the now-refused `database:` section — config files still using `database:` are refused on startup. Restrict agent-server read access per account by setting `access: read: account` on a database source in `sources:`; the agent server now allows every signed-in person to read every table by default, removing the previous implicit per-account row restriction, which can be restored with `access: read: account` if needed. An `access:` block on a database source is an error in `mode: api` — move its rules under `roles`, since the API service no longer auto-generates per-account row-level access rules using `account_id`; such limits must be written explicitly under `roles` before upgrading. On an agent server with accounts, clients that connect without signing in are now refused entirely, whereas previously they connected but every tool call was refused.

— All config fields named, migration path explicit Names config keys and exact before/after security behavior

v3.23.0

02 Explicit API service mode and project scaffolding

Not checked yet

NEW

85

Create a new API service project with `deeporg serve new my-api`, which starts DeepORG from that folder using the project's own config files; choose API service mode explicitly with `mode: api` in config (the older `mode: prod` remains equivalent). In this mode, `production: true` has no effect on the agent server (it is meaningful only in the API service), and setting `mcp.disable: false` in a production API service config no longer enables the MCP server.

Scaffold a new API service project and start it immediately, keeping agent and MCP features out of a pure database-API deployment.

```
$ deeporg serve new my-api && cd my-api && deeporg
```

— Runnable command plus named config keys and caveats v3.23.0

03 **Diskless operation and internal storage changes** Not checked yet

75

BREAKING

GraphJin servers can now store all internal data in Postgres and run without any local disk. DeepORG now stores its own files and encrypted credentials in `~/.deeporg/` and auto-creates its internal tables on each start, removing the need for `deeporg serve db store init`. The `deeporg serve db store init` and `deeporg serve db store status` commands are removed, and the `artifacts.auto_init` setting is removed — if `auto_init: false` is present in config, DeepORG refuses to start.

— Names removed commands and config key, migration implied v3.23.0 v3.26.0

04 **Claude Code connection via mcp add** Not checked yet NEW

75

Connect Claude Code to DeepORG without copying a secret token using `deeporg mcp add claude <url>`, which opens a browser prompt to approve the connection.

Connect Claude Code to your DeepORG instance without sharing a secret — the URL alone is enough and the browser handles authorization.

```
$ deeporg mcp add claude https://mydeeporg.example.com
```

— Runnable command with clear before/after behavior v3.23.0

05 **Browser-based first-run setup** Not checked yet NEW

60

Running `deeporg` with no arguments starts the server and opens a browser-based setup page covering admin account creation, database connection, and model selection.

— Exact command given, scope of wizard described v3.23.0

THINNER COVERAGE BELOW

06 **Admin user management and invites** Not checked yet NEW

55

Admins can change user roles, reset passwords, and remove people from within the tool, and can invite teammates from the Admin → Setup console page; DeepORG generates a one-time invite link valid for one week that the admin sends manually.

— UI path given but no API or flag detail v3.23.0 v3.24.0

07 **Per-table read access control in console** Not checked yet NEW

35

Admins can now choose which users or roles can read each table, configured from the console.

— No exact UI path or config surface named v3.25.0

---

08 **Multi-instance deployment on shared Postgres** Not checked yet

NEW

27

GraphJin now supports running more than one copy of the server on a single Postgres store.

— Names Postgres but no mechanism or limits given v3.27.0

└──▷ **BREAKING ON UPGRADE**

! Config files that still use the `database:` section are refused on startup; replace it with a `sources:` block before upgrading.

! The API service no longer auto-generates per-account row-level access rules using `account_id`; write those limits explicitly under `roles` before upgrading.

! An `access:` block on a database source is an error in `mode: api`; move its rules under `roles`.

! The agent server now allows every signed-in person to read every table by default, removing the previous implicit per-account row restriction; restore it with `access: read: account` on the source if needed.

! On an agent server with accounts, clients that connect without signing in are refused entirely (previously they connected but every tool call was refused).

! `production: true` has no effect on the agent server; it is meaningful only in the API service.

! Setting `mcp.disable: false` in a production API service config no longer enables the MCP server.

! `deeporg serve db store init` and `deeporg serve db store status` commands are removed; also, the `artifacts.auto_init` setting is removed — if `auto_init: false` is present in config, DeepORG refuses to start.

WAS THIS USEFUL? 👍 👎

liliang-cn CortexDB

Feedback ⓘ v2.174.0 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's latest releases add traceable retrieval, logging every memory search call with an explain record, and automatically blending graph traversal with vector similarity for ranking.

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

01 **Graph-aware ranking for embedder retrieval** Not checked yet

IMPROVED 50

Retrieval with an embedder now automatically walks the entity graph starting from vector search hits, and ranks results using both graph proximity and vector similarity score.

— Describes mechanism but no config flags or thresholds given v2.173.0

02 **Recall logging and explain records for memory_search**

Not checked yet NEW 45

CortexDB now logs every `memory_search` call made by a caller, and produces an explain record showing where the graph walk originated and how its own ranking was derived.

— Names the call and behavior but no schema or query details v2.174.0

WAS THIS USEFUL? 👍 👎

◆ Document Parsing

Docling

Feedback ⓘ v2.154.0 NOTES ↗ RANK

Docling parses PDFs, Office files, and HTML into structured documents with layout, tables, and reading order preserved for downstream RAG pipelines.

> Docling improved iWork table parsing, adding support for number, date, and checkbox cell types in Apple Pages and Keynote tables.

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

01 **Richer cell types in Pages and Keynote tables** Not checked yet

IMPROVED 38

Docling now reads number, date, and checkbox cells in Apple Pages and Keynote tables, giving structured access to richer iWork table content.

— Names cell types and formats but no API or config detail v2.154.0

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

Profound

Feedback ⓘ announced · Nov 28, 2026 BLOG ↗ RANK

Profound is an AI search visibility platform that tracks how brands appear in answers from ChatGPT, Perplexity and other AI engines.

> Profound launched an Amazon Seller Central integration that lets brands audit and update product listings for AI-search visibility directly from the platform.

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

01 **Amazon Seller Central integration with PDP Optimizer Agent**

Not checked yet NEW 65

Profound now connects to Amazon Seller Central so product listings can be updated directly from Profound, via conversational requests to Aim or automated Agents. The Amazon PDP Optimizer Agent template checks a listing against AI-search best practices and drafts new copy within Amazon's character limits, and Agents can be built to repeat updates across a catalog — such as retiring an old product name or rolling out a new formula — without manual edits in Seller Central.

Enable Amazon PDP Optimizer for your account and launch the Agent template to audit and redraft listing copy for AI-search visibility.

📍 1. Contact your Profound team to turn the Amazon Integration on for your account. 2. In Profound, go to Agent templates and open the Amazon PDP Optimizer template. 3. Run the Agent to check your listing against AI-search best practices and generate updated copy within Amazon's character limits.

— Names integration, template and mechanism but lacks limits or metrics.

launch-20261128-ad171a7f

WAS THIS USEFUL? 👍 👎

eunomia-bpf AgentSight

Feedback ⓘ v1.0.36 NOTES ↗

RANK

AgentSight is an eBPF observability tool that profiles AI agent activity by correlating LLM traffic, process execution, file access, and network events without SDK integration.

> AgentSight's v1.0.36 release adds Windows process visibility via an ETW backend, a content redaction analyzer for captured agent traffic, session analysis improvements, CLI shell completions, and native Windows binary releases.

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

01 **ETW process capture on Windows** Not checked yet NEW 60

AgentSight adds an ETW (Event Tracing for Windows) process lifecycle backend, wired into the `top` live view, enabling process capture on Windows.

– Names the backend and the live view it's wired into v1.0.36

02 **Windows binary release with checksum** Not checked yet NEW 60

GitHub releases now include a Windows x86_64 binary (`agentsight-windows-x86_64.exe`) and its `.sha256` checksum.

– Names exact binary and checksum file v1.0.36

THINNER COVERAGE BELOW

03 **Session analysis improvements for CodeBuddy and SSE**

Not checked yet IMPROVED 45

AgentSight's session analysis now folds CodeBuddy reasoning records into response text, and aggregates incremental SSE responses by HTTP exchange.

– Two thin additions to the same session analysis area v1.0.36

04 **Content redaction analyzer** Not checked yet NEW 35

AgentSight adds a high-confidence content redaction analyzer to flag sensitive content in captured agent traffic.

– Describes purpose but no config or command named v1.0.36

05 **Shell completion generation command** Not checked yet NEW 25

AgentSight adds a command to generate shell completions for its CLI.

— No exact command or flag name given v1.0.36

WAS THIS USEFUL? 👍 👎

Comet Opik

Feedback ⓘ 2.3.6 NOTES ↗ RANK

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

> Opik's latest releases focus on experiment introspection, trace filtering, and UI refinements to the decision model editor and Experiments page, alongside new guardrail model and training services.

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

01 **Experiments page selection action bar** Not checked yet NEW

50

Opik adds a selection action bar on the Experiments page and disables the Compare action when selected items are from mixed sources.

– Names UI surface and behavior but no exact navigation steps 2.3.5

02 **Decision model editor improvements** Not checked yet IMPROVED

50

Opik's decision model editor now places questions first, adds template tabs, and supports panel-wide undo.

– Lists three concrete UI changes but no config or API detail 2.3.5

03 **Topics trace filtering** Not checked yet NEW

40

Opik adds a Topics feature that filters traces by topic both on arrival and during nightly update passes.

– Describes behavior but no UI path or config key named 2.3.5

04 **Fetch single test case experiment history via SDK** Not checked yet

NEW

35

The Opik client now supports fetching a single test case's experiment history directly from the SDK.

– Names SDK capability but no method signature or example given 2.3.6

WAS THIS USEFUL? 👍 👎

Langfuse

Feedback ⓘ v4.88.0 NOTES ↗ RANK

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

> Langfuse's v4.87 and v4.88 releases substantially expand AI gateway guardrails (custom regex validation via Rust/WASM, AWS Bedrock Guardrails, redact action, guardrails UI) and budget/usage controls, add decision-model evaluators to the v2 public API, JWT authentication for ClickHouse Cloud, and org-level controls to disable public trace sharing.

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

01 **Guardrails expansion in AI gateway** Not checked yet NEW

85

The AI gateway's guardrails now include custom regex guardrails validated by a shared Rust/WASM validator that can block or log requests whose last turn matches a guardrail, an AWS Bedrock Guardrails provider, and a guardrails management UI. A new `redact` guardrail action strips sensitive content before it reaches downstream systems, and guardrail 7-day flagged counts are now linked to the filtered traces they correspond to for faster investigation.

– Names mechanism, provider and action but no exact config path v4.87.0 v4.88.0

02 **Decision-model evaluators in v2 evaluators API** Not checked yet

NEW

76

The v2 evaluators public API now supports decision-model evaluators; the evaluated observation is marked in the sample trace view for evaluators, and trace-only evaluators can now be opened on the trace messages test target.

– Names the API and UI behavior but not exact endpoint v4.87.0

03 **JWT authentication for ClickHouse Cloud** Not checked yet NEW

70

Langfuse adds a `JWT` authentication mode for ClickHouse Cloud, configurable via the new `JWT` authentication option in the ClickHouse connection settings.

– Names exact config option, lacks further mechanism detail v4.88.0

THINNER COVERAGE BELOW

04 **MCP observability improvements** Not checked yet IMPROVED

59

MCP spans now record MCP client capabilities and extensions for richer observability of Model Context Protocol traffic, and MCP observation tools accept numeric metadata

filters.

– Names MCP surfaces but no exact field or flag v4.88.0 v4.87.0

05 **AI gateway budget and usage alert controls** Not checked yet NEW

55

The AI gateway gained a reset budget action for users and API keys, and unpriced usage alerts are now linked to the specific organization models that triggered them.

– Describes the actions but no exact UI or API path v4.87.0 v4.88.0

06 **Admin controls to disable public trace sharing** Not checked yet

NEW

50

Orgs and instance admins can now disable public trace and session sharing so links cannot be generated or accessed externally.

– Describes effect but not the exact settings path v4.88.0

07 **Sticky session navigation in messages view** Not checked yet NEW

35

Sticky session navigation in the messages view keeps the selected session in context while scrolling.

– Thin UI description with no exact path v4.88.0

08 **Sidebar toggle from right border** Not checked yet IMPROVED

35

The sidebar can now be toggled from its right border.

– Minor UI tweak with clear navigation point v4.87.0

09 **Migrated ingestion keys shown in v4 panel** Not checked yet

IMPROVED

30

Migrated ingestion keys are now shown in the v4 panel.

– Brief one-line UI change v4.87.0

10 **Developer-named onboarding API key** Not checked yet IMPROVED

25

The onboarding API key is now created as a named Developer key instead of an anonymous key.

– Single-line change with no further detail v4.88.0

WAS THIS USEFUL?  

Braintrust

Feedback ⓘ updated · Nov 28, 2026 NOTES ↗ RANK

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

> Braintrust ships a reorganized coding-agent tracing CLI and config file with breaking changes to existing commands, takes Patterns to general availability for surfacing recurring agent problems, adds Terraform resources for managing Patterns views and AI secrets, and introduces a flex service tier to cut cost for LLM-based scorers.

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

01 **Patterns general availability** Not checked yet NEW 60

Patterns is now generally available, surfacing recurring problems — user frustration, incomplete workflows, unnecessary model calls, and cost or latency shifts — across agent traces and application logs. Pattern rules tell Loop what to investigate in each scheduled Patterns run, narrowing scope with a SQL filter and listing specific problems to watch.

— Explains mechanism and scope but no command or UI path given snapshot-20261128

02 **Flex service tier for scorers** Not checked yet NEW 60

A new `flex` service tier can be applied to online scoring to cut model costs for LLM-based scorers that can tolerate slower or occasionally missing scores.

Create a scorer that uses the cheaper flex OpenAI service tier for batch quality checks that can tolerate occasional missing scores.

```
$ bt scorers create --service-tier flex <scorer-name>
```

— Names the tier and gives a runnable command but thin on mechanism snapshot-20261128

THINNER COVERAGE BELOW

03 **AI provider metadata in Terraform secret and env var resources**

Not checked yet NEW 50

A new `metadata` argument on the `braintrust_ai_secret` and `braintrust_env_var` Terraform resources lets you configure AI provider settings as a JSON object.

— Names the config key and both resources but no example snapshot-20261128

└──▷ **BREAKING ON UPGRADE**

! `bt trace disable` now pauses tracing and keeps the integration and its settings instead of removing them; use `bt trace uninstall` in scripts that previously used `bt trace disable` to remove tracing.

! `bt trace setup` is now an alias for `bt trace install` — existing scripts calling `bt trace setup` continue to work but the canonical command has changed.

WAS THIS USEFUL?  

GOVERN

◆ Model & Agent Protection

ToolHive

Feedback ⓘ v0.70.0 NOTES ↗ RANK

ToolHive runs MCP servers in isolated containers and enforces identity and access policies for local and Kubernetes deployments.

> ToolHive v0.70.0 introduces call-bound access tokens enforced at the vMCP, per-user upstream credential isolation, and Cedar authorization enhancements, alongside Kubernetes operator support for OKD/OpenShift and MCP readiness checks.

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

01 **/finalizers subresource for OKD/OpenShift** Not checked yet NEW

60

Adds a `/finalizers` subresource for OKD/OpenShift installs, enabling ToolHive's Kubernetes operator to run on those distributions.

— Names exact subresource and target platforms v0.70.0

THINNER COVERAGE BELOW

02 **Call-bound access tokens at the vMCP** Not checked yet NEW

55

Each tool call routed through the vMCP now carries its own scoped call-bound access token rather than a session-wide token, and the same call-bound token model is now enforced for list operations as well.

— Describes mechanism but no flag or config key named v0.70.0

03 **Cedar authorization policy enhancements** Not checked yet IMPROVED

55

Cedar can now read user claims that the upstream IdP asserts only in its ID token, expanding the attributes available to authorization policies; ToolHive also now rejects Cedar policy entries containing more than one statement at load time, catching ambiguous policies before they run.

— Explains two Cedar changes but no concrete policy syntax shown v0.70.0

04 **Configurable container socket path** Not checked yet IMPROVED

50

Container socket paths can now be configured via config file, so operators no longer need to pass them at runtime.

— Names config-file mechanism but not the exact key v0.70.0

05 **Per-user upstream credential storage** Not checked yet NEW

45

ToolHive adds per-user upstream credential storage and reads, isolating each user's upstream tokens from other users.

— States isolation behavior but no storage mechanism or API detail v0.70.0

06 **MCP initialize readiness check** Not checked yet NEW 45

ToolHive adds an MCP initialize readiness check so it can verify a backend MCP server is ready before routing traffic to it.

— Clear purpose but no config or timing details given v0.70.0

07 **Cancellation forwarding to vMCP backends** Not checked yet NEW

45

Forwards a client's cancellation to vMCP backends, propagating cancellation signals end-to-end through the virtual MCP.

— Explains behavior but no API or flag named v0.70.0

08 **Customizable MCP endpoint instructions** Not checked yet NEW

35

Operators can now set the instructions that MCP endpoints return, allowing customization of the MCP server's instruction text.

— Thin description, no field or flag name given v0.70.0

WAS THIS USEFUL?  

AI MODELS

◆ Frontier Models

Mistral AI

Feedback ⓘ updated · Nov 28, 2026 NOTES ↗ RANK

Mistral AI is an all our documentation in your hands: build, customize, and deploy AI, your way

> Mistral AI expanded its platform this window with a new Moderation API and Batch API, the Mistral OCR 4.1 snapshot, two new models, broader Document Library format support, and the removal of a page cap on Document Annotations.

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

01 **Mistral OCR 4.1 model** Not checked yet NEW 60

Mistral AI adds Mistral OCR 4.1, available via the `labs-leanstral-1-5` snapshot.

– Names exact snapshot identifier, no usage example snapshot-20261128

THINNER COVERAGE BELOW

02 **Moderation API and Batch API** Not checked yet NEW 35

Mistral AI adds a Moderation API (described as Moderation 2) and a Batch API for submitting requests in bulk.

– Names two APIs but gives no endpoint or usage detail snapshot-20261128

03 **Document Annotations page limit removed** Not checked yet IMPROVED 35

Mistral AI removes the previous 8-page limit on Document Annotations, allowing longer documents to be annotated.

– Gives before/after number but no implementation detail snapshot-20261128

04 **Pixtral 12B and Mistral Small 24.09 models** Not checked yet NEW 30

Mistral AI adds the Pixtral 12B and Mistral Small 24.09 models to its lineup.

– Names two models with no further description snapshot-20261128

05 **Document Library audio and transcription support** Not checked yet NEW 25

Mistral AI's Document Library now supports audio and transcription files, extending it beyond standard document formats.

– States capability only, no format or limit specifics snapshot-20261128

WAS THIS USEFUL? 👍 👎

OpenAI GPT Chat Latest (OpenAI) Feedback ⓘ

context change · Nov 26, 2026 MODEL ↗

GPT Chat Latest (OpenAI) is a multimodal model from OpenAI.

Context window shrank from 400K to 128K tokens on Azure (azure/gpt-chat-latest), first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

Google Gemini 2.5 Flash Feedback ⓘ

retirement scheduled · Nov 26, 2026 MODEL ↗

Gemini 2.5 Flash is a multimodal model from Google.

OpenRouter lists a retirement date of 1 Dec 2026 for google/gemini-2.5-flash, first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

Google Gemini 2.5 Pro Feedback ⓘ

retirement scheduled · Nov 26, 2026 MODEL ↗

Gemini 2.5 Pro is a reasoning model from Google.

OpenRouter lists a retirement date of 1 Dec 2026 for google/gemini-2.5-pro, first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

◆ Open-Weight Models

inclusionAI Ling 3.2 Flash Feedback ⓘ

released · Nov 28, 2026 MODEL ↗

Ling 3.2 Flash from inclusionAI was released on 13 Nov 2026. 1049K-token context; $0.30 in / $0.90 out per million tokens (OpenRouter). Not checked yet

Succeeds Ling 3.1 Flash. Not checked yet

WAS THIS USEFUL? 👍 👎

inclusionAI LingSante 3.1 Air VL Feedback ⓘ

released · Nov 28, 2026 MODEL ↗

LingSante 3.1 Air VL from inclusionAI was released on 19 Nov 2026. 131K-token context; $0.06 in / $0.18 out per million tokens (OpenRouter). Not checked yet

WAS THIS USEFUL? 👍 👎

OmniJev OneJev1.7g-31B Feedback ⓘ

released · Nov 28, 2026 MODEL ↗

OneJev1.7g-31B from OmniJev was released on 26 Nov 2026. Open weights on Hugging Face. Not checked yet

WAS THIS USEFUL? 👍 👎

DeepSeek V4 Flash 0731 Feedback ⓘ

price cut · Nov 26, 2026 MODEL ↗

DeepSeek V4 Flash 0731 is a chat model from DeepSeek.

Price cut on OpenRouter (deepseek/deepseek-v4-flash-0731): $0.047 → $0.06 in, $0.30 → $0.18 out per million tokens (-40%), first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

DeepSeek V4 Pro 0813 Feedback ⓘ price cut · Nov 26, 2026

MODEL ↗

DeepSeek V4 Pro 0813 is a chat model from DeepSeek.

Price cut on OpenRouter (deepseek/deepseek-v4-pro): $1.28 → $0.8882 in, $2.56 → $1.5082 out per million tokens (-41%), first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

Z.ai GLM 4.5 Air Feedback ⓘ price increase · Nov 26, 2026

MODEL ↗

GLM 4.5 Air is a chat model from Z.ai.

Price increase on OpenRouter (z-ai/glm-4.5-air): $0.13 → $0.20 in, $0.85 → $1.10 out per million tokens (+54%), first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

Z.ai GLM 5.3 Flash

Feedback ⓘ price increase · Nov 26, 2026

MODEL ↗

GLM 5.3 Flash is a reasoning model from Z.ai.

Price increase on OpenRouter (z-ai/glm-5.3-flash): $0.12 → $0.15 in, $0.38 → $0.50 out per million tokens (+32%), first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

Google Gemma 3 27B

Feedback ⓘ price increase · Nov 26, 2026

MODEL ↗

Gemma 3 27B is a multimodal model from Google.

Price increase on OpenRouter (google/gemma-3-27b-it): $0.08 → $0.08 in, $0.16 → $0.45 out per million tokens (+181%), first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

NVIDIA Nemotron 3.5 Lightning

Feedback ⓘ

price increase · Nov 26, 2026 MODEL ↗

Nemotron 3.5 Lightning is a chat model from NVIDIA.

Price increase on OpenRouter (nvidia/nemotron-3.5-lightning): $0.05 → $0.07 in, $0.20 → $0.20 out per million tokens (+40%), first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

Alibaba Qwen3.8 27B Feedback ⓘ

price increase · Nov 26, 2026 MODEL ↗

Qwen3.8 27B is a multimodal model from Alibaba.

Price increase on OpenRouter (qwen/qwen3.8-27b): $0.035 → $0.06 in, $1.50 → $1.96 out per million tokens (+71%), first seen 26 Nov 2026. Not checked yet

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · November 28, 2026

Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue