THE AI TOOLCHAIN

NO. 095

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED NOVEMBER 19, 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 101 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 18 of 19 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. Gemini API's Genie 4 streams interactive world generation — steerable video and audio — straight into

vim ? help

an app. SLayer now exports traces, metrics and logs over OTLP, so its pipelines land in the backend you already page on. And Lovable blocks publishing outright while critical security findings are open.

BUILD

Stream a steerable, generated world into an app instead of rendering clips offline

Genie 4 generates an interactive world in real time and streams video and audio to the client, which is a different shape of product from batch video generation — the user steers it rather than waiting on a render. Runway's API adds Seedance 2.5 for both image-to-video and text-to-video, so one endpoint covers both entry points.

Gemini API · Runway

OPERATE

See a pipeline's traces and latency in the backend you already run, not a vendor dashboard

SLayer v1.9.0 emits OpenTelemetry tracing, metrics and logs to any OTLP collector, so ingest runs correlate with the rest of your system instead of living in a separate UI; it also now reports when an ingest only touched key metadata, which otherwise looks like a silent no-op. DeepInfra adds richer live performance metrics to its monitoring API for the same reason — provider-side latency becomes something you can chart and alert on.

SLayer · DeepInfra

GOVERN

Stop an unsafe deploy at the publish button rather than in review

Lovable's workspace-wide safeguard refuses to publish while critical security findings are open, turning a scan result from advisory into a gate nobody can click past. Arcade tightens API key and organization controls and introduces a client-identity/OAuth model for MCP servers, so a tool server can tell which client is calling it instead of treating every caller as one trusted identity.

Lovable · Arcade

Does **Opus 5** read better? 👍 👎

BUILD

◆ AI Coding Agents

Cognition Devin Desktop

Feedback ⓘ

changelog · Nov 19, 2026 | CHANGELOG ↗ | RANK

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

> Devin Desktop ships Devin Bots for outcome-driven autonomous work, moves local sessions to a background process so they survive app restarts and support multiple clients, adds tabs to the Agent Command Center, and gives admins a proxy auto-detection device policy alongside a batch of smaller editor and Devin Local UX improvements.

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

01 **Proxy auto-detection for Desktop language server** Not checked yet

NEW

90

Managed macOS, Windows, and Linux machines can enable proxy auto-detection for the Devin Desktop language server via the `DetectProxy` device policy or the `devin.network.detectProxy` setting, pushed through an MDM or device management console without requiring per-user manual configuration.

As an enterprise admin, enable proxy auto-detection for the language server on all managed machines without requiring users to configure it manually.

📍 In your MDM or device management console, set the DetectProxy device policy to enabled, or push the `devin.network.detectProxy` setting to managed macOS, Windows, and Linux machines.

— Names exact policy and setting with admin setup steps. changelog-20261119-a1531d72

THINNER COVERAGE BELOW

02 **Starred default agent for new sessions** Not checked yet NEW

55

New sessions now start on the agent starred in Settings > Agents, or on the agent picked last.

Set a preferred default agent so every new Devin Local session opens on the right model without manually selecting it each time.

📍 Go to Settings › Agents, find the agent you want as default, and click the star icon next to it.

– Exact settings path and steps provided for setup. changelog-20261119-a1531d72

03 **Add page elements to chat from browser preview** Not checked yet

NEW

55

In IDE mode, picking a page element or console output in a Devin Local session's browser preview now adds it to the chat composer.

– Describes exact interaction mechanism for this feature. changelog-20261119-a1531d72

04 **Local sessions run as background process** Not checked yet IMPROVED

50

Local sessions now run in a background process, so closing or reopening Devin Desktop or the CLI no longer stops a session, and the same session can be opened in multiple clients at once.

– Explains mechanism and benefit without a command. changelog-20261119-a1531d72

05 **Zoom controls in image viewer** Not checked yet NEW

50

The image viewer now includes zoom buttons and supports `Cmd/Ctrl` + `+` / `-` keyboard shortcuts.

– Exact keyboard shortcut named for a simple feature. changelog-20261119-a1531d72

06 **Devin Bots for outcome-driven automation** Not checked yet NEW

45

Devin Desktop adds Devin Bots: give a Bot an outcome to own in its chat and it plans the work and runs sessions until the work is done.

– Describes behavior but no setup steps or limits given. changelog-20261119-a1531d72

07 **Tabs in Agent Command Center** Not checked yet NEW

45

The Agent Command Center now supports tabs, letting users keep several sessions and pages open in one window, each with its own independent Back and Forward history.

– Clear UI change but no navigation path specified. changelog-20261119-a1531d72

08 **Files tab preview for PDF, CSV/TSV, and media** Not checked yet

IMPROVED

40

The Files tab now previews local PDF files, renders CSV and TSV files as tables, and plays more media types.

– Names formats supported but no interaction detail. changelog-20261119-a1531d72

09 **Masked password prompts in command palette** Not checked yet

IMPROVED

35

Password prompts in the command palette, such as the Remote SSH password prompt, now hide what you type.

— Specific prompt named; minor UX fix. changelog-20261119-a1531d72

---

10 **Nested subagent output in Devin Local** Not checked yet IMPROVED

30

Devin Local now nests the output of each subagent under it and shows how many subagents are running.

— Describes UI change only, no further mechanism. changelog-20261119-a1531d72

└──▷ **ALSO FROM THESE RELEASES**

WAS THIS USEFUL? 👍 👎

Command Code

Feedback ⓘ changelog · Nov 19, 2026 CHANGELOG ↗

RANK

> Command Code's latest changelog introduces default auto-continuing goals with an opt-out config key, a shortcut to background running sub-agents, a cross-app mini chat overlay, and expanded agent control over the app itself, alongside smaller additions like a new theme, an easter egg command, and streaming fixes for chat integrations.

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

01 Goal auto-continuation with opt-out config

Not checked yet NEW

75

Goals now keep running by default across turns until verified, out of budget, or stalled, instead of stopping after each turn. This new default can be disabled with the `goalAutoContinue: false` config key.

Prevent `/goal` from auto-continuing after each turn — useful when you want to review intermediate results before the agent proceeds.

yaml

```yaml
goalAutoContinue: false
```

— Explains new default behavior and exact config key to disable it.

changelog-20261119-1c9feb77

02 Background a running sub-agent with Ctrl+B

Not checked yet NEW

60

Press `Ctrl+B` to move a running sub-agent to the background so you can keep chatting while it continues working.

Push a long-running sub-agent task to the background mid-session so you can ask follow-up questions without waiting for it to finish.

```
$ Ctrl+B
```

— Clear mechanism and exact shortcut, but no further scope detail.

changelog-20261119-1c9feb77

03 Vice theme via /theme

Not checked yet NEW

60

Adds the `vice` theme — a Miami-dusk palette with a chrome-on-sunset header — selectable via `/theme`.

— Names theme and selection command with visual description. changelog-20261119-1c9feb77

04 **Persistent /rc streaming to Telegram and Discord** Not checked yet

IMPROVED

60

`/rc` replies now keep streaming to Telegram and Discord after a model switch, `/clear`, or `/resume`, where previously streaming would stop.

— Names the command and the three triggers that previously broke it.

changelog-20261119-1c9feb77

THINNER COVERAGE BELOW

05 **Mini chat overlay from any app** Not checked yet NEW 55

Open a mini chat window from any app using Shift+Cmd+Space.

— Exact shortcut given, but behavior described only briefly. changelog-20261119-1c9feb77

06 **Agent-driven app control** Not checked yet NEW 55

The agent can now control the app directly: rename a chat, start or fork a chat, open a link, show a pane, and send notifications.

— Lists concrete actions but no commands or flags to invoke them.

changelog-20261119-1c9feb77

07 **GTA VI intro easter egg command** Not checked yet NEW 55

The `/gta` command replays the GTA VI launch-day intro, and `/gta off` skips it.

— Exact commands named but trivial, thinly described feature. changelog-20261119-1c9feb77

08 **Startup announcements without CLI release** Not checked yet

IMPROVED

30

Startup announcements about new models and free tiers now appear without requiring a CLI release.

— States the change but no mechanism or surface detail. changelog-20261119-1c9feb77

WAS THIS USEFUL? 👍 👎

yetone magpie

Feedback ⓘ v0.1.4555 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 deeper per-project and per-agent control: project-scoped plugin installation, accurate per-vendor image token accounting, new agent support with full MCP tool schemas, and several targeted additions to routing status, PII-model storage, session management and Pi package installation.

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

01 **Project-scoped plugin application** Not checked yet NEW

90

Apply a whole library plugin to a single project — including its skills, MCP servers, subagents and hooks — via `ProjectPlugin`. Claude Code hooks are written per-entry into `.claude/settings.local.json` and Codex hooks into `.codex/hooks.json`, managed independently of agent-wide settings; removing the plugin reverses only what magpie placed, leaving user-edited entries untouched.

— Names exact config files and removal behavior, but no CLI example v0.1.4546

02 **Custom storage location for PII-redaction models** Not checked yet

NEW

85

Set a custom directory for local PII-redaction models via the `RedactModelDir` setting, which is read on each use so a running instance follows changes immediately. Downloaded PII models (`privacy-filter` and `astrlink-guard`) can be moved to a user-chosen folder or back to magpie's cache via the new Move action in the Advanced settings UI.

— Exact setting key, model names, and UI action named v0.1.4551

03 **Delete and restore Grok Build sessions** Not checked yet NEW

85

Delete Grok Build sessions from the Sessions page via `/api/sessions/manage` and `/api/sessions/delete` , which now remove the session's rows from Grok Build's `session_search.sqlite` FTS5 index alongside the session folder so orphaned search results no longer appear.

— Names both endpoints and the index file they clean up v0.1.4549

04 **Install Pi packages by git URL or exact name** Not checked yet

NEW

85

The Pi library search field now accepts `git:github.com/<owner>/<repo>` and `https://github.com/<owner>/<repo>@<ref>` forms in addition to npm package names, letting packages install directly without a prior market listing. The `PiMarketPackage` struct now exposes `kind` (`npm` or `git`) and `id` fields to identify how each package's source pins it.

— Exact URL syntax and struct fields given, ready to use v0.1.4548

05 **Per-vendor image token accounting** Not checked yet IMPROVED

65

Inline images sent in a request now have their context-window token cost calculated from the image's own pixel dimensions using the vendor's protocol-specific rule (Anthropic, OpenAI Responses, Chat, Gemini), rather than the previous flat 1600-token estimate for every image.

— Clear before/after and named vendors, but automatic with no user action v0.1.4540

06 **Antigravity and agy agent support in hub gateway** Not checked yet

NEW

60

Adds support for Antigravity and `agy` agents in the hub on-demand gateway, sending them the full set of hub tools rather than routing through `find_tools` and `call_tool` stubs, avoiding double-search behaviour.

— Names agents and mechanism but no user-facing action v0.1.4547

THINNER COVERAGE BELOW

07 **Routing group fallback status text** Not checked yet NEW

50

Routing groups with no rules now show a status line explaining fallback order: when smart routing is active, the member whose allowance renews soonest goes first (ties broken by list order); when a specific routing mode is set under Advanced, that mode name is shown instead.

— Explains logic but only points to a UI label, not a config key v0.1.4555

WAS THIS USEFUL? 👍 👎

Yuval Sarel cones

Feedback ⓘ v0.4.21 NOTES ↗ RANK

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

> cones adds per-session Codex provider selection, letting users target providers like Amazon Bedrock without touching their global config.

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

01 Per-session Codex provider selection

Not checked yet NEW

95

Set the Codex model provider per launch with `cones launch --provider`, or persistently with the `codex_provider` setting in `defaults`, to target providers such as `openai` or `amazon-bedrock`. When a provider other than the one in `config.toml` is used, cones automatically creates a separate home directory at `~/.codex-<provider>` with its own daemon, without restarting the daemon already running in `~/.codex`. Resumed and forked threads stay on the provider they started with, and Codex jobs follow the `defaults.codex_provider` setting.

Launch a Codex session against Amazon Bedrock without changing your default config.toml.

```
$ cones launch --dir ~/my-project --provider amazon-bedrock
'Refactor the auth module'
```

— Names flag, config key, directory convention and behaviour in full. v0.4.21

WAS THIS USEFUL? 👍 👎

Tester Army e2e

Feedback ⓘ `@e2e-dev/revyl@0.0.1` NOTES ↗

RANK

e2e is an end-to-end testing framework that automates UI tests across web and mobile applications.

> e2e ships a new `@e2e-dev/core` package for driving browsers and mobile apps without a test runner — complete with secrets, encrypted session-state export, and agent support — alongside a new moli() browser provider, CI pipeline generation in `e2e init`, and a fixture system overhaul that validates and initializes fixtures lazily at collection time.

└──▷ **WHAT SHIPPED · 21 FEATURES** most completely described first ? what's the number?

01 **New @e2e-dev/core package for runner-free automation**

Not checked yet NEW 96

New package `@e2e-dev/core` lets you drive a browser or native app from plain TypeScript with no test runner, config file, or report: launch(web()) prepares an engine and returns a host, host.newSession() opens an isolated session with `screen`, `app`, `platform`, and engine fixtures, and session.saveState() captures state exportable as AES-256-GCM-encrypted `session-state-1` bytes via SessionState.export({ key }); secrets can be passed to launch({ secrets }) and retrieved via host.secret(name); and the new `@e2e-dev/core/agent` export adds agent support with createAgent(session, { model, cache }) (where `cache` causes every `act` to name its recording with `cacheKey`) and re-exports `defineTool` alongside the rest of `e2e/agent`.

Drive a browser from plain TypeScript outside any test runner — useful for scripting, tooling, or one-off automation where you do not want a full test suite.

typescript

```typescript
import { launch } from '@e2e-dev/core';
import { web } from '@e2e-dev/web/engine';

const host = await launch(web(), { secrets: { AUTH_TOKEN:
process.env.AUTH_TOKEN } });
const session = await host.newSession();
const { app, screen } = session;

await app.open('/dashboard');
const title = await screen.getByRole('heading').textContent();
console.log(title);

const state = await session.saveState();
const encrypted = await state.export({ key: encryptionKey });

await session.close();
```

Drive a mobile app from a plain script without a test runner — open a session, act, capture encrypted state, and tear down.

typescript

```typescript
import { launch } from '@e2e-dev/core';
import { mobile } from '@e2e-dev/mobile/engine';
import { createAgent } from '@e2e-dev/core/agent';

const host = await launch(mobile(), { secrets: { apiKey:
process.env.API_KEY } });
const session = await host.newSession();
const agent = createAgent(session, { model: 'gpt-4o', cache: true
});

await session.app.open('/orders');
await agent.act('add the first item to the cart', { cacheKey:
'add-first-item' });

const state = await session.saveState();
const encrypted = await state.export({ key: process.env.STATE_KEY
});
// encrypted is AES-256-GCM 'session-state-1' bytes

await session.close();
```

Drive a browser from a plain script — no test runner needed — capture encrypted session state for later replay, and add an agent that caches its recordings.

typescript

```typescript
import { launch } from '@e2e-dev/core';
import { web } from '@e2e-dev/web/engine';
import { createAgent } from '@e2e-dev/core/agent';

const host = await launch(web(), { secrets: { apiKey:
process.env.API_KEY } });
const session = await host.newSession();
const agent = createAgent(session, { model: myModel, cache: true
});

await session.app.open('/dashboard');
await agent.act('export the weekly report');

const state = await session.saveState();
const encrypted = await state.export({ key: encryptionKey }); //
AES-256-GCM session-state-1 bytes
```

Drive a browser from a plain script — no test runner, config file, or report — to open a page, act on it, and export encrypted session state.

typescript

```typescript
import { launch } from '@e2e-dev/core';
import { web } from '@e2e-dev/web/engine';
import { createAgent } from '@e2e-dev/core/agent';

const host = await launch(web(), { secrets: { password:
process.env.APP_PASSWORD } });
const session = await host.newSession();
const agent = createAgent(session, { model: 'gpt-4o', cache: true
});

await session.app.open('https://example.com/login');
await agent.act('log in', { cacheKey: 'login-step' });

const state = await session.saveState();
const encrypted = await state.export({ key: process.env.STATE_KEY
});
// encrypted is session-state-1 bytes, safe to store

await host.close();
```

Add an agent to a headless @e2e-dev/core session so act steps are recorded and replayed from cache, cutting model calls on repeated runs.

```typescript
import { launch, createAgent } from '@e2e-dev/core';
import { web } from '@e2e-dev/web/engine';

const host = await launch(web());
const session = await host.newSession();
const agent = createAgent(session, {
  model: myModel,
  cache: { cacheKey: 'checkout-happy-path' },
});

await agent.act('add the first product to the cart and proceed to
checkout');
await session.close();
```

— Full API surface and encryption format named. Names both functions but no further detail. Names API, cache mechanism, and re-export.

`@e2e-dev/web@0.16.1` `@e2e-dev/core@0.24.0` `@e2e-dev/mobile@0.12.8`

02 **Computed CSS style reads via getToolContext** Not checked yet NEW

93

Read a node's computed CSS style from a read-only project tool with getToolContext(options).read({ id }, { style: ['color', 'font-weight'] }), returning allow-listed values secret-redacted under the step's deadline; engines can implement the optional `readStyle` method over the engine protocol for the `computed-style` capability, and a read on an engine without it fails with `POLICY_DENIED`.

Read a node's computed CSS colour and font weight inside a project tool to verify visual styling, without granting write access to the tool.

```typescript
import { getToolContext } from 'e2e';

export const checkBadgeStyle = {
  name: 'check_badge_style',
  async execute(options) {
    const ctx = getToolContext(options);
    const badge = await ctx.read(
      { id: 'status-badge' },
      { style: ['color', 'font-weight'] }
    );
    return badge.style;
  },
};
```

— Full API, capability name, and failure mode given. `@e2e-dev/web@0.16.1`

03 **CI pipeline generation in e2e init** Not checked yet NEW

93

`e2e init` now generates a CI pipeline — a GitHub Actions workflow or a GitLab CI/CD file — with one job per platform; pass `--ci github-actions` or `--ci gitlab` to answer non-interactively, or `--yes` alone to skip; for GitLab it writes alongside an existing `.gitlab-ci.yml` as `.gitlab/e2e.gitlab-ci.yml` and prints the `include:` to add.

— Exact flags, file paths, and output named. `e2e@0.24.0`

04 **Worker-scoped fixtures via isolation: 'worker'** Not checked yet

NEW

90

`defineEngine` now accepts `isolation: 'worker'`, carried in the engine protocol manifest, making the engine's fixtures, `app`, and `screen` worker fixtures; `@e2e-dev/mobile` declares it, so a mobile `beforeAll` can set a permission or open the app. test.extend() now accepts worker fixtures typed as `[fn, { scope: 'worker' }]` with the second type argument (base.extend<{}, { database: Database }>()); `beforeAll` and `afterAll` receive `platform` and worker fixtures, and reading a test fixture there fails with `HOOK_FAILED` naming the fixture.

Share a device across all tests in a worker so a `beforeAll` can set permissions or open the app once, cutting per-test startup cost.

```typescript
import { test as mobileTest } from '@e2e-dev/mobile';

// Worker fixture: set up once per worker, torn down when the
worker ends
const test = mobileTest.extend<{}, { readyApp: void }>({
  readyApp: [
    async ({ app }, use) => {
      await app.open('/login');
      // grant permissions, seed state, etc.
      await use();
    },
    { scope: 'worker' },
  ],
});

test.beforeAll(async ({ platform, readyApp }) => {
  console.log('Running on', platform);
});

test('checkout flow', async ({ screen }) => {
  await screen.getByRole('button', { name: 'Continue' }).tap();
});
```

— Full mechanism, typing, and failure mode described. `@e2e-dev/mobile@0.12.8`

---

05 **Fixture misconfiguration caught at collection time**

Not checked yet

90

BREAKING

Fixture misconfiguration is now caught at collection time, before any engine starts: destructuring the fixture parameter is required — `async (fixtures) => {}`, `async (_, use) => {}`, a rest property (`{ app, ...rest }`), and a bound function (fn.bind()) are a `COLLECTION_ERROR`, and a fixture needing nothing writes `async ({}, use) => {}`. A fixture in test.extend() can now name other fixtures declared in the same test.extend() call in any order; a cycle is a `COLLECTION_ERROR`. A fixture name the target does not have (previously `UNSUPPORTED_CAPABILITY`) is now a `COLLECTION_ERROR`; a `beforeAll`, `afterAll`, or worker fixture naming a test fixture (previously `HOOK_FAILED`) is now a `COLLECTION_ERROR`; and a fixture named like a core fixture (`agent`, `app`, `screen`, `platform`, `session`) or like one already defined by an earlier `extend` call (previously `TEST_SETUP_FAILED`) is now a `COLLECTION_ERROR`.

— Enumerates every error condition and code precisely. `e2e@0.24.0` `@e2e-dev/core@0.24.0`

---

06 **lightpanda() loads stylesheets by default** Not checked yet

86

BREAKING

The lightpanda() integration now defaults `loadResources` to `['stylesheet']` instead of `[]`, loading linked stylesheets so externally-hidden elements are correctly treated as invisible in visible queries and agent steps without any config change; set `loadResources: []` in your lightpanda() config to opt out and restore the previous no-fetch behaviour.

Restore the old no-fetch behaviour when your test environment cannot reach external stylesheets.

typescript

```
lightpanda({ loadResources: [] })
```

— Exact config key, default values, and opt-out shown. `@e2e-dev/lightpanda@0.0.3`

07 Fixture override in test.extend()

Not checked yet NEW

86

Override built-in fixtures in test.extend(), including core ones like `device`, by naming the fixture's own name in the extender — base.extend({ device: async ({ device }, use) => { … } }) — keeping the original scope and `auto` setting; a mismatch is a `COLLECTION_ERROR`.

Override the built-in `device` fixture in a mobile test suite to deny camera permission for every test in that suite.

typescript

```
import { test as base } from 'e2e';

const test = base.extend({
  device: async ({ device }, use) => {
    await device.setPermission('camera', 'deny');
    await use(device);
  }
});

test('upload falls back to file picker when camera is denied',
async ({ app, screen }) => {
  await app.open('/upload');
  // camera permission already denied for this test
});
```

— Exact API shape and error condition named. `e2e@0.24.0`

08 Deep-link navigation via app.linkBase

Not checked yet NEW

83

Open deep links inside a pinned app by setting `app.linkBase` on device targets and calling app.open('/orders/42'), which routes through the new optional engine hook `session.openLink` (protocol method `session/openLink`); `@e2e-dev/mobile` implements it.

— Names config field, method, and protocol hook. `@e2e-dev/web@0.16.1`

09 **HTML reporter enhancements** Not checked yet NEW

83

The HTML report now lets you filter results by target on multi-target runs, with target tabs appearing above the status tabs driving the counts, test tree, and header; sessions a test opens with launch() appear in the report as named steps (`launch-1`, `launch-2`, …) with their own video playback, model usage counted with the test's, and an `actor` field on steps and video artifacts (also carried on `progress.identity` for reporters); and the new `e2e report` command opens the `@e2e-dev/html-reporter` HTML report in the default browser.

— Describes UI behaviour, no screenshot or path. Names fields and step naming scheme precisely. One command, no flags or output detail.

`@e2e-dev/html-reporter@0.0.2` `e2e@0.24.0`

10 **On-demand fixture initialization with auto: true** Not checked yet

83

BREAKING

test.extend() fixtures are now set up on demand (as in Playwright and Vitest): only fixtures named in a test body, `beforeEach`, `afterEach`, or suite hook are initialised. Mark side-effect-only fixtures `auto: true` with `[fn, { auto: true }]` (per-test) or `[fn, { scope: 'worker', auto: true }]` (per-worker) to keep them running unconditionally.

Keep a side-effect-only fixture (e.g. a data seeder) running for every test even though no test names it directly.

typescript

```typescript
import { test as base } from 'e2e';

const test = base.extend({
  seedDatabase: [async ({}, use) => {
    await seedTestData();
    await use(undefined);
    await cleanupTestData();
  }, { auto: true }],
});
```

— Exact syntax for both scopes given. `e2e@0.24.0` `@e2e-dev/core@0.24.0`

11 **moli() browser provider for web engine** Not checked yet NEW

80

A new moli() browser provider for the web engine, used via web({ browser: moli() }), gives each test worker a local Moli CDP server with layout and resources enabled, automatic port selection, startup cancellation, and process cleanup when the lease ends.

Run your e2e web tests against a local Moli browser, one CDP server per worker with automatic port selection and cleanup.

typescript

```
import { web } from '@e2e-dev/web';
import { moli } from '@e2e-dev/moli';

export default {
  engine: web({ browser: moli() }),
};
```

— Full usage shown; no limits or numbers given. `@e2e-dev/moli@0.0.1`

12 **Engine module exports without the test runner** Not checked yet

NEW

78

`@e2e-dev/web/engine` and `@e2e-dev/mobile/engine` now export engine factories and their types without loading the test runner; defineEngine<Fixtures>() and `EngineHandle<Fixtures>` carry engine-contributed fixtures as a type parameter, and `defineEngine` refuses a fixture named `close`, `saveState`, or `setDefaultTimeout`. `@e2e-dev/core/engine` (and `e2e/engine`) now exports `isRefusedLinkScheme`, the list of schemes app links are refused by.

— Names modules, functions, and refused fixture names. `@e2e-dev/web@0.16.1`

13 **agentDevicePlugin for mobile provider registration**

Not checked yet NEW

69

agentDevicePlugin(entry) from `@e2e-dev/mobile/lease` registers provider plugins and checks their versions against the engine's agent-device, for use in a lease daemon setup.

Register your provider plugin against the engine's agent-device version inside a lease daemon setup, so mismatched versions are caught at startup.

typescript

```
import { agentDevicePlugin } from '@e2e-dev/mobile/lease';

agentDevicePlugin(entry);
```

Register a provider plugin with the lease daemon so its agent-device version is validated before tests run.

```typescript
import { agentDevicePlugin } from '@e2e-dev/mobile/lease';

// In your provider plugin's entry point:
export default agentDevicePlugin(import.meta.url);
```

— Names plugin API and purpose, lacks config details. `@e2e-dev/testmu@0.0.6`

14 **Provider download naming and lease release batching**

Not checked yet IMPROVED 65

A browser provider's `downloads.read` now receives the site's suggested file name as `context.suggestedFilename` for services that store downloads by name; `releaseEach` now releases every lease a provider's `sweep` found at once, sorting results into released and failed so a failed release no longer skips the others.

— Names fields and sweep/release mechanism. `@e2e-dev/web@0.16.1`

15 **Dot reporter for compact CI output** Not checked yet NEW 60

Print one character per test result by setting `reporters: ['dot']` in config or passing `--reporter dot` on the command line.

Run tests and emit a compact dot-per-result output for CI pipelines where full progress lines are too noisy.

```
$ npx e2e run --reporter dot
```

— Exact config key and CLI flag given. `e2e@0.24.0`

16 **e2e mcp drops run_tests tool** Not checked yet BREAKING 60

`e2e mcp` no longer serves the `run_tests` tool; run tests with `npx e2e run` instead. The MCP server retains `open_session`, `tools`, `call`, and `close_session`.

— Names removed and retained tools exactly. `e2e@0.24.0`

THINNER COVERAGE BELOW

17 **@e2e-dev/core as explicit Yarn peer dependency** Not checked yet

54

BREAKING

A Yarn project must now list `@e2e-dev/core` as an explicit peer dependency alongside the engine package; `e2e init` and `e2e upgrade` add it automatically for npm, pnpm, and Bun, but existing Yarn projects require a manual addition.

— Names package and affected package managers. `@e2e-dev/web@0.16.1` `@e2e-dev/mobile@0.12.8` `@e2e-dev/core@0.24.0`

---

18 **mobile() rejects invalid snapshot/videoTouches values**

Not checked yet IMPROVED

48

mobile() now rejects a `snapshot` value other than `'full'` or `'interactive'` and a non-boolean `videoTouches` value instead of silently using the default.

— States exact rejected inputs, no other detail. `@e2e-dev/mobile@0.12.8`

---

19 **iOS device.back() navigation bar handling** Not checked yet

IMPROVED

47

iOS device.back() now presses the navigation bar's back button when one is present, with bounded idle waits for keyboard dismiss and enter actions.

— Describes new behaviour and wait conditions concretely. `@e2e-dev/testmu@0.0.6`

---

20 **Lease device selector allocates directly** Not checked yet

BREAKING

32

A lease's `device` selector now allocates the device directly instead of passing it to the worker's client, since the lease scope already picks the device.

— One-line behavioural note, no mechanism detail. `@e2e-dev/testmu@0.0.6` `@e2e-dev/mobile@0.12.8`

---

21 **SuiteFixtures and SuiteHookFn deprecated** Not checked yet

DEPRECATED

28

`SuiteFixtures` and `SuiteHookFn` are deprecated aliases of `WorkerFixtures` and `TestHookFn<WorkerFixtures>` respectively.

— Bare rename note, no migration guidance. `@e2e-dev/mobile@0.12.8`

NEW INTEGRATION Revyl hosted iOS simulators and Android emulators integration

└──▷ ALSO FROM THESE RELEASES

Pin an assertion message so failures name the failing UI context instead of showing a raw matcher error.

```typescript
import { test, expect } from 'e2e';

test('cart is populated after adding an item', async ({ app, screen })
=> {
  await app.open('/shop');
  const badge = screen.getByRole('status', { name: 'cart count' });
  await expect(badge, 'cart badge should show 1 after add-to-
cart').toContainText('1');
});
```

└──▷ **BREAKING ON UPGRADE**

! The `loadResources` default in lightpanda() changed from `[]` to `['stylesheet']`; setups relying on no external resources being fetched must now explicitly pass `loadResources: []`.

! A Yarn project must now list `@e2e-dev/core` as an explicit peer dependency alongside the engine package; `e2e init` and `e2e upgrade` add it automatically, but existing Yarn projects require a manual addition.

! Yarn projects must add `@e2e-dev/core` explicitly beside the engine package; npm, pnpm, and Bun install it automatically as a peer.

! `SuiteFixtures` and `SuiteHookFn` are deprecated aliases of `WorkerFixtures` and `TestHookFn<WorkerFixtures>` respectively.

! A lease's `device` selector no longer passes through to the worker's client; it is consumed by the lease scope to allocate the device.

! test.extend() fixtures are now set up on demand: a fixture that nothing names no longer runs. Add `auto: true` — `[fn, { auto: true }]` for per-test or `[fn, { scope: 'worker', auto: true }]` for per-worker — to any fixture that runs for its side effects. Destructuring the first parameter is now required; `async (fixtures) => {}`, `async (_, use) => {}`, a rest property, and a bound function are a `COLLECTION_ERROR`.

! `e2e mcp` no longer serves the `run_tests` tool. Run tests with `npx e2e run` instead; the MCP server retains `open_session`, `tools`, `call`, and `close_session`.

! test.extend() fixtures are now set up on demand: a fixture that exists only for its side effect and is not named in any test, hook, or other fixture no longer runs. Add `auto: true` (or `{ scope: 'worker', auto: true }`) to restore the old behaviour.

! Destructuring the fixture parameter is now required: `async (fixtures) => {}`, `async (_, use) => {}`, a rest property (`{ app, ...rest }`), and a bound function (fn.bind()) are now a `COLLECTION_ERROR`; a fixture that needs nothing writes `async ({}, use) => {}`.

! A fixture name the target does not have is now a `COLLECTION_ERROR` at collection time (previously `UNSUPPORTED_CAPABILITY` during the test).

! A `beforeAll`, `afterAll`, or worker fixture that names a test fixture is now a `COLLECTION_ERROR` at collection time (previously `HOOK_FAILED` at runtime).

! A fixture named like a core fixture (`agent`, `app`, `screen`, `platform`, `session`) or like one already defined by an earlier `extend` call is now a `COLLECTION_ERROR` (previously

`TEST_SETUP_FAILED` in the test).

! Yarn projects must add `@e2e-dev/core` as an explicit dependency beside the engine package; `e2e init` and `e2e upgrade` add it automatically, but existing Yarn projects need a manual update.

`@e2e-dev/revyl@0.0.1` NOTES ↗ **LATEST** STABLE

└──▷ **GET THIS VERSION** ▾

```
$ git clone --branch @e2e-dev/revyl@0.0.1 https://github.com/e2e-inc/e2e.git
# already have the repo? check out this version:
$ git checkout @e2e-dev/revyl@0.0.1
```

WAS THIS USEFUL?  

agentbro

Feedback ⓘ v3.2.0 NOTES ↗ RANK

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

> AgentBro v3.2.0 adds two new desktop agent integrations — DeepSeek Harness and QwenWork CN — surfaced live in the macOS Island, extends Agent Management to detect and manage their Skills, and documents Homebrew/DMG install options.

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

01 **DeepSeek Harness integration in macOS Island** Not checked yet

NEW 85

AgentBro now reports DeepSeek Harness turn activity directly in the macOS Island, with Allow once and Deny approval actions available once a Hook is installed for it via Agent Management. To enable it: open DeepSeek Harness once and quit it, install its Hook from Agent Management, then reopen it so turn activity and approval prompts appear in the Island.

Connect DeepSeek Harness to the macOS Island so AgentBro can surface its turn activity and approval prompts.

📍 1. Open DeepSeek Harness once, then quit it. 2. In AgentBro, go to Agent Management, find DeepSeek Harness, and install its Hook. 3. Reopen DeepSeek Harness — turn activity now appears in the macOS Island with Allow once and Deny options.

— Names the Hook mechanism and exact approval actions, plus a runnable setup sequence.

v3.2.0

02 **Homebrew and DMG install for AgentBro** Not checked yet IMPROVED

70

AgentBro can be installed on macOS via Homebrew with `brew tap shirenchuang/tap && brew install --cask agentbro`, or by downloading `AgentBro_3.2.0_universal.dmg`.

— Exact install command and file name make it immediately runnable. v3.2.0

03 **Agent Management support for DeepSeek Harness and QwenWork CN**

Not checked yet IMPROVED 65

Agent Management now detects both DeepSeek Harness and QwenWork CN as separate desktop entries and manages their local directory-format Skills, with DeepSeek Harness

also surfacing inherited Skills from the default shared directory.

— Describes skill detection and inheritance but only points to a UI area. v3.2.0

04 **QwenWork CN task integration in macOS Island** Not checked yet

NEW

60

AgentBro adds a QwenWork CN integration that reads local task phases and shows queued, running, finished, and waiting states live in the macOS Island.

— States mechanism and states shown but gives no setup steps. v3.2.0

WAS THIS USEFUL?  

◆ Model Training & Fine-Tuning

fizgig

Feedback ⓘ v9.8.1 NOTES ↗ RANK

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

> Fizgig's v9.8.1 release adds a dedicated Studio render queue and auto-saving outputs folder, MiniMax H3 first/last-frame preview anchoring, single-pane rendering for faster comparisons, epoch-preserving run deletion, and a default-off INT8 fast inference switch for Krea 2 on AMD cards.

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

01 **Dedicated Studio Outputs folder with auto-save** Not checked yet

NEW 80

Finished Studio renders are now saved automatically into a dedicated `Studio_Outputs` folder (with `images`, `videos`, and `songs` subfolders), configurable in `Preferences > Other folders`, with timestamps in every filename so nothing is overwritten. A new Save every render tick beside the Save buttons lets you auto-save every finished render, off by default and stored per model.

— Names config path and exact toggle; could add storage limits. v9.8.1

02 **First/last-frame anchoring for MiniMax H3 previews**

Not checked yet NEW 75

Set First frame and Last frame pictures on the Samples tab for MiniMax H3 to anchor clip previews to specific images, cropped to the preview's shape; on the Reference (ref2va) Training Base the same slots become Reference 1 and Reference 2, named `<Picture 1>` and `<Picture 2>` in the preview prompt.

— Names exact UI slots and prompt tokens; clear mechanism. v9.8.1

03 **Krea 2 AMD inference defaults to fp8** Not checked yet IMPROVED

75

Krea 2 on AMD cards now starts with INT8 fast inference off by default, using the stored fp8 file instead for roughly 2× faster previews at a quarter of the RAM (tested at 2048 × 2048 on an RX 9070 XT); NVIDIA behaviour is unchanged.

— Concrete benchmark and hardware named, but no override flag given. v9.8.1

04 **Epoch-preserving deletion of training runs** Not checked yet NEW

70

Delete a training run while keeping its newest epochs using the new `Delete, keep the newest 5 epochs` or `Delete, keep all epochs` options on a run's card.

— Exact option labels given, directly usable from run card. v9.8.1

05 **Render queue in Studio** Not checked yet NEW 65

Studio now has its own render queue: use `Add to queue` (or ↻ for a new seed) to line up jobs behind the current render, with each job saved into the Studio outputs folder and links shown in the queue bar above the preview.

— Clear UI controls named but no limits on queue size given. v9.8.1

06 **Single-pane rendering to speed previews** Not checked yet NEW 60

Hide either the Baseline or Tweaked pane in Studio using the arrows between them so only the visible pane renders — hiding the Baseline of a MiniMax H3 clip halves each render time.

— Concrete performance claim but limited mechanism detail. v9.8.1

THINNER COVERAGE BELOW

07 **Full clip previews with audio for MiniMax H3/Vimoo**

Not checked yet IMPROVED 45

A previews helper rendering MiniMax H3 or Vimoo previews now returns the full clip and its sound rather than only the middle-frame still.

— Describes behaviour change but no UI path or trigger given. v9.8.1

08 **Persistent picture folder memory in Studio** Not checked yet

IMPROVED 40

Studio now remembers the last folder you picked pictures from across restarts for the Reference picker, First/Last frame, Reference slots, and metrics reference.

— Simple quality-of-life note, minimal mechanism or scope. v9.8.1

WAS THIS USEFUL? 👍 👎

Hugging Face PEFT

Feedback ⓘ v0.22.0 NOTES ↗ RANK

PEFT implements parameter-efficient fine-tuning methods including LoRA, QLoRA, prefix tuning, and adapters for large pretrained models.

> PEFT v0.22.0 ships a new PaCA fine-tuning method, Astra and NoRA LoRA initializations, a model health-check API, wider LoRA-conversion support, and FSDP2/tensor-parallel improvements, alongside several breaking changes including a stricter `modules_to_save` match, a corrected OFT Conv2d forward pass, and a raised minimum Python version.

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

01 **New LoRA initialization schemes** Not checked yet NEW 95

Adds two new LoRA initialization schemes: `init_lora_weights="astra"`, paired with `preprocess_astra`, initializes LoRA in the lowest-variance directions of each target layer's output-activation covariance computed from a calibration dataloader for a task-aware starting point; and `init_lora_weights="nora"` (used with `lora_alpha` set equal to `r`) normalizes each column of the LoRA A matrix along the rank axis for faster and better convergence than standard LoRA initialization.

Use Astra initialization to give a new LoRA adapter a task-aware starting point before fine-tuning, using a small calibration dataset.

python

```python
from peft import LoraConfig, get_peft_model
from peft import preprocess_astra

peft_config = LoraConfig(
    r=16,
    lora_alpha=32,
    init_lora_weights="astra",
    target_modules=["q_proj", "v_proj"],
)
preprocess_astra(model, peft_config, calibration_dataloader)
model = get_peft_model(model, peft_config)
```

Use NoRA initialization for LoRA to normalize A-matrix columns along the rank axis, which the paper shows converges faster than standard initialization.

```python
from peft import LoraConfig, get_peft_model

peft_config = LoraConfig(
    r=16,
    lora_alpha=16,
    init_lora_weights="nora",
    target_modules=["q_proj", "v_proj"],
)
model = get_peft_model(model, peft_config)
```

— Mechanism, config, and runnable example all given. Named flag, pairing rule, and runnable example given.

v0.22.0

02 **Model health-check API** Not checked yet NEW 85

Call model.print_healthcheck() or model.healthcheck() on any PEFT model to get a summary of adapter state, dtypes, device placement, and parameter counts plus flagged issues; this replaces model.print_trainable_parameters().

Run a health check on a LoRA-wrapped model to inspect adapter state, dtypes, device placement, and trainable parameter counts in one call.

```python
model.print_healthcheck()
```

— Names exact replaced and new methods with a runnable example. v0.22.0

03 **Exact-match requirement for modules_to_save** Not checked yet 85

BREAKING

`modules_to_save` now requires the whole sub-module name to match exactly rather than matching by suffix — `"classifier"` no longer matches `"other_classifier"`. Update existing configs or `adapter_config.json` to list all intended module names explicitly.

— Exact before/after behavior and migration note given. v0.22.0

04 **Adapter-key mismatch strictness control** Not checked yet NEW 78

PeftModel.from_pretrained() and load_adapter() gain a `strict_adapter_check` argument: set to `"error"` to fail on an adapter-key mismatch or `"ignore"` to suppress warnings.

— Exact parameter and values named, no code example. v0.22.0

05 **Activation checkpointing default changed for kbit training prep**

Not checked yet IMPROVED 78

prepare_model_for_kbit_training() now uses the Transformers non-reentrant checkpointing default; pass `activation_checkpointing_kwargs={"use_reentrant": True}` to opt back into reentrant checkpointing.

— Names exact kwarg and new default, no runnable snippet. v0.22.0

06 **FSDP2/HSDP and tensor-parallel LoRA support** Not checked yet

IMPROVED 78

Save FSDP2 or HSDP adapter shards by calling save_pretrained() on every participating rank, passing `is_main_process=True` only on the rank that writes files. LoRA now also supports the `embedding_colwise` tensor-parallelism plan, with replicated weights synchronized across TP ranks at initialization.

— Both distributed-training surfaces named exactly, no code sample. v0.22.0

07 **create_dummy_lora_config helper for adapter hotswapping**

Not checked yet NEW 73

`peft.helpers.create_dummy_lora_config` builds a dummy starting adapter with enough rank and layer coverage to accommodate a set of compatible LoRA adapters for hotswapping across different target layers.

— Exact function named, purpose explained, no usage snippet. v0.22.0

08 **LoRA conversion support for OFT, BOFT, DEFT, MiSS** Not checked yet

IMPROVED 68

LoRA conversion now supports OFT and BOFT linear adapters, DEFT, and standard and mini MiSS modes, letting more PEFT checkpoints be converted to LoRA for frameworks that only support LoRA.

— Lists all supported adapter types but no conversion command shown. v0.22.0

THINNER COVERAGE BELOW

09 **Corrected forward pass in OFT Conv2d layers** Not checked yet

55

BREAKING

Existing OFT checkpoints that target `Conv2d` layers may produce different outputs when used without merging, due to a corrected forward pass in `Conv2d` OFT layers.

— Names affected layer type and impact, no migration steps. v0.22.0

10 **PaCA fine-tuning method** Not checked yet NEW 53

PaCA (Partial Connection Adaptation) is a new PEFT method that fine-tunes randomly selected input columns of pretrained weights without adding adapter layers, using less memory and training time than LoRA.

— Mechanism described but no API name or usage example. v0.22.0

11 **Minimum Python version raised to 3.11** Not checked yet BREAKING 50

PEFT now requires Python 3.11 or newer; Python 3.10 is no longer supported.

— Clear version numbers but no further detail. v0.22.0

└──▷ **BREAKING ON UPGRADE**

! `modules_to_save` matching changed from suffix-only to exact sub-module name: `modules_to_save=["classifier"]` no longer matches `model.other_classifier`. Update existing configs or `adapter_config.json` to list all intended module names explicitly.

! PEFT now requires Python 3.11 or newer; Python 3.10 is no longer supported.

! Existing OFT checkpoints that target `Conv2d` layers may produce different outputs when used without merging, due to a corrected forward pass in `Conv2d` OFT layers.

WAS THIS USEFUL?  

◆ AI App Builders

Lovable

Feedback ⓘ Nov 18, 2026 (2026-11-19) BLOG ↗ RANK

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

> Lovable added a full ClickUp integration, a workspace-wide safeguard that blocks publishing when critical security findings are open, and smaller updates to database sizing, mobile commenting, and OAuth verification — while removing public profile pages.

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

01 **Critical findings can block publishing everywhere** Not checked yet

NEW 90

When a workspace enables Block publishing with critical issues in Settings > Security, critical findings from the Deep scan, dependency audit, and AI pentests now block publishing from project chat, Chats, Lovable in Slack, the Lovable API, and the Lovable MCP server.

Ensure no one can ship a build with unresolved critical findings, no matter which publish path they use — including Slack, the API, and MCP.

📍 In the workspace, go to Settings › Security and enable Block publishing with critical issues. Once on, the Deep scan, dependency audit, and AI pentest findings all gate publishing from project chat, Chats, Lovable in Slack, the Lovable API, and the Lovable MCP server.

— Names the exact setting, scan sources, and every gated publish path.

Nov 18, 2026 (2026-11-19)

02 **ClickUp integration for tasks, docs, and time** Not checked yet

NEW

78

Connect a Lovable app to ClickUp to list, create, and update tasks, comment on tasks, browse spaces, folders, and lists, read and write docs, and read and record time entries. Add it from Connectors in the project or ask Lovable in the project chat to connect it.

Add a ClickUp integration to a Lovable project so the app can read and write tasks, docs, and time entries.

> 📍 In the project, go to Connectors and select ClickUp, then authorise the connection. Alternatively, open the project chat and ask Lovable to connect ClickUp.

— Names every ClickUp capability and two setup paths. Nov 18, 2026 (2026-11-19)

THINNER COVERAGE BELOW

03 **Built-in database upgrade from Tiny to Mini** Not checked yet

IMPROVED

50

Upgrade a built-in database from Tiny to Mini directly in the publish dialog when deploying changes, or ask Lovable to recommend a size based on expected traffic.

— Names the two tiers and where to change them, no further mechanism.

Nov 18, 2026 (2026-11-19)

04 **Richer OAuth client verification requests** Not checked yet

IMPROVED

40

Workspace admins and owners on Business and Enterprise plans can now describe their app — including its homepage and why it needs each sensitive scope — when requesting verification of an OAuth client.

— Names the plans and fields added but not the request flow. Nov 18, 2026 (2026-11-19)

05 **Removal of public profile pages** Not checked yet BREAKING

40

Public profile pages for people and workspaces have been removed, along with follows and the contribution heatmap; links to a profile now open the Lovable home page instead.

— Clear before/after but no migration guidance given. Nov 18, 2026 (2026-11-19)

06 **Mobile commenting in the project editor** Not checked yet NEW

35

Read, add, and reply to comments on a phone in the project editor, from either a mobile browser or the Lovable mobile app.

— States the capability but no further detail on scope or limits.

Nov 18, 2026 (2026-11-19)

NEW INTEGRATION Adyen payment connector

NEW INTEGRATION Airwallex payment connector

└──▷ ALSO FROM THESE RELEASES

└──▷ BREAKING ON UPGRADE

! Public profile pages for people and workspaces have been removed, along with follows and the contribution heatmap — links to a profile now open the Lovable home page.

Nov 19, 2026 (2026-11-19) BLOG ↗ LATEST STABLE

WAS THIS USEFUL? 👍 👎

AGENT

◆ AI Agent Frameworks

Arcade

Feedback ⓘ updated · Nov 19, 2026 NOTES ↗ RANK

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

> Arcade's latest snapshot adds connected-account profiles, new Datadog and Snipe-IT toolkits, stronger API key and organization controls, and a new client-identity/OAuth model for MCP servers that comes with several breaking changes operators must address before upgrading.

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

01 Client-identity and OAuth resource identity for MCP servers

Not checked yet BREAKING 90

Arcade now issues client-identity documents and a public OAuth resource identifier for remote and deployed MCP servers, showing end-users a confirmation page on deployed-server authorization. OAuth negotiation is now refused for servers that have not confirmed PKCE support — operators with remote servers relying on silent PKCE support must enable Assume PKCE support after upgrading; non-internet-reachable Coordinator operators should check the `engine.clientIdMetadataDocuments` setting before upgrading; and custom headers on an Arcade-hosted MCP server can no longer be edited to override Arcade's own injected authentication headers.

— Names mechanism, config key, and migration steps for three related breaking changes

snapshot-20261119

02 Disable automatic database migrations on Helm upgrade

Not checked yet NEW 75

Setting `migrations: false` under `engine.database` and `coordinator.database` in `values.yaml` prevents Helm upgrade from automatically running database migrations.

Prevent helm upgrade from automatically running database migrations in a controlled environment.

```yaml
engine:
  database:
    migrations: false
coordinator:
  database:
    migrations: false
```

— Includes a runnable config snippet with exact keys snapshot-20261119

03 **Connected-account profiles per integration** Not checked yet NEW

60

Connected-account profiles let end-users keep multiple connected accounts per integration and choose which one a tool call uses; the feature is opt-in per organization.

— Describes mechanism and opt-in scope but no UI/API path given snapshot-20261119

04 **Worker toolkit secrets via environment variable** Not checked yet

NEW

60

Worker toolkits can now receive secrets through the `ARCADE_WORKER_SECRETS` environment variable.

— Names exact env var but lacks usage example snapshot-20261119

THINNER COVERAGE BELOW

05 **Removal of account gateway binding type** Not checked yet BREAKING

55

The `account` gateway binding type is removed; Arcade refuses to apply the migration if any row still uses it.

— Names the removed type and migration guard but no remediation steps snapshot-20261119

06 **Kerberos/GSSAPI package requirement before upgrade**

Not checked yet

55

BREAKING

Servers needing Kerberos/GSSAPI support must have `apt_packages` set before upgrading.

— Names the exact config key required pre-upgrade snapshot-20261119

07 **Datadog and Snipe-IT toolkits** Not checked yet NEW

50

Arcade adds a Datadog toolkit to query metrics and build charts, and a Snipe-IT toolkit for IT asset management.

— Names two toolkits but gives no detail on calls or capabilities snapshot-20261119

08 **Organization suspend and unsuspend control** Not checked yet NEW

50

Operators can suspend or unsuspend an organization, which removes the suspended organization's deployed workers in one command.

— Describes effect but not the exact command or endpoint snapshot-20261119

09 **Configurable sign-up domain blocklist** Not checked yet NEW

50

Sign-ups can be blocked by domain at runtime via a configurable domain blocklist, with built-in refusal of common disposable-email domains.

— Explains behavior but omits the config surface used to set it snapshot-20261119

10 **Tool-execution record filtering by key and gateway**

Not checked yet NEW

45

Tool-execution records and queries can now be filtered by project API key and gateway identifier.

— States the filter fields but no query syntax or endpoint snapshot-20261119

└──▷ **BREAKING ON UPGRADE**

! The `account` gateway binding type is removed; Arcade refuses to apply the migration if any row still uses it.

! OAuth negotiation is now refused for servers that have not confirmed PKCE support; operators with remote servers that rely on silent PKCE support must enable `Assume PKCE support` after upgrading.

! List API endpoints for Secrets, OAuth Providers, and Gateways now reject unrecognized filter/sort values with a bad-request error; scripts sending undocumented filter/sort values will break.

! Servers needing Kerberos/GSSAPI support must have `apt_packages` set before upgrading.

! Non-internet-reachable Coordinator operators should check the `engine.clientIdMetadataDocuments` setting before upgrading.

! Custom headers on an Arcade-hosted MCP server can no longer be edited to override Arcade's own injected authentication headers.

WAS THIS USEFUL? 👍 👎

trigger.dev

Feedback ⓘ v4.8.7 NOTES ↗ RANK

Trigger.dev – build and deploy durable AI agents and workflows

> trigger.dev's v4.8.7 release extends concurrency key configuration flexibility to sessions and chat agents.

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

01 **Object-form concurrencyKey for sessions and chat.agent**

Not checked yet **IMPROVED** 65

Sessions and `chat.agent` in `triggerConfig` now accept the object form of `concurrencyKey`, giving each concurrency limit its own key — matching the existing behaviour already available on task.trigger().

– Names exact config field and surfaces but no example shown v4.8.7

WAS THIS USEFUL? 👍 👎

◆ MCP Servers & Tooling

mcp-server

Feedback ⓘ v2.6.11 NOTES ↗ RANK

BrowserStack's Official MCP Server

> mcp-server adds scoped accessibility scanning and speeds up startup by caching the capability index across requests.

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

01 **Scoped accessibility scanning with scanScope** Not checked yet

NEW

60

Accessibility scans can now target a specific region of a page instead of scanning the full page, by setting `scanScope` in accessibility scans. This reduces noise from irrelevant regions when only a section needs checking.

Target an accessibility scan at a specific section of the page rather than scanning everything, reducing noise from irrelevant regions.

```text
scanScope
```

– Names config key but lacks value format or examples v2.6.11

THINNER COVERAGE BELOW

02 **Capability index loaded once per process** Not checked yet IMPROVED

35

The capability index is now loaded once per process rather than on every request, cutting repeated startup overhead.

– Explains mechanism but no metrics or user action v2.6.11

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ Inference Providers & GPU Clouds

CoreWeave

Feedback ⓘ updated · Nov 19, 2026 NOTES ↗ RANK

CoreWeave is an AI cloud platform offering GPU compute, managed Kubernetes and storage for training and inference.

> CoreWeave introduced scheduling priority control for Node Pools, letting organizations order capacity allocation across pools with matching characteristics.

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

01 **Node Pool scheduling priority** Not checked yet NEW 85

Set `spec.schedulingPriority.value` on a Node Pool spec to control the relative order in which Node Pools receive available capacity. This is applied org-wide across pools in different clusters when instance type, Availability Zone, network fabric, and compute class match.

Prioritize a high-priority Node Pool so it receives capacity before lower-priority pools sharing the same instance type and Availability Zone.

yaml

```yaml
spec:
  schedulingPriority:
    value: 100
```

— Named config field with exact behavior and matching criteria described.

snapshot-20261119

WAS THIS USEFUL? 👍 👎

◆ AI / LLM Gateways

Mozilla AI Otari

Feedback ⓘ v0.54.0 NOTES ↗ RANK

Mozilla AI Otari is an open-source AI gateway you run yourself. One endpoint for 40+ providers, with virtual keys, budgets, and usage tracking.

> Mozilla AI Otari's v0.54.0 release opens OAuth connections to unregistered sites, surfaces per-provider performance data, adds country-based access gating, and reworks organization invitation handling as a breaking change.

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

01 **OAuth connections for unregistered sites** Not checked yet NEW 45

Otari now allows unregistered sites to connect via OAuth apps, subject to a key limit and expiry.

– Mechanism named (key limit, expiry) but no exact config or flag given. v0.54.0

02 **Provider latency and throughput on model catalog page**

Not checked yet NEW 45

Otari now shows each provider's latency and throughput on the model catalog page.

– Names exact UI location but no metrics detail or thresholds. v0.54.0

03 **Organization invitation accept/decline actions** Not checked yet 45

BREAKING

Otari adds accept and decline actions for organization invitations; this change is marked BREAKING because invitation handling was altered in #3409.

– Names the PR but no migration steps or API detail. v0.54.0

04 **Country-based provider gating** Not checked yet NEW 35

Otari now checks an account's country in the provider country gate before allowing access.

– Behavior described but no config key or affected providers named. v0.54.0

└──▷ **BREAKING ON UPGRADE**

! Organization invitation handling is marked BREAKING: accepting and declining invitations changed in #3409.

WAS THIS USEFUL? 👍 👎

◆ ML Platforms & SDKs

MotleyAI SLayer

Feedback ⓘ v1.9.0 NOTES ↗ RANK

SLayer is an embeddable semantic layer that lets AI agents and people query warehouse data through shared metric definitions.

> SLayer v1.9.0 adds full OpenTelemetry observability — tracing, metrics and log export to any OTLP backend — plus clearer reporting when ingest runs only update key metadata.

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

01 **OpenTelemetry tracing, metrics and logs export** Not checked yet

NEW 95

SLayer can now export traces, metrics and logs to any OTLP backend. Install the `otel` extra with `pip install 'motley-slayer[otel]'`, set `SLAYER_OTEL=on`, and point `OTEL_EXPORTER_OTLP_ENDPOINT` at your collector; it works with the OpenTelemetry Collector, SigNoz, Grafana, OpenObserve, Honeycomb, and any OTLP/HTTP backend. Setting `SLAYER_OTEL_CAPTURE_CONTENT=on` additionally captures SQL text, filter values, row data and error messages in traces. You can instrument SLayer directly inside your own Python code by calling slayer.otel.instrument() instead of relying on the `opentelemetry-instrument` wrapper, and the new `slayer otel status` and `slayer otel test` CLI commands check your configuration and confirm the backend accepts data.

Send per-request spans and duration metrics from a SLayer CLI session to a local OpenTelemetry Collector.

```
$ pip install 'motley-slayer[otel]'
export SLAYER_OTEL=on
export OTEL_EXPORTER_OTLP_ENDPOINT=http://localhost:4318
slayer query --source-model orders --measures 'sum(order_total)'
```

Instrument SLayer inside a Python application so traces are attached to your existing OpenTelemetry providers.

```python
import slayer.otel

slayer.otel.instrument()

# subsequent SLayer calls are now traced
from slayer import SLayer
sl = SLayer()
result = sl.query(source_model='orders', measures=
['sum(order_total)'])
```

Verify your OTLP backend is reachable and inspect what SLayer will export before running in production.

```
$ slayer otel status
slayer otel test
```

— Names every flag, env var and command with runnable examples v1.9.0

THINNER COVERAGE BELOW

02 **Ingest run reporting for metadata-only fills** Not checked yet

IMPROVED 35

Ingest runs that only fill in key metadata now report those fills as `+key metadata` instead of silently appearing as no-op re-ingests.

— Names the new label but gives no usage example v1.9.0

WAS THIS USEFUL?  

Palantir AIP

Feedback ⓘ Build Insight workbooks with AIP

NOTES ↗ RANK

Palantir's Artificial Intelligence Platform (AIP) connects AI with your data and operations, with builder tools for production AI-powered workflows, agents, and functions on top of the Ontology.

> AIP introduces AIP Insight, a copilot embedded in Insight that turns natural-language questions into editable analysis workbooks.

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

01 **AIP Insight copilot for workbook analysis** Not checked yet NEW

70

AIP Insight is a copilot built into Insight that converts natural-language questions about your Ontology into analysis workbooks — it finds relevant data, constructs analysis steps, and shows results alongside the conversation. It's available to all users with AIP enabled, and can be started by selecting Start an analysis with AIP on the Insight homepage or AIP in the analysis path ribbon inside any workbook; every AI-generated analysis step can be inspected and edited directly within the workbook, and the conversation continued to refine results using existing Insight tools.

Start an AI-driven workbook analysis from the Insight homepage when you have a question about your Ontology data but are not sure where to begin.

📍 1. Open the Insight homepage. 2. Select Start an analysis with AIP. 3. Describe your question in natural language (e.g., 'Which product lines had the highest return rates last quarter?'). 4. Review the analysis steps AIP constructs, then edit or extend them using Insight's existing tools.

— Names exact UI entry points and workflow but no API/config surface

Build Insight workbooks with AIP

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

Superlog Labs Superlog Feedback ⓘ

changes since 2026-10-20 CODE ↗

Superlog monitors AI agent telemetry with OpenTelemetry and groups signals into incidents for observability.

Superlog adds org-level monthly spend limits, MCP request forwarding for migrated projects, and two ClickHouse ingest tuning options

└──▷ TRY IT

Prevent slow-replica write accumulation when running ClickHouse behind a TCP load balancer with insert_quorum enabled.

```
$ INGEST_CLICKHOUSE_KEEP_ALIVE=false
INGEST_CLICKHOUSE_MIN_INSERT_INTERVAL_MS=500 node dist/ingest-
consumer.js
```

Control ClickHouse insert pacing by setting `INGEST_CLICKHOUSE_MIN_INSERT_INTERVAL_MS` to enforce a minimum gap between inserts into the same table, growing batch sizes instead of producing more parts when inserts speed up. Not checked yet

Disable ClickHouse keep-alive per insert by setting `INGEST_CLICKHOUSE_KEEP_ALIVE=false` to open a fresh connection per insert and spread load evenly across replicas behind a TCP load balancer. Not checked yet

Let orgs set a monthly spend limit that freezes all billable features once the estimated bill reaches the ceiling, without blocking usage of already-included allowances. Not checked yet

Forward MCP requests from a cut-over project to its new workspace at `${workspace_url}/api/mcp` with the same JSON-RPC body streamed back, keyed on the new `project_migrations` table. Not checked yet

Cache GCP Pub/Sub connection lookups for 60 seconds and collapse concurrent misses into one query via `createGcpConnectionCache` in `apps/proxy/src/gcp-pubsub-connection.ts`, keeping high-volume push bursts off Postgres. Not checked yet

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Frontier Models

Google Gemini API

Feedback ⓘ updated · Nov 19, 2026

NOTES ↗ RANK

Gemini API provides Gemini models for generating text, images, and other multimodal content in applications.

> Gemini API introduces Genie 4, a real-time interactive world-generation model that streams steerable video and audio directly to applications.

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

01 Genie 4 interactive world generation

Not checked yet NEW

90

Accessible via the `v1beta` Interactions API at the `models/genie-4` endpoint, Genie 4 streams steerable `24 fps` `16:9` video with synchronized `48 kHz` stereo audio over WebRTC and WebSockets. Worlds can be seeded from text, a `16:9` reference image, or both, and steered live using `WASD` or arrow keys plus mid-session text events. On the paid tier it is priced at `$1.50` per 1M input tokens, `$9.00` per 1M text output tokens, and `$17.50` per 1M video output tokens (approximately `$0.10/sec`, stereo audio included).

— Names endpoint, streaming specs, controls, and exact pricing. snapshot-20261119

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · November 19, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue