



Head

THE WEEK'S TOP PICKS

PUBLISHED OCTOBER 23, 2026 · FRIDAYS

// SHARE THIS WEEK

▼ Top 5

▼ Top Pick

EDITIONS tail | grep | head | ls | diff | uniq | perf

The best tooling updates that shipped this week.

► // HOW THESE PICKS ARE MADE

SHOW

▼ // BENCHMARKS MOVED FOR TOOLS WE TRACK

HIDE

Anthropic ARC-AGI-2 Claude Haiku 5.5 low

2.8% -90.5 pts released 2026-10-07 · [leaderboard ↗](#)

Anthropic ARC-AGI-2 Claude Haiku 5.5 high

30.7% -62.6 pts released 2026-10-07 · [leaderboard ↗](#)

Anthropic ARC-AGI-2 Claude Haiku 5.5 medium

9.4% -83.9 pts released 2026-10-07 · [leaderboard ↗](#)

Gemini API ARC-AGI-2 Gemini 4 Argon high

91.2% +2.0 pts released 2026-09-30 · [leaderboard ↗](#)

Gemini API ARC-AGI-2 Gemini 4 Argon medium

87.1% -2.1 pts released 2026-09-30 · [leaderboard ↗](#)

Gemini API ARC-AGI-2 Gemini 4 Argon low

42.1% -47.1 pts released 2026-09-30 · [leaderboard ↗](#)

Anthropic ARC-AGI-2 Claude Haiku 5.5 xhigh

61.3% -32.0 pts released 2026-10-07 · [leaderboard ↗](#)

vim ? help

Week Models

SimpleQA Verified DeepSeek V4.1 Flash 1017

max

42.5% -0.4 pts 2026-10-19 · [Epoch run ↗](#)

DeepSeek Models FrontierMath (tiers 1-3) DeepSeek V4.1 Flash 1017 max

71.2% +3.8 pts 2026-10-19 · [Epoch run ↗](#)

DeepSeek Models GPQA Diamond DeepSeek V4.1 Flash 1017 max

93.2% +3.4 pts 2026-10-19 · [Epoch run ↗](#)

Scores from [Epoch AI Benchmarking Hub](#) (CC BY 4.0), evaluated 2026-10-16 to 2026-10-23.

▼ // **THE BRIEF** HIDE

What stands out in this week's picks, grouped by what it lets you do. Every tool named links to its pick below.

Three unrelated things stand out. Coding agents stopped running with your whole machine in reach — Kiro enforces a sandbox at the OS level, Agno puts ClaudeAgent and CodexAgent in per-session Docker. Mistral refreshed its frontier lineup and added Custom Guardrails you define yourself. And Pinecone BYOC drops shard pre-sizing for an on-demand pool.

BUILD

Let a coding agent run shell commands without betting your filesystem on its judgment

Kiro enforces local sandboxing at the OS layer rather than through prompt instructions, and Agno runs ClaudeAgent and CodexAgent inside isolated per-session Docker or AgentCore computers with Postgres-backed checkpointing. The usual containment work — hand-rolling a devcontainer, maintaining an allowlist of safe commands — stops being the price of letting an agent act.

[Kiro](#) · [Agno](#)

GOVERN

Write your own refusal rules instead of taking the model vendor's defaults

Mistral's Custom Guardrails apply to Agents and Conversations, and Anthropic adds per-customer safety enforcement through a user profiles API with admin webhooks plus org-level web fetch and search domain controls for Claude Managed Agents. For anyone running a multi-tenant

product, policy becomes per-account configuration rather than one global system prompt everyone shares.

[Mistral AI](#) · [Anthropic](#)

DEPLOY

Stop guessing shard counts before you know your query load

BYOC gains an on-demand capacity mode backed by a shared pool that rebalances dynamically, with [Prometheus](#)-based monitoring. Pre-sizing a [vector index](#) means either paying for headroom you never use or re-sharding under load; neither decision has to be made up front now.

[Pinecone](#)

BUILD

Get a tighter retrieval step and a pointer to the span that justified it

Rerank 5 ships two model variants plus Context Focus [span extraction](#) and [calibrated relevance scores](#). Calibrated scores mean a cutoff threshold transfers across queries instead of being re-tuned per corpus, and the extracted span gives a [RAG pipeline](#) something to cite rather than passing a whole chunk to the generator.

[Cohere Models](#)

VIEW

ISSUE VIEW

scroll

SCROLL

EXPAND IN PLACE

PREVIEW PANE

INFORMATION DENSITY

everything

BRIEFED

EVERYTHING

Every change in this week's picks, in full.

all 74 changes

Do you prefer scroll?

// THE PICKS

01 vLLM

Inference Serving & Optimization

02 **Mistral AI**

Frontier Models

03 **Kiro**

AI Coding Agents

04 **SAM**

Agentic Infrastructure

05 **Anthropic**

Frontier Models

01 vLLM

announced · Oct 20, 2026

Inference Serving & Optimization

A high-throughput and memory-efficient inference and serving engine for LLMs

// WHY IT MADE THE LIST

DEPTH 5/5

IMPACT 5/5

NOVELTY 4/5

Launches a TorchTPU backend plugin that serves vLLM models on Google Cloud TPUs across all six parallelism dimensions with no model-code changes, plus day-0 support for Flint 40B and Reflection Beam 501B with speculative decoding — opening both a hardware class and new model classes to practitioners immediately.

vLLM launched a new TorchTPU backend plugin for serving on Google Cloud TPUs, shipped v0.32.0 with a `/v1/systemone` endpoint, FlashInfer kernel tooling, multi-worker API scaling, generic `hf` parsers, and several breaking changes (Python 3.11+, restricted CORS defaults), and added a push-based NIXL KV connector plus an Adaptive Verification extension for headless draft models.

→ WHAT SHIPPED · 17 FEATURES most completely described first ? what's the number?

01 TorchTPU backend plugin for Google Cloud TPUs

NEW SUPPORTED PLATFORM

NEW

95

Adds `vllm-torchtpu` (`pip install vllm-torchtpu`), a new out-of-tree vLLM platform plugin that serves vLLM models on Google Cloud TPUs via the TorchTPU stack with no model code changes, supporting `--tensor-parallel-size`, `--data-parallel-size`, `--pipeline-parallel-size`, `--enable-expert-parallel`, `--prefill-context-parallel-size`, `--decode-context-parallel-size`, and `--async-scheduling` flags for single- and multi-host TPU slices, the `VLLM_XLA_CACHE_PATH` environment variable to persist compiled XLA graphs across restarts, and `--kv-cache-dtype fp8` for KV cache quantization; TPU v7x (Ironwood) is validated in CI on every pull request, with v6e, v5e and v5p available as experimental targets. It ships custom Pallas kernels for Ragged Paged Attention v3 (MHA/GQA), dense and sparse Multi-head Latent Attention (MLA), Gated DeltaNet, fused Mixture-of-Experts with expert parallelism, and FP8/INT8/MXFP4/NVFP4 quantized matmuls; enables zero-copy KV cache offloading to host DRAM and disaggregated prefill/decode serving across TPU hosts via TPU Sync and vLLM's KV connector interface; and supports speculative decoding (MTP, EAGLE-3, DFlash, DSpark, n-gram, suffix decoding) and automatic prefix caching, including linear-attention state caching for hybrid models like Qwen3.5 and Kimi K3, via existing vLLM flags.

Serve a large MoE model on a TPU v7x host with data-parallel attention and expert parallelism to maximise throughput, caching compiled graphs so subsequent restarts

skip recompilation.

```
$ VLLM_XLA_CACHE_PATH=~/.cache/vllm_xla vllm serve Qwen/Qwen3-Coder-30B-A3B-Instruct-FP8 \
  --tensor-parallel-size 1 \
  --data-parallel-size 8 \
  --enable-expert-parallel \
  --kv-cache-dtype fp8 \
  --max-model-len 16384 \
  --async-scheduling
```

Serve a long-context model on TPU with prefill and decode context parallelism to split the KV cache across devices for sessions that exceed per-device HBM.

```
$ VLLM_XLA_CACHE_PATH=~/.cache/vllm_xla vllm serve meta-llama/Llama-3-70B-Instruct \
  --tensor-parallel-size 4 \
  --prefill-context-parallel-size 4 \
  --decode-context-parallel-size 4 \
  --max-model-len 65536 \
  --async-scheduling
```

– Names every flag and env var with runnable serve commands Names kernel types but no flag or command to invoke them Describes mechanism but no configuration surface given Names supported methods but relies on unnamed existing flags

launch-20261020-af0bc9d5

02 Push-based NIXL KV connector for disaggregated serving NEW

95

Adds `NixlPushConnector`, a push-based alternative to `NixlConnector` for KV cache transfer in disaggregated prefill/decode deployments, overlapping decode-side block registration with prefill computation to remove post-prefill coordination from the TTFT critical path. It moves NIXL descriptor preparation and transfer submission off the decode worker's engine loop onto a dedicated background transfer thread on each prefill worker, supports direct GPU/Trainium memory transfers over RDMA interconnects (including AWS EFA with GPUDirect RDMA and Neuron Direct RDMA) without staging through host memory, and allows prefill and decode instances to use different internal block layouts and block sizes. Disabling prefix caching on the decode node with `--no-enable-prefix-caching` forces full KV cache retrieval from the prefill node for every request, enabling clean isolation of P→D transfer performance in benchmarks; this cuts TTFT by up to 3x and TPOT by up to 30% in disaggregated inference.

Benchmark pure P→D transfer performance by disabling prefix caching on the decode node so every request retrieves the full KV cache from the prefill node.

```
$ vllm serve <model> --no-enable-prefix-caching
```

– Named class, flag, mechanism, hardware support and benchmark numbers

launch-20261014-98cafc78

03 Multi-worker API server load balancing NEW

85

Adds `--api-server-count` flag: when set above 1, each connection is routed to the least-loaded API worker, yielding 43-85% more req/s on short requests.

Serve a model using the generic `hf` tool-call and reasoning parsers so no model-specific parser is needed, and distribute incoming connections across 4 API workers for higher short-request throughput.

```
$ vllm serve meta-llama/Llama-3.1-8B-Instruct --tool-call-parser hf --reasoning-parser hf --api-server-count 4
```

– Named flag with quantified benefit and runnable command

v0.32.0

04 FlashInfer kernel caching and autotune warm-up NEW

80

Adds `vllm download-kernels` to install FlashInfer precompiled kernels for faster startup, and `vllm preload --preload-autotune` to keep the FlashInfer autotune table warm across restarts.

Pre-install FlashInfer precompiled kernels after upgrading vLLM to eliminate JIT compilation overhead at server startup.

```
$ vllm download-kernels
```

– Two runnable subcommands with a working example

v0.32.0

05 systemone endpoint for structured reasoning questions NEW

75

Adds `POST /v1/systemone` (enabled by default), answering `choice`, `noul`, and `score` questions over a structured state, with support for `Cloudflare/clef-omni`, `LiquidAI/d1-3B`, and DiffusionGemma.

– Named endpoint and models, but no usage example given

v0.32.0

06 Generic hf tool-call and reasoning parsers NEW

75

Adds `--tool-call-parser hf` and `--reasoning-parser hf`, which read the checkpoint's `response_template` so no model-specific parser is needed for models that ship one.

Serve a model using the generic `hf` tool-call and reasoning parsers so no model-specific parser is needed, and distribute incoming connections across 4 API workers for higher short-request throughput.

```
$ vllm serve meta-llama/Llama-3.1-8B-Instruct --tool-call-parser hf --reasoning-parser hf --api-server-count 4
```

– Named flags and mechanism with a runnable serve command `v0.32.0`

07 Adaptive Verification extended to headless draft models

IMPROVED

75

Enables `enable_adaptive_verification` for draft models that lack a confidence head (EAGLE, MTP, DFlash, and headless DSpark checkpoints), using an online-fitted logistic regression on `logit(q_max)` to estimate per-token acceptance probability. The online estimator uses one shared slope and one per-position intercept, continuously updated via IRLS from live accept/reject decisions, so any existing draft checkpoint gains Adaptive Verification without retraining.

– Named config key and mechanism, no usage example given `launch-20261014-03851032`

08 Sleep mode parameter retention and NCCL suspend controls NEW

65

Adds `--sleep-preserve-parameter-names` (e.g. `'draft.*'`) to selectively retain named parameters during sleep, and `--enable-nccl-comm-suspend` to release the MNNVL workspace during sleep.

– Two named flags but no example invocation shown `v0.32.0`

09 Decoder-replay trim threshold for CUDA graphs NEW

65

Adds `compilation_config.decoder_replay_trim_threshold` config key to trim decoder-replay layers in PIECEWISE CUDA graphs for DeepSeek V4/V4.1.

– Named config key and scope, no example value shown `v0.32.0`

THINNER COVERAGE BELOW

10 Shared-memory tensor arena for multimodal tensor broadcast

IMPROVED

55

Adds `--enable-shm-tensor-arena` (on by default), enabling a shared-memory tensor arena that broadcasts large multimodal tensors to workers with a single copy.

– Named flag and default, no example of effect measured v0.32.0

11 Minimum Python and dependency versions raised

BREAKING

55

Python 3.10 support is dropped (Python 3.11+ now required);

`transformers>=5.16.1` and `tokenizers>=0.23.2` are now required minimum versions.

– Exact version floors named, straightforward migration note v0.32.0

12 Weight-transfer validation endpoint for RL

NEW

50

Adds `POST /weight_checker` endpoint for RL weight-transfer validation.

– Named endpoint but no usage detail beyond its purpose v0.32.0

13 max_tokens no longer reserved from context window

BREAKING

35

`max_tokens` is no longer reserved from the context window.

– Behavior change named but no before/after numbers given v0.32.0

14 Default CORS origins restricted

BREAKING

30

Default CORS origins are now restricted – the previous default of allowing every origin is removed.

– States the change but no new default values given v0.32.0

15 mamba_cache_mode='all' removed

BREAKING

25

`mamba_cache_mode='all'` is removed.

– Named config value removed, no alternative stated v0.32.0

16 Cascade attention removed

BREAKING

20

Cascade attention is removed from vLLM.

– Bare removal notice with no replacement or migration path v0.32.0

17 Responses API reasoning stream events renamed

BREAKING

20

Responses API reasoning stream events are renamed.

– No old or new event names given v0.32.0

NEW SUPPORTED PLATFORM

[Recursive Flint \(40B MoE\) model support](#)

NEW SUPPORTED PLATFORM

[Humming kernel library: NVIDIA GPU support from Turing onward](#)

NEW SUPPORTED PLATFORM

[Humming kernel library: T-Head Zhenwu 810E support](#)

NEW INTEGRATION

[Reflection Beam 501B MoE model support](#)

NEW SUPPORTED PLATFORM

[NVIDIA Blackwell \(GB200/GB300\) platform support for Beam](#)

NEW INTEGRATION

[SkyRL RL fine-tuning integration for Beam at 500B scale](#)

NEW INTEGRATION

[Muse Spark Open LoRA RL rollout with Verl FSDP2](#)

⚠️ BREAKING ON UPGRADE

- ! Python 3.10 support is dropped; Python 3.11+ is now required.
- ! Default CORS origins are now restricted — the previous default of allowing every origin is removed.
- ! `transformers>=5.16.1` and `tokenizers>=0.23.2` are now required minimum versions.
- ! Cascade attention is removed.
- ! `mamba_cache_mode='all'` is removed.
- ! `max_tokens` is no longer reserved from the context window.
- ! Responses API reasoning stream events are renamed.

announced · Oct 21, 2026

vLLM gains day-0 Flint support via `vllm/vllm-openai:flint` image and `vllm serve`, with new `--hf-config-path`, `--tool-call-parser glm47`, `--reasoning-parser glm47`, and `--exclude-tools-when-tool-choice-none` flags, plus Flint-specific Blackwell GPU kernels for faster decode.

Adds `vllm/vllm-openai:flint` Docker image (x86_64 and arm64, CUDA 13.0) for serving Recursive's Flint model with no extra setup.

NEW SUPPORTED PLATFORM

Supports serving Flint's NVFP4 GGUF checkpoint via `vllm serve RecursiveSI/flint-nvfp4-gguf` with `--hf-config-path` and `--tokenizer` pointing to `RecursiveSI/flint`.

Supports serving Flint's BF16 checkpoint via `vllm serve RecursiveSI/flint` with GLM-4.7-style tool-call and reasoning parsers.

Adds `--tool-call-parser glm47` and `--reasoning-parser glm47` to support Flint's GLM-4.7-style tool calls and thinking blocks.

Adds `--exclude-tools-when-tool-choice-none` flag to drop tool definitions from the prompt when `tool_choice` is `none`.

+2 more

announced · Oct 16, 2026

vLLM adds day-0 serving of Reflection Beam 501B MoE with DFlash2 speculative decoding, adaptive verification, Attention DP + MegaMoE, and RL fine-tuning support.

NEW INTEGRATION Adds support for Reflection Beam 1.0 501B-A23B via Docker image `vllm/vllm-openai:reflection-beam`, with `--reasoning-parser beam` and `--tool-call-parser beam` flags for reasoning and tool-calling out of the box.

Adds `--speculative-config` with `'method': 'dflash'` to enable DFlash2 speculative decoding for Beam, reaching 708 tokens/s on TP8 with an acceptance length of 3.7.

Adds `'enable_adaptive_verification': true` inside `--speculative-config` to dynamically decide per step how many draft tokens to verify, yielding +10.8% output throughput at concurrency 128.

Adds `--moe-backend flashinfer_moe_ep_cuteds1` with `--data-parallel-size` and `--enable-expert-parallel` flags to enable Attention DP + MegaMoE, delivering 1.71× higher peak prefill throughput over TP4.

Adds `VLLM_BEAM_ALWAYS_CONSTRAIN_OUTPUT=0` environment variable to disable the default structured-output constraint that enforces Beam's response template and tool-call JSON schema.

+1 more

NEW SUPPORTED PLATFORM NVIDIA Blackwell (GB200/GB300) platform support for Beam

announced · Oct 21, 2026

Ant Group and the vLLM community open-source Humming, a JIT-compiled kernel library for quantized inference supporting dense and MoE workloads across NVIDIA GPU generations.

NEW SUPPORTED PLATFORM Introduces Humming (github.com/vllm-project/humming), a general-purpose, high-performance kernel library for quantized inference co-developed by Ant Group and the vLLM community.

Provides a unified quantization representation covering integer and floating-point weight/activation format combinations (e.g., WNA16, W8A8, W4A8, W4A4) for both dense and MoE workloads.

Compiles kernels via NVRTC or NVCC (JIT) from reusable CUDA components, caching compiled kernels for reuse and enabling parallel warmup compilation in approximately 2–4 seconds per layer.

Exposes three-level configuration — Layer Config, Compute Config, and Tuning Config — for tile shapes, pipeline stage counts, warp specialization, TMA, MMA instructions, CTA concurrency, and Stream-K scheduling.

Outperforms vLLM Marlin, FlashInfer, and DeepGEMM on representative workloads, with JIT compile time dropping from 23.35 s (CUTLASS/NVCC) to approximately 1.82 s (Humming/NVCC) in a single W4A16 kernel test on H200 with CUDA 13.4.

+1 more

announced · Oct 15, 2026

vLLM v0.31.0 adds day-0 serving of Nemotron 3.5 Super VL via `vllm serve` with `-reasoning-parser nemotron_v3`, MTP speculative decoding, NVFP4 support, and `video_url` chat input.

NEW SUPPORTED PLATFORM Serves `nvidia/NVIDIA-Nemotron-3.5-Super-VL-120B-A12B-NVFP4` via `vllm serve` with `--reasoning-parser nemotron_v3`, `--speculative-config.method mtp`, and `--speculative-config.num_speculative_tokens` for multi-token-prediction speculative decoding on a multimodal model.

Supports NVFP4 quantized checkpoint of Nemotron 3.5 Super VL at launch alongside the BF16 baseline, enabling lower-precision inference on compatible NVIDIA hardware.

Accepts `video_url` content-type in OpenAI-compatible chat completions messages, enabling video understanding through the existing `/v1/chat/completions` API.

announced · Oct 15, 2026

vLLM gains day-0 serving of Muse Spark Open (816B MoE, 1M context) with DSpark speculative decoding, native audio/video/image multimodal API, and a Rust frontend option.

NEW SUPPORTED PLATFORM Serves `meta-models/Muse-Spark-Open-816B-A42B-NVFP4` via `vllm serve` with `--tensor-parallel-size`, `--enable-expert-parallel`, `--max-model-len 1048576`, `--reasoning-parser muse`, `--tool-call-parser muse`, and `--enable-auto-tool-choice` flags.

Enables DSpark speculative decoding via `--speculative-config` with `method: dspark`, `model: Inferact/Muse-Spark-Open-DSpark`, `num_speculative_tokens: 7`, `draft_sample_method: probabilistic`, `rejection_sample_method: block`, and `enable_adaptive_verification: true`.

Adds `audio_url` content-part type to the `/v1/chat/completions` API, enabling native audio perception (24 kHz raw waveform, 80 ms per token) alongside existing

`image_url` and `video_url` parts.

Supports `reasoning_effort` field (e.g. `xhigh`) on `/v1/chat/completions` requests for Muse Spark Open.

Enables `VLLM_USE_RUST_FRONTEND=1` environment variable to opt into the Rust frontend, recommended for Muse Spark Open serving.

[+1 more](#)

GOOD PICK?



GOOD REASON?



02 Mistral AI

announced · Oct 20, 2026

Frontier Models

Mistral AI provides open-source and commercial language models for developers to build AI applications with flexible, efficient inference.

// WHY IT MADE THE LIST

DEPTH 5/5

IMPACT 5/5

NOVELTY 5/5

Voxtral 3 launches a managed real-time voice pipeline — streaming speech-to-text and text-to-speech with zero-shot voice cloning across 50+ languages through a voice-agent SDK — alongside the open-weight 1M-context Mistral Large 4 and Timestral, Mistral's first zero-shot time-series foundation models that run on CPU.

Mistral shipped a wave of new frontier models — Mistral Large 4, Medium 3.5, and Small 4 — alongside the Voxtral 3 audio family, a Voice Agents SDK, Custom Guardrails for Agents and Conversations, expanded OCR API output controls, and its first time-series foundation model family, Timestral, while tightening API validation and retiring several older model aliases.

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

01 OCR API output control parameters

IMPROVED

90

The OCR API gains `confidence_scores_granularity` ('page' , 'block' , or 'word' , each level adding finer-grained scores), `include_blocks` (returns a `blocks` array with paragraph-level bounding boxes and structural labels in reading order when `true`), `pages` (accepting comma-separated digit strings and ranges like '0,1,2' , '0-5' , or '0,2-4' in addition to a list of integers), `table_format` (`markdown` or `html`), and `extract_footer` / `extract_header` parameters plus `hyperlinks` output.

– Every parameter and accepted value enumerated; no endpoint path given.

snapshot-20261023

snapshot-20261020

02 Voxtral 3 audio family: transcription and TTS models

NEW

85

Releases the Voxtral 3 audio family in Public Preview, spanning `voxtral-mini-tts-3-0` , `voxtral-mini-transcribe-3-0` , `voxtral-mini-transcribe-realtime-3-0` , `voxtral-small-transcribe-3-0` (also launched as Voxtral Small Transcribe 3, Voxtral Mini Transcribe 3, and Voxtral Realtime Transcribe 3), and `voxtral-tts-2603` /Voxtral Mini TTS 3.0 (batch and realtime). The transcription models support 50+ languages with `diarization`, context biasing, and end-of-turn detection, with the realtime model delivering sub-500ms latency across 35+ languages; the TTS models support 35+ languages and 50+ dialects with `zero-shot voice cloning` and contextual `emotion` ,

`spacing`, and `emphasis` control plus a vocabulary library and phoneme hints, priced at \$13 per 1M characters. Context biasing lets developers supply key terms or free-form text to improve recall on domain vocabulary without `fine-tuning`, and the family is available via Mistral's EU-hosted API or for self-deployment on private infrastructure with customer control over audio retention.

– Enumerates all models, specs, and params; no sample call shown.

`snapshot-20261023`

`snapshot-20261020`

`launch-20261020-26d56ceb`

03 Timestral time-series foundation models NEW

85

Introduces Timestral, Mistral's first family of `time-series foundation models`, in four sizes — 5M, 28M, 123M, and 209M parameters. It delivers probabilistic forecasts across 21 quantiles with multivariate targets, past-only covariates, and known-future covariates as inputs, supports zero-shot forecasting out of the box on `CPU` without requiring a `GPU`, and supports fine-tuning and architectural extensions via Mistral Forge, including domain-specific covariates and control over training data mix. It is available for on-premises, `VPC`, and air-gapped deployment under a Modified MIT licence, with a GitHub repository for running it on your own data.

– Rich detail on sizes, outputs, deployment; has GitHub repo.

`launch-20261015-5fba9386`

04 Model deprecations and renames across the API DEPRECATED

80

`mistral-tiny-2312` is renamed from `mistral-tiny`, with the old endpoint to be deprecated. `mistral-ocr-4-0` (OCR 4.0) is deprecated and retires September 30, 2026, with the `mistral-ocr-4` and `mistral-ocr-latest` aliases now silently pointing to `mistral-ocr-4-1`; use `mistral-ocr-4-1` or `mistral-ocr-latest` going forward. `labs-leanstral-1-5` (Leanstral 1.5) is deprecated and retires September 30, 2026. `zai-glm-5-2` (Z.ai GLM 5.2) is deprecated and retires October 31, 2026; migrate to `zai-glm-5-3`. Devstral 2.0 moves from free to paid API access, and the temperature parameter for `pixtral-12b` was downscaled by a multiplier of 0.43, changing output behavior for existing integrations.

– Multiple deprecations with exact dates and replacement targets.

`snapshot-20261023`

`snapshot-20261022`

`snapshot-20261020`

05 Z.ai GLM 5.3 Flash model release NEW

75

`zai-glm-5-3-flash`, a `multimodal 320B-parameter (18B active) Mixture-of-Experts` model from Z.ai, ships with a `1M-token context window`, priced at \$0.15/M input, \$0.03/M cached input, and \$0.50/M output tokens.

– Full pricing and architecture specs; no usage example given.

`snapshot-20261023`

06 Mistral Medium 3.5 model release NEW

74

`mistral-medium-3-5` (Mistral Medium 3.5) is a frontier-class multimodal model with adjustable reasoning via the `reasoning_effort` parameter, released with `open`

weights under a Modified MIT license.

– Names model, tunable parameter, and license; no usage example.

snapshot-20261023

snapshot-20261020

07 Custom Guardrails for Agents and Conversations NEW

70

Adds a `guardrails` parameter for configuring Custom Guardrails directly on an Agent, and a `guardrails` field on `POST /v1/conversations` for per-request guardrail configuration.

– Exact endpoint and parameter named; internal mechanism not detailed.

snapshot-20261023

snapshot-20261020

08 Mistral Large 4 model release NEW

65

`mistral-large-4` (Mistral Large 4) entered Public Preview as an open-weight, multimodal Mixture-of-Experts model with a 1M-token context window.

– Names model, architecture, and context window; no benchmarks given.

snapshot-20261023

snapshot-20261020

09 Mistral Small 4 model release NEW

65

`mistral-small-2603` (Mistral Small 4) is a hybrid multimodal model unifying instruct, reasoning, and coding capabilities with a 256k-token context window.

– Names model and context window; lacks performance detail.

snapshot-20261023

snapshot-20261020

10 API breaking changes to streaming, validation, and tool messages

65

BREAKING

A new `p` parameter was added to chunks streamed by the Completion API, which may break applications relying on strict parsing — users of the `mistral-common` package must upgrade to version 1.8.4 or higher (official SDK users are unaffected).

`tool_call_id` is now mandatory in chat messages with the `tool` role, with requests omitting it rejected. The API now strictly enforces parameter validity, returning **Extra inputs are not permitted** for unsupported parameters instead of silently ignoring them.

– Exact parameter, version, and error text named for migration.

snapshot-20261023

snapshot-20261022

snapshot-20261020

11 Voice Agents SDK for real-time voice pipelines NEW

60

Releases Voice Agents, a managed real-time bidirectional voice pipeline (microphone capture, STT, LLM, TTS, playback) exposed through the Voice Agents SDK, which combines listening, reasoning, tool calling, and speaking into a single streaming pipeline with telephony, browser, and custom-audio interfaces for phone and web deployments.

– Names SDK and interfaces; no code sample provided. `snapshot-20261023`

`launch-20261020-26d56ceb`

12 **Voxtral Mini Transcribe 2 release** NEW 60

Releases `voxtral-mini-2602` (Voxtral Mini Transcribe 2) and `voxtral-mini-transcribe-realtime-2602` (Voxtral Mini Transcribe Realtime), adding context biasing and `diarize` support to the Audio Transcriptions API.

– Names models and flag; thin on behavior detail. `snapshot-20261022`

THINNER COVERAGE BELOW

13 **Pro Voice adaptation for speaker voice cloning** NEW 45

Adds Pro Voice adaptation, which clones a speaker's voice from 30 minutes of audio and reproduces their emotional range and speaking style across languages.

– Describes mechanism briefly; no parameter or API name. `launch-20261020-26d56ceb`

14 **Leanstral code agent for Lean 4** NEW 45

Releases `labs-leanstral-2603` (Leanstral), an open-source code agent for Lean 4 formal proof engineering.

– Names model id only; no usage detail given. `snapshot-20261022`

15 **Inline batching for batch jobs** NEW 35

Adds inline batching, letting developers create batch jobs without uploading a file first.

– States feature name only; no mechanism or example. `snapshot-20261023`

`snapshot-20261022`

16 **Documentation path redirects to /vibe/*** BREAKING 35

Documentation paths previously under `/le-chat/*` and `/mistral-vibe/*` now redirect to a new `/vibe/*` tree; links or integrations using the old paths need to be updated.

– States redirect scope; no further detail given. `snapshot-20261023` `snapshot-20261022`

17 **Mistral Moderation 2603 release** NEW 25

Releases `mistral-moderation-2603` (Mistral Moderation 2603).

– Bare model name; no description of capability. `snapshot-20261022`

↳ BREAKING ON UPGRADE

- ! The new `p` parameter added to Completion API streamed chunks may break applications relying on strict parsing of chunks; users relying on the `mistral-common` package must update to version 1.8.4 or higher (applications using the official SDK are unaffected).
- ! `tool_call_id` is now mandatory in chat messages with the `tool` role; requests omitting it will be rejected.
- ! The API now strictly enforces the validity of all parameters and returns `Extra inputs are not permitted` for unsupported parameters, whereas previously they were silently ignored.
- ! `mistral-tiny-2312` is renamed from `mistral-tiny`; the old endpoint will be deprecated.
- ! `OCR 4.0` (`mistral-ocr-4-0`) is deprecated and retires on September 30, 2026; use `mistral-ocr-4-1` or `mistral-ocr-latest` instead.
- ! `Leanstral 1.5` (`labs-leanstral-1-5`) is deprecated and retires on September 30, 2026.
- ! `Z.ai GLM 5.2` (`zai-glm-5-2`) is deprecated and retires on October 31, 2026; use `zai-glm-5-3` instead.
- ! Documentation paths previously under `/le-chat/*` now redirect to `/vibe/*`; links or integrations using the old `/le-chat/*` or `/mistral-vibe/*` paths will need to be updated.
- ! A new parameter `p` added to chunks streamed by the Completion API may break applications relying on strict parsing of chunks; users relying on `mistral-common` package must update to version 1.8.4 or higher.
- ! `mistral-ocr-4-0` (OCR 4.0) is deprecated and retires on September 30, 2026; migrate to `mistral-ocr-4-1` or `mistral-ocr-latest`.
- ! `labs-leanstral-1-5` (Leanstral 1.5) is deprecated and retires on September 30, 2026.
- ! `zai-glm-5-2` (Z.ai GLM 5.2) is deprecated and retires on October 31, 2026; migrate to `zai-glm-5-3`.
- ! Documentation paths `/le-chat/*` and `/mistral-vibe/*` now redirect to the new `/vibe/*` tree.
- ! Devstral 2.0 moves to paid API access.
- ! `tool_call_id` is now mandatory in chat messages with the `tool` role.
- ! The API now strictly enforces validity of all parameters; requests with unsupported parameters return error `Extra inputs are not permitted` instead of silently ignoring them.
- ! `mistral-tiny-2312` is renamed from `mistral-tiny`; the old endpoint will be deprecated in three months.
- ! The `p` parameter added to Completion API streamed chunks may break applications relying on strict parsing of those chunks; users of the `mistral-common` package must update to version 1.8.4 or higher.
- ! The temperature parameter for `pixtral-12b` was downscaled by a multiplier of 0.43, changing model output behavior for existing integrations using that model.
- ! `mistral-ocr-4` and `mistral-ocr-latest` now point to `mistral-ocr-4-1`; applications pinned to those aliases will silently switch model versions.
- ! `tool_call_id` is now mandatory in chat messages with the `tool` role; requests omitting it will fail.

- ! The API now strictly enforces validity of all parameters and returns **Extra inputs are not permitted** for unsupported parameters, where previously they were silently ignored.
- ! **mistral-ocr-4-0** is deprecated and retires on September 30, 2026; **labs-leanstral-1-5** retires on September 30, 2026; **zai-glm-5-2** retires on October 31, 2026.
- ! Devstral 2.0 moves to paid API access; previously free access is removed.

GOOD PICK?



GOOD REASON?



// WHY IT MADE THE LIST

DEPTH 4/5

IMPACT 4/5

NOVELTY 3/5

Persistent memory now carries repository conventions, corrections, and ruled-out approaches across sessions automatically between the IDE and CLI, and a separate launch confines agent commands with OS-kernel-enforced sandboxing (Seatbelt, bubblewrap, AppContainer).

Kiro's biggest window items were OS-enforced local sandboxing and enterprise managed-settings sign-in controls, alongside a new persistent cross-session Memory feature, Kimi K3 model support, and a Crew 0.9 release packed with dynamic dashboards, remote live windows, and new settings — plus workflow plan review, Mermaid diagram rendering, and a Markdown Review panel.

→ WHAT SHIPPED · 21 FEATURES most completely described first ? what's the number?

01 Local sandboxing for agent commands NEW

96

Adds OS-kernel-enforced sandboxing of agent-run commands via `sandbox.json` (at `~/.kiro/settings/sandbox.json` for global or `<project-root>/~/.kiro/settings/sandbox.json` for project scope), restricting filesystem and network access without a container or VM. Extends `permissions.yaml` (`~/.kiro/settings/permissions.yaml`) with `fs_read`, `fs_write`, and `network` capability rules enforced at both the tool-call layer and the OS kernel (Seatbelt on macOS, bubblewrap on Linux, AppContainer on Windows). Adds `/sandbox on` and `/sandbox` (to choose scope and network mode) in `kiro-cli chat --v3`, plus `/sandbox setup` and a one-time administrator setup prompt for Windows AppContainer initialization; the IDE gets a **Kiro: Open User Sandbox Config (JSON)** Command Palette action and a status bar indicator showing **Command sandbox: Active**. Agent network access is forced through a local proxy that filters by domain, blocking non-approved destinations while allowing package registries and other permitted services.

·
·
·

Enable local sandboxing globally so every Kiro session on your machine confines agent commands to your project directory and denies reads of credential paths like `~/.ssh` and `~/.aws`.

 json`~/.kiro/settings/sandbox.json`

Turn on sandboxing for a single CLI session without editing any file — useful when starting a long refactor task you want OS-level restricted.

```
$ kiro-cli chat --v3
# then inside the session:
/sandbox on
```

Override the default network policy for a specific project by adding `sandbox_network` rules to the project-scoped permissions file, allowing only approved registries.

```
yaml

<project-root>/..kiro/settings/permissions.yaml
```

— Full config paths, commands, and platform mechanisms named `launch-20261014-a0c24100`

02 Enterprise sign-in controls via managed-settings.json NEW

93

Adds an admin-protected `managed-settings.json` file that Kiro IDE and CLI read on each managed device to enforce sign-in rules. `deny` rules with `exclude` (allowlist) or `match` (blocklist) restrict which sign-in methods — `idc`, `builder_id`, `google`, `github`, `social` — are shown; `idc_start_url` and `idc_region` under the top-level `settings` key prefill IAM Identity Center details; `signin_help_url` surfaces an org IT help link on the sign-in screen; and administrators can enable or disable `device code flow` (used by Kiro CLI in remote, browser-less environments) from the Kiro console.

Lock a managed fleet to IAM Identity Center only, pre-populate the org start URL and region, and surface an IT help link — so users never see personal sign-in options or have to look up Identity Center details.

```

{
  "settings": {
    "idc_start_url": "https://my-org.awsapps.com/start",
    "idc_region": "us-east-1",
    "signin_help_url": "https://it.my-org.com/kiro-help"
  },
  "rules": [
    {
      "deny": {
        "exclude": ["idc"]
      }
    }
  ]
}

```

Block only specific personal sign-in methods (Google and GitHub) while leaving AWS Builder ID and IAM Identity Center available to users.

```

{
  "rules": [
    {
      "deny": {
        "match": ["social"]
      }
    }
  ]
}

```

Enable or disable device code flow for Kiro CLI in remote environments where no browser is available — controlled per org from the Kiro console.

 In the Kiro console, navigate to sign-in settings and toggle **Device code flow** on or off to control whether Kiro CLI users in browser-less remote environments can authenticate via device flow.

— Every rule key and value named with runnable config examples `launch-20261013-eff0b881`

03 Persistent Memory in IDE and CLI NEW

88

Kiro IDE 1.4.0+ and Kiro CLI 2.29.0+ gain persistent Memory that carries working preferences, repo-specific conventions, corrections, and debugging insights across chat sessions, scoped as global or per-repository and shared between IDE and CLI on the same machine. Memory is off by default and the **Kiro Agent: Memory** setting

(default changed to Off in 1.4.10, preserving existing choices) offers three modes — **Use and learn**, **Use but don't learn**, and **Use only** — with chat showing when Kiro accesses or changes a memory entry; manage memories via the `/memories` CLI command or the Memories view in the IDE's activity bar.

Review and manage what Kiro has learned about your project before starting a new CLI session.

```
$ /memories
```

Enable memory in Kiro IDE so it automatically learns and applies your preferences without being reminded each session.

📌 In Kiro IDE: open the Memories view from the activity bar > open its settings > select **Use and learn** to allow Kiro to use existing memories and automatically create or refine memories as you work.

Configure your preferred memory mode after the default changes to Off, so Kiro learns from sessions without requiring manual re-enabling each time.

📌 In Kiro, open Settings, search for **Kiro Agent: Memory**, and set it to **Use and learn** to restore the previous behavior for new sessions.

Prevent Kiro from updating memories while still benefiting from previously stored context — useful on shared or sensitive repositories.

📌 1. Open Settings and navigate to Kiro Agent: Memory. 2. Set the value to **Use only**. 3. Kiro will read existing memories in chat but will not create or modify any memory entries during the session.

– Full mechanism, settings, and commands named across three releases

IDE 1.4.10: Memory Across Chat Sessions...

IDE: Memory Setting Defaults to Off, Wo...

launch-20261020-288b45b1

04 Workspace trust prompts for V3 sessions NEW

84

Kiro CLI shows a "Trust this workspace?" screen before loading hooks, agents, MCP servers, steering, workflows, or language server configuration from a folder, saving the answer per folder under `~/.kiro/workspace-roots/`; remove the `workspaceTrusted` key from the per-workspace file to reset your choice.

Reset workspace trust for a specific project so Kiro prompts you again on next launch.

📌 In the per-workspace file listed on the 'Trust this workspace?' screen (under `~/.kiro/workspace-roots/`), remove the `workspaceTrusted` key and save. Kiro

will prompt for trust again the next time you start a V3 session in that folder.

– Exact storage path and reset key given `CLI: Workspace Trust, Workflow Plan Ste...`

05 **Kimi K3 model support** NEW

78

Adds Kimi K3 (Moonshot AI open-weight model) with experimental support in Kiro IDE, CLI, Crew, and Web for Pro, Pro+, Pro Max, and Power customers in the US geography – it accepts text and images, offers a 1M-token context window, and carries a 2.4x credit multiplier; reload via an IDE/CLI restart, `kirocrew restart` for Crew, or a browser refresh.

Reload the latest model list in Kiro Crew so Kimi K3 becomes selectable for long-context coding or reasoning tasks.

```
$ kirocrew restart
```

– Names tiers, context size, multiplier, and reload command

`Models: Kimi K3 now available in Kiro (...)`

06 **Telegram Bot API proxy and forum-topic sessions** NEW

75

Adds the `TELEGRAM_API_BASE_URL` environment variable to route the Telegram Bot API through a reverse proxy where the public host is blocked, and adds Telegram forum-topic-per-session support, treating each private-chat forum topic as its own session.

Route all Telegram Bot API traffic through a self-hosted reverse proxy when the public Telegram host is blocked on your network.

```
$ export TELEGRAM_API_BASE_URL=https://telegram-proxy.internal/bot
```

– Names the exact environment variable with a runnable example

`Crew: Crewmate Profiles, Dynamic Dashbo...`

07 **Workflow plan review and step tracking** IMPROVED

74

With Workflows enabled, Kiro shows proposed steps in a permission card before execution, letting you Allow, Deny, or submit feedback via chat input to revise the plan; approved steps then follow those plan-feedback corrections rather than the original request. The plan view also displays a compact, expandable step tree for reading step details, and workflow step updates in the parent chat render as a single status row (`Info` , `Completed` , `Needs input` , or `Failed`) that expands to the full message and links to the step's session.

Review and approve proposed workflow steps before they run, or request changes inline.

1. Enable Workflows in Kiro settings. 2. Start a workflow-driven task in chat — Kiro displays a permission card listing the proposed steps and their descriptions. 3. Click Allow to proceed, Deny to cancel, or type your requested changes in the chat input and press Enter to revise the plan before it executes.

– Mechanism and UI states named, but no API or flag

IDE: Markdown Review, What's New Announ...

CLI: Workspace Trust, Workflow Plan Ste...

IDE 1.4.10: Memory Across Chat Sessions...

IDE: Memory Setting Defaults to Off, Wo...

08 New kirocrew CLI subcommands and flags NEW

73

Adds `kirocrew memory create-store` to create a declared memory store, `kirocrew cron add --managed-by` to let a tool own a cron job by key, and `--ami` and `--extra-packages` flags to `kirocrew cloud launch`.

Let an automation tool own a cron job by a stable key so it can update or remove the job without knowing its internal ID.

```
$ kirocrew cron add --managed-by my-deploy-tool "0 9 * * 1" run-weekly-scan
```

– Three exact subcommands and flags named verbatim

Crew: Crewmate Profiles, Dynamic Dashbo...

09 Markdown Review panel NEW

73

Opens any `.md` file in a rendered review view where you can comment on exact passages and send all comments to Kiro in one batch; accessible from the editor title bar, Command Palette, Specs view, or the Review button on supported spec documents.

Review a Markdown spec file and send all inline comments to Kiro at once, without switching to a separate tool.

1. Open a `.md` file in Kiro, then click the Review button in the editor title bar (or open the Command Palette and search `Markdown Review`) to open the rendered review panel. Highlight a passage, add your comment, repeat for any other sections, then click Send to deliver all comments to Kiro in one batch.

– Multiple UI entry points given with worked example

IDE: Markdown Review, What's New Announ...

10 MCP server agent grants and auto-upgrade NEW

70

Adds `kiro-cli mcp add --agent` to grant a new MCP server to an agent whose tools list did not include it, and upgrades the agent file for V3 automatically when `chat.enableAutoAgentUpgrade` is on.

Add an MCP server and grant it to a specific agent that did not previously include it in its tools list.

```
$ kiro-cli mcp add --agent
```

– Exact flag and config key given with example command

CLI: Workflows Start Only When You Ask ...

11 Mermaid diagram rendering in CLI chat IMPROVED 70

Kiro CLI renders Mermaid flowcharts, sequence diagrams, state diagrams, and entity-relationship diagrams as terminal art directly in chat, falling back to Mermaid source for unsupported syntax, oversized diagrams, or when ASCII art is off. Flowcharts additionally render subgraphs as titled frames, apply style-class colors, wrap node labels across up to four rows, and render `<br/>` as a second line.

Ask Kiro to describe a system architecture and get an interactive Mermaid diagram rendered as terminal art in the chat pane.

📌 In the Kiro chat panel, ask: 'Draw a sequence diagram for our OAuth login flow.' Kiro renders the diagram as terminal art inline; if the diagram is too large or uses unsupported syntax, it falls back to showing the raw Mermaid source.

– Lists diagram types and rendering rules, no toggle named

CLI: Workspace Trust, Workflow Plan Ste...

CLI: Mermaid Diagram in Chat (2026-10-1...

12 New Crew settings for messaging, display, voice, and chat

NEW

68

Adds `Settings → Messaging Channels → Slack → Connect new sessions to Slack automatically` to thread new persistent sessions into a Slack DM with the bot; `Settings → Import / Export` option to include persistent chat history in an export; `Settings → Display → Terminal` options for `Cursor style` and `Reuse the current terminal` (`Ctrl+` toggles the terminal and `Ctrl+Shift+` opens an additional one); a `Settings → Display → Zoom` exact-percent input; `Settings → Voice` controls including a `Use a key to start dictation` switch, a microphone level test, and a `Custom vocabulary list` for Amazon Transcribe; and `Settings → Chat → Advanced → Compaction Time Limit (seconds)` as an editable field.

– Every setting path named verbatim across six areas

Crew: Crewmate Profiles, Dynamic Dashbo...

13 Crew 0.9 default and settings reorganization

BREAKING

New installs default **Settings → Chat → Transcript → Response Verbosity** to **Answer only** (existing configs keep their stored level); the **Ultra** verbosity level is retired, with stored **Ultra** configs now behaving as the default; **Settings → Display → View → Translucent panels** is now off by default (toggle to restore the prior frosted-glass look); chats that ran on a peer crew through the retired turn relay now open read-only; and **Agent Capabilities** is renamed **Customize**, with **Pre-started Sessions** and the Warm Pool card moved under **Settings → Agent Harness** as **Subagent Settings** and **Warm Pool Settings**.

– Every renamed setting and new default named verbatim

Crew: Crewmate Profiles, Dynamic Dashbo...

14 **Generate Commit Message from unstaged changes** **IMPROVED**

67

Generate Commit Message now works from unstaged working-tree changes, including untracked non-ignored files, when nothing is staged, with an explanation shown when repository changes can't be read.

Draft a commit message from work-in-progress changes before staging anything, so you can describe intent while context is fresh.

📌 In the Kiro Source Control view, click **Generate Commit Message** without staging any files. Kiro reads working-tree changes, including untracked non-ignored files, and drafts the message — noting when it is using unstaged changes.

– Exact command and behavior with worked example `IDE: Markdown Review, What's New Announ...`

15 **New Crew feature previews: Dynamic Dashboard and Remote crew sessions**

NEW

65

Adds two Developer feature previews: **Settings → Developer → Feature Previews → Dynamic Dashboard**, which opens a Dashboard tab per crewmate with twelve built-in report pages (standup, roadmap, kanban, and more) drawn from crew logs, and **Settings → Developer → Feature Previews → Remote crew sessions**, which opens a peer crew's session as a live window onto that crew's own transcript, running state, and approvals without storing data locally.

Enable the Remote Crews live-window preview so you can monitor and approve tasks on a peer machine without leaving your local IDE.

📌 In Kiro, go to **Settings → Developer → Feature Previews → Remote crew sessions** and toggle it on. Your sidebar will show peer crews grouped by machine with Online / Reconnecting / Error / Offline badges. Click any session to open it as a live window.

– UI path and page count given, no walkthrough UI path plus step-by-step example to enable it

Crew: Crewmate Profiles, Dynamic Dashbo...

THINNER COVERAGE BELOW

16 Model fallback command in CLI NEW 55

`/model fallback` lets an active V3 CLI session switch to a fallback model automatically when the primary model is unavailable.

Set a model fallback in an active V3 session so the agent switches automatically when the primary model is unavailable.

```
$ /model fallback
```

– Only shown via example, no bullet description CLI: Workspace Trust, Workflow Plan Ste...

17 What's New announcement browser NEW 50

The status bar and Agent Focus footer entry now open a list of announcements with an unread count, docs and changelog links per entry, and a feature-request link.

– UI location named but no deeper interaction detail

IDE: Markdown Review, What's New Announ...

18 MCP config file restricted to OS account IMPROVED 43

Limits the MCP configuration file (which can hold server credentials) to the current OS account on macOS and Linux, protecting files created by earlier versions on next startup.

– Names the protected file but no path or command

IDE 1.4.10: Memory Across Chat Sessions...

19 Powers resilience to MCP server failures IMPROVED 38

Powers continue running when an MCP server fails to connect, surfacing a chat warning instead of an error card; servers requiring sign-in still prompt as before.

– Behavior described but no setting or command to act on

IDE: Memory Setting Defaults to Off, Wo...

20 Workflows only run when explicitly requested

BREAKING

35

With Workflows enabled in a V3 session, a workflow no longer starts automatically when you turn the feature on — it now only runs when you explicitly ask for one.

– Before/after stated but no config or command given

CLI: Workflows Start Only When You Ask ...

21 V2 grep tool parallel scanning

IMPROVED

30

The V2 grep tool now scans files in parallel across CPU cores for faster searches on large projects.

– States the change but no benchmark or flag

CLI: Workspace Trust, Workflow Plan Ste...

↳ BREAKING ON UPGRADE

- ! With Workflows enabled in a V3 session, a workflow no longer starts automatically when you turn the feature on — it now only runs when you explicitly request one.
- ! The **Kiro Agent: Memory** setting default changes to Off; new sessions will have memory disabled unless you explicitly choose **Use and learn**, **Use but don't learn**, or **Use only**.
- ! New installs default **Settings → Chat → Transcript → Response Verbosity** to **Answer only**; existing configs keep their stored level.
- ! The **Ultra** verbosity level is retired; configs that stored **Ultra** now behave as the default level — choose another level under **Response Verbosity** if you want longer replies.
- ! Translucent panels are now off by default (top bar, message box, and search fields render as solid cards); enable **Settings → Display → View → Translucent panels** to restore the frosted-glass appearance from 0.8.
- ! Chats that ran on a peer crew through the retired turn relay open read-only; sending to them is no longer possible.
- ! **Agent Capabilities** is now labeled **Customize**; **Pre-started Sessions** and the Warm Pool card are now under **Settings → Agent Harness** as **Subagent Settings** and **Warm Pool Settings**.

GOOD PICK?



GOOD REASON?



04 SAM v0.1.0

Agentic Infrastructure

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.

// WHY IT MADE THE LIST

DEPTH 4/5

IMPACT 4/5

NOVELTY 5/5

Debuts a private, policy-enforced network for AI agents — a control plane, router, and node sidecar that put MCP servers, A2A agents, or OpenAI-compatible models on an identity-verified mesh with deny-by-default policy and JS/Python SDKs, a capability no other tracked tool offers.

Agent Mesh debuted in v0.1.0 as a full private network stack for AI agents — control plane, router, node sidecar, single-binary trial mode, JS/Python SDKs, Biscuit credentials, and multi-identity enrollment — then v0.1.1 hardened operations with rotatable signing keys, resilient workload-issuer resolution, bootstrap re-enrollment, and new auth-refusal metrics.

---> WHAT SHIPPED • 13 FEATURES most completely described first ? what's the number?

01 Operator-owned signing key rotation NEW 95

The `--signing-key-path` flag accepts a PEM file with an Ed25519 key (for Biscuits) and a P-256 key (for ES256 JWTs on `/sts/token`); the file is re-read every 30 seconds so rotation requires only a file edit with no restart.

Supply operator-owned signing keys so the control plane can rotate credentials by editing a Secret Manager-backed file on Cloud Run without restarting.

```
$ agentmesh-one --signing-key-path /secrets/signing-keys.pem --workload-issuer https://issuer.example.com
```

— Exact flag, file formats, refresh interval and a runnable command v0.1.1

02 Member auth-refusal metrics NEW 80

Exports `agentmesh_control_plane_member_auth_refusals_total{reason}` with labeled reasons (`malformed` , `invalid_biscuit` , `unknown_node` , `banned` , `session_expired`) plus a log line with the peer id.

Query the auth-refusal counter by reason to triage member authentication failures in a running mesh.

```
$ curl -s http://localhost:9090/metrics | grep
agentmesh_control_plane_member_auth_refusals_total
```

– Metric name, all reason labels, and a query command given v0.1.1

03 Improved issuer configuration handling IMPROVED 75

Improves issuer handling: the control plane now starts even when a `--workload-issuer` is unreachable, exposing `agentmesh_control_plane_workload_issuer_ready{issuer}`, retrying resolution with backoff, and returning 401 responses that name the unresolved issuer; additionally, the `--issuer` and `--workload-issuer` flags now accumulate when repeated instead of the last value replacing the others.

Register multiple workload issuers so the control plane accepts tokens from all of them, each accumulating rather than overwriting.

```
$ agentmesh-one --workload-issuer https://issuer-a.example.com --
workload-issuer https://issuer-b.example.com
```

– Metric and behaviour named, no query example attached Exact flags named with a runnable example of the new behaviour

v0.1.1

04 Bootstrap-token re-enrollment NEW 70

The `--bootstrap-token-path` flag enables re-enrollment: a node receiving `401 node credential required` automatically re-enrolls using the bootstrap token and accepts the response only under an already-trusted control plane key.

– Flag and error condition named, no runnable example provided v0.1.1

05 Node sidecar for MCP, A2A and model mesh access NEW 65

`agentmesh-node` sidecar puts MCP servers, A2A agents, or OpenAI-compatible models on the mesh, with `ext_authz` integration for Envoy and Istio, and Android support.

– Names integration points but no exact config shown v0.1.0

THINNER COVERAGE BELOW

06 Single-binary deployment mode NEW 55

`agentmesh-one` runs the entire mesh on one port for rapid local trials, letting an evaluator spin up a complete local mesh in seconds.

Spin up a complete local mesh in seconds to evaluate policy enforcement before deploying a sidecar topology.

```
$ agentmesh-one
```

– Ships a runnable single command to try the mesh v0.1.0

07 Control plane for enrollment and policy NEW 50

`agentmesh-control-plane` handles enrollment, policy management, signing-key rotation, and a web console, backed by Postgres or SQLite.

– Names the binary and backing stores but no flags or commands v0.1.0

08 Enrollment identity sources NEW 50

Supports OIDC, Kubernetes service accounts, SPIFFE, and bootstrap tokens as enrollment identity sources for joining the mesh.

– Four providers named, no config shown to wire them up v0.1.0

09 Biscuit credentials with delegation narrowing NEW 50

Issues short-lived Biscuit tokens as in-mesh credentials, letting members narrow their own credential before delegating it to an agent.

– Mechanism explained but no command or format detail v0.1.0

10 Validated scale and latency NEW 50

Validated at 10,000 simultaneous members across 12 routers, with control-plane p99 latency under 250 ms.

– Concrete numbers given but nothing actionable to do v0.1.0

11 Router for cross-machine brokering NEW 45

`agentmesh-router` provides authenticated connection brokering, service directory, and traffic relay for members that cannot reach each other directly.

– Describes role of the binary without setup detail v0.1.0

12 JavaScript and Python SDKs NEW 40

JavaScript and Python SDKs let agents join the mesh in-process without a sidecar.

– Names the SDKs but no code sample given v0.1.0

13 Deployment guides for Kubernetes, Cloud Run, Codespaces, SkyPilot

Provides Helm charts and deployment guides covering Kubernetes, Cloud Run, Codespaces, and SkyPilot.

– Names four targets but no guide content detailed v0.1.0

GOOD PICK?



GOOD REASON?



05 Anthropic

snapshot-20261021

Frontier Models

Anthropic develops Claude, an AI assistant API for developers to build applications with advanced language understanding and reasoning capabilities.

// WHY IT MADE THE LIST

DEPTH 4/5

IMPACT 4/5

NOVELTY 4/5

A user profiles API scopes safety enforcement and bans to individual end customers instead of the whole organization, with webhooks for restriction lifecycle events — closing a real gap for platforms built on Claude; Claude Models on Google Cloud adds IAM-controlled, Marketplace-billed access.

Anthropic rolled out per-customer safety enforcement via a new user profiles API and admin webhooks, gave Claude Managed Agents org-level web fetch/search domain controls, bumped every SDK and the CLI to new breaking major versions on API `2026-10-21`, added a compliance endpoint for user instruction preferences, and took Fast mode for Claude Opus 5.5 to general availability.

↳ WHAT SHIPPED · 6 FEATURES most completely described first ? what's the number?

01 User profiles API and enforcement status tracking NEW

85

Adds `/v1/user_profiles` endpoints (supported in the SDK and `ant` CLI) to associate API requests with specific end customers, scoping safety enforcement actions to individual users rather than the whole organization. Profiles expose new `enforcement_status` and `enforcement_actions` fields showing whether Anthropic has restricted or banned a customer, and the list-profiles endpoint supports an `enforcement_status[]` filter to surface only profiles with an open restriction or ban.

– Endpoints and fields named, no usage example given. snapshot-20261021

snapshot-20261022

02 Org-level web fetch and search domain controls for Managed Agents

NEW INTEGRATION

NEW

85

Adds an org-level web fetch toggle on the Capabilities page of the Claude Console; when off, Messages API requests including the web fetch tool fail with a `400` error, and `web_fetch` calls in Claude Managed Agents return an error result to the agent. Adds allowed/blocked domain filtering for web fetch, where a URL (including redirect destinations) must pass both the org's domain list and the `allowed_domains` or

`blocked_domains` list on the agent's `web_fetch` tool. Claude Managed Agents now also enforce the org's `Allow web search` setting and web search domain restrictions, so agents receive only search results passing both the org's domain restrictions and the `allowed_domains` / `blocked_domains` list on the agent's `web_search` tool.

– Names UI path, tool names and error codes precisely. `snapshot-20261022`

03 Major SDK and CLI version bump for API 2026-10-21

BREAKING

80

Releases API version `2026-10-21` with major SDK versions defaulting to it: Python SDK `2.0.0`, TypeScript SDK `1.0.0`, C# SDK `13.0.0`, Go SDK `2.0.0`, Java SDK `3.0.0`, PHP SDK `1.0.0`, Ruby SDK `2.0.0`, and `ant` CLI `2.0.0`. These new major versions carry independent breaking changes and switch licensing from the MIT License to Apache 2.0.

– All version numbers named; specific breaking changes unspecified. `snapshot-20261022`

04 Compliance API endpoint for user instruction preferences

NEW

80

Adds `GET /v1/compliance/organizations/{org_uuid}/user_preferences` endpoint (requiring `read:compliance_user_data` scope) to return each Enterprise member's saved Instructions for Claude and Global instructions from Cowork.

– Full endpoint, required scope and payload content named. `snapshot-20261016`

05 Organization webhooks for profile lifecycle events

NEW

78

Adds `/v1/organizations/webhooks` endpoints to the Admin API (beta, requires the `webhooks-2026-10-07` beta header) to register webhook listeners for user-profile restriction and ban lifecycle events.

– Endpoint and beta header named, no payload example. `snapshot-20261021`

06 Fast mode for Claude Opus 5.5 reaches general availability

IMPROVED

70

Fast mode for Claude Opus 5.5 is now generally available — the `fast-mode-2026-02-01` beta header is no longer required, and the model has a default fast mode rate limit across Start, Build, and Scale usage tiers. Fast mode is also now available in Claude on Microsoft Foundry for Azure-hosted deployments.

– Names tiers and header removal but no numeric rate limit. `snapshot-20261016`

↳ **BREAKING ON UPGRADE**

! The new SDK and CLI major versions (Python `2.0.0`, TypeScript `1.0.0`, C# `13.0.0`, Go `2.0.0`, Java `3.0.0`, PHP `1.0.0`, Ruby `2.0.0`, `ant` CLI `2.0.0`) contain breaking changes

independent of the API version bump.

- ! The new SDK and CLI versions are licensed under Apache 2.0 instead of the MIT License.
- ! When the org-level web fetch toggle is turned off, any Messages API request that includes the web fetch tool fails with a **400** error.

snapshot-20261023

Fast mode is now available for Claude Opus 5.5 on Claude Models on Google Cloud (Anthropic managed).

NEW INTEGRATION

Claude Models on Google Cloud (Anthropic managed)

GOOD PICK?



GOOD REASON?



// END OF TRANSMISSION

The AI Toolchain · This Week's Highlights · October 23, 2026

The full firehose — every feature release, every day — is in [the newsletter](#).

▷ **Subscribe**