CLOUDFLARE

Building EmDash

10 months of AI:
what worked, what broke, what changed

Matt Kane · Astro core team · Cloudflare

Cloudflare Connect · 21 October 2026

Dashboard
Calendar

Content
Pages
Posts
Media

Manage
Comments
Menus
Redirects
Widgets
Sections
Categories
Tags
Bylines

Admin
Content Types
Byline Schema
Users
Plugins
Import
Settings

Title

The Quiet Craft of the Second Draft

Featured Image



typewriter-desk.jpg

1200 × 800 · image/jpeg

Replace

Edit asset

Remove

Content

The first draft is for you. The second draft is for everyone else. It's where a pile of notes becomes an argument, and the sentence you loved finally earns its place.

Read it out loud

Paragraph

Your ear catches what your eye forgives. **If you run out of breath, the sentence is too long.** If you stumble, so will your reader. *Skim it, and they'll skip it.*

- Lead with the point, then earn it.
- Trade every *very* for a stronger word.
- End each paragraph on its sharpest line.

> I have rewritten, often several times, every word I have ever published. My pencils outlast their erasers. — Vladimir Nabokov

Cut what argues with you

Publish

Live version

Visitors still see the published version

Draft changes

Ready to publish now or schedule for later

Discard changes

Publication date

Oct 7, 2026, 7:48 PM

Created and updated

Schedule

URL & language

Slug

the-quiet-craft-of-the-second-draft

Content language

English EN

Ownership

Jordan Ellis

jordan@fieldnotes.blog

Bylines

Jordan Ellis

Taxonomies

CLOUDFLARE

The prototype

Project Ecliptic: PRD

Version: 0.1 (Draft)
Last Updated: January 18, 2026
Author: Matt Kane + Innie
Origin: Dane idea

Executive Summary

Project Ecliptic: an agent-portable reimplementation of WordPress on Astro.

Not a 1:1 port - a spiritual successor that keeps WordPress's good bits (extensibility, admin UX, plugin ecosystem) while ditching 20 years of PHP cruft. The key insight: WordPress concepts map cleanly enough to modern equivalents that an AI agent can port most plugins in under an hour.

What "agent-portable" means:

- Not API-compatible (different language, different runtime)
- Not a transpiler (no deterministic conversion)
- Concepts map clearly enough that an agent can reimplement with minimal guidance
- Same intent, modern implementation

Key points:

- **Astro-native** - Content Collections, Actions, Sessions, Live Collections
- **Edge-ready** - SQLite/libSQL database, S3-compatible storage, deploy anywhere



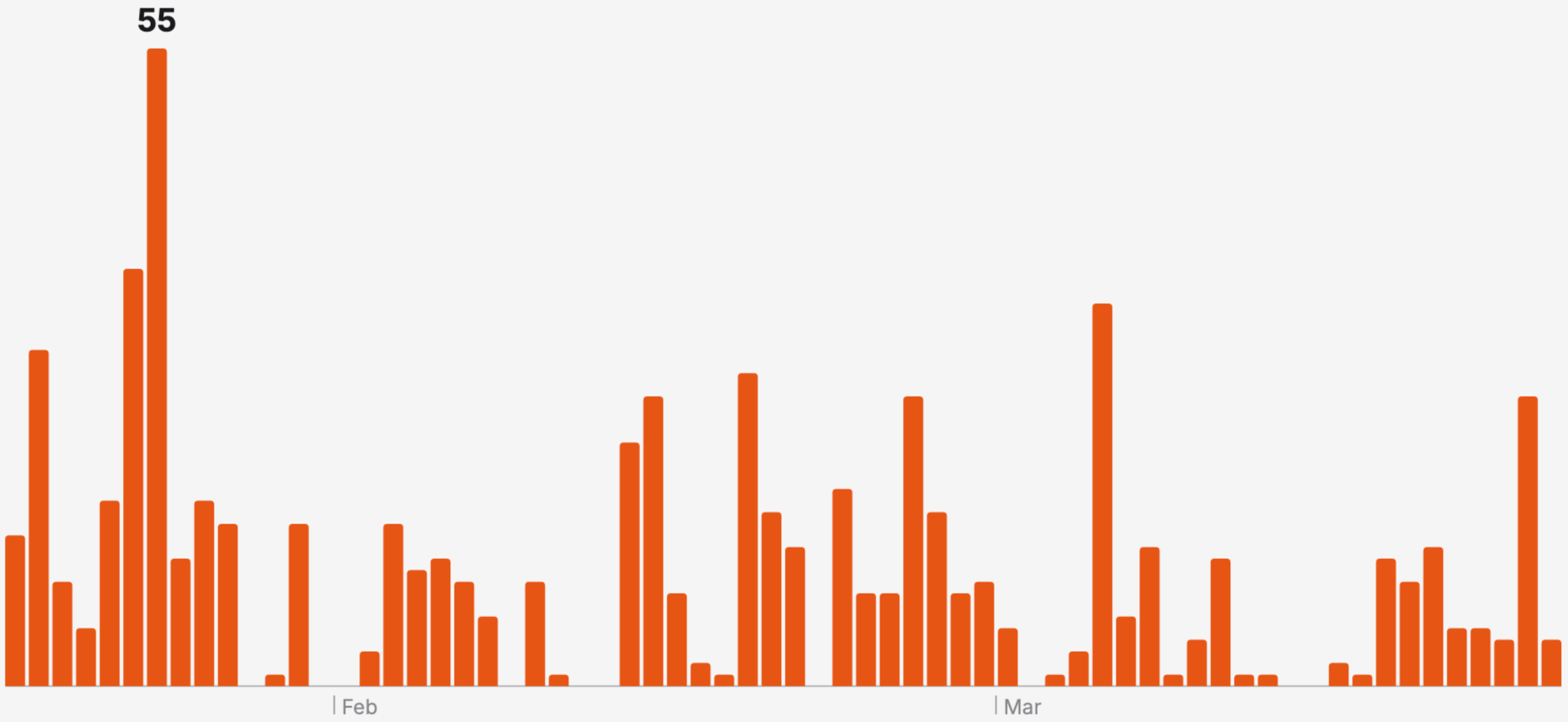
.opencode/ralph-loop.state.json

```
### Admin UI - Editor

- [x] Create menu editor page
- [ ] Implement "Add Items" panel with Pages/Posts/Categories sections
- [x] Implement "Custom Link" form
- [ ] Implement draggable menu item list (using dnd-kit)
- [x] Support reorder (basic up/down buttons implemented)
- [ ] Support drag to nest
- [x] Item expansion panel for editing
- [x] Save functionality with loading state
- [x] Error handling

## Remaining Work

The build is currently failing due to missing UI components. Fix these first,
then complete the remaining features.
```


55
Feb
Mar

CLOUDFLARE

Opening the repo

The Register

Sign in

AI | Security | Software | Microsoft | Developer | Open Source | BOFH | Who, Me? | On Call | Science

DEVOPS

Cloudflare previews 'EmDash' – an AI-driven rebuild of WordPress in TypeScript

Name is a joke but the project is real, said main engineer

24

Tim Anderson

Published Thu 02 Apr 2026 // 14:04 UTC

Make us preferred on Google

The world's most popular CMS has been remade with the help of AI. Cloudflare has released EmDash version 0.1, described as a rebuild of the WordPress CMS (content management system) but using TypeScript rather than PHP.

In contrast to the one week claimed for recreating Next.js using agentic AI,

---

Hacker News new | past | comments | ask | show | jobs | submit

EmDash – A spiritual successor to WordPress that solves plugin security (cloudflare.com)

703 points by ddthaw 6 months ago | hide | past | favorite | 504 comments

embedding-shape 6 months ago | next [–]

> Our name for this new CMS is EmDash. We think of it as the spiritual successor to WordPress. It's written entirely in TypeScript. It is serverless, bu securely sandboxed and can run in their own isolate, via Dynamic Workers, solving the fundamental security problem with the WordPress plugin ar framework for content-driven websites.

To me this sounds of the polar opposite of the direction CMS's need to go, instead simplify and go back to the "websites" roots where a website are with than server-side rendered websites.

But of course, then they wouldn't be able to sell their own "workers" product, so suddenly I think I might understand why they built it the way they bui

I'm not sure it actually solves the "fundamental security problem" in actuality though, but I guess that remains to be seen.

perlgeek 6 months ago | parent | next [–]

I love building static (or statically generated) websites, but all too often, customers want dynamic content. And what's worse, they don't tell yc

"I need a website for my bakery". "What's supposed to be on it?" "Our address, opening times, a few pictures". I build them a static website.

"Now I need a contact form". Ok, that doesn't really fit into a static website, but I can hack something together. "Now I need to show inventory,

When you develop for clients, especially those that you don't know very well, it's a bad idea to back yourself into a corner that's not very exte

yurishimo 6 months ago | root | parent | next [–]

Joost.blog

EmDash: a CMS built for 2026

The way we build software changed forever at the end of 2025. Suddenly our AI agents went from smart auto-complete to actually being able to develop software. We stopped writing

---

techradar | techradar pro | trg

techradar pro THE BUSINESS TECHNOLOGY EXPERTS

RSS Sign in Search

News | Reviews | Features | Perspectives | Website builders | Web hosting | Security

Cloudflare launches EmDash — the 'spiritual successor' that wants to take on WordPress

News By Craig Hale | Published 2 April 2026

Cloudflare's EmDash tackles WordPress's plugin problem



Search Engine Journal

AI Search | SEO | Paid Media | Digital Marketing | Resources | Webinars

Mullenweg To Cloudflare: Keep WordPress Out Of Your Mouth

Matt Mullenweg invokes Will Smith's Oscars slap in response to Cloudflare boasting that EmDash is a successor to WordPress.



SEJ STAFF

Roger Montti

April 3, 2026 · 5 min read

4.3K READS

MA.TT UNLUCKY IN CARDS

EmDash Feedback

April 2, 2026 AI, WordPress

So, two other Matts at Cloudflare announced EmDash — the spiritual successor to WordPress that solves plugin security.

(Is it nominative determinism or a simulation glitch that everyone trying to terraform the web has some variation of "Matthew" in their name? I was in a call set up by Matthew Prince, talking to Matt Taylor and Matt Kane, with

---

Slashdot

Stories | Firehose | All | Popular | Polls | Software | Thought Leadership | Submit

Topics: Devices | Build | Entertainment | Technology | Open Source | Science | YRO

Search

Cloudflare Announces EmDash As Open-Source 'Spiritual Successor' To WordPress (phoronix.com)

Posted by BeauHD on Wednesday April 01, 2026 @04:02PM from the not-a-joke dept.

In classic Cloudflare fashion, the CDN provider used April Fool's Day to unveil an actual, "not a joke" product. Today, the company announced EmDash -- an open-source "spiritual successor" to WordPress that aims to solve plugin security. Phoronix reports:

> With the help of AI coding agents, Cloudflare engineers have been rebuilding the WordPress open-source project "from the ground up." EmDash is written entirely in TypeScript and is a server- design. Making plug-ins more secure than the WordPress architecture, EmDash plug-ins are sandboxed and run in their own isolate. EmDash builds upon the Astro web framework. EmDash d rely on any WordPress code but is designed to be compatible with WordPress functionality. EmDash is open-source now under the MIT license.

The EmDash code is available on GitHub.

WEBHOSTINGTODAY

Home | Web Hosting M&A | Jobs | Buzz | Podcast | More | Market Insights

Cloudflare Just Released a WordPress Competitor. It Is Not April Fools.

by Łukasz Nowak · April 2, 2026 · Software reviews · 8 min read



---

techradar | techradar pro | trg

techradar pro THE BUSINESS TECHNOLOGY EXPERTS

News | Reviews | Features | Perspectives | Website builders | Web hostin

'I think EmDash was created to sell more Cloudflare services': WordPress co-founder Matt Mullenweg gives his verdict on Cloudflare's EmDash

News By Craig Hale | Published 3 April 2026

WordPress co-founder says he welcomes new competition



1 – 8 Apr 2026

88 open

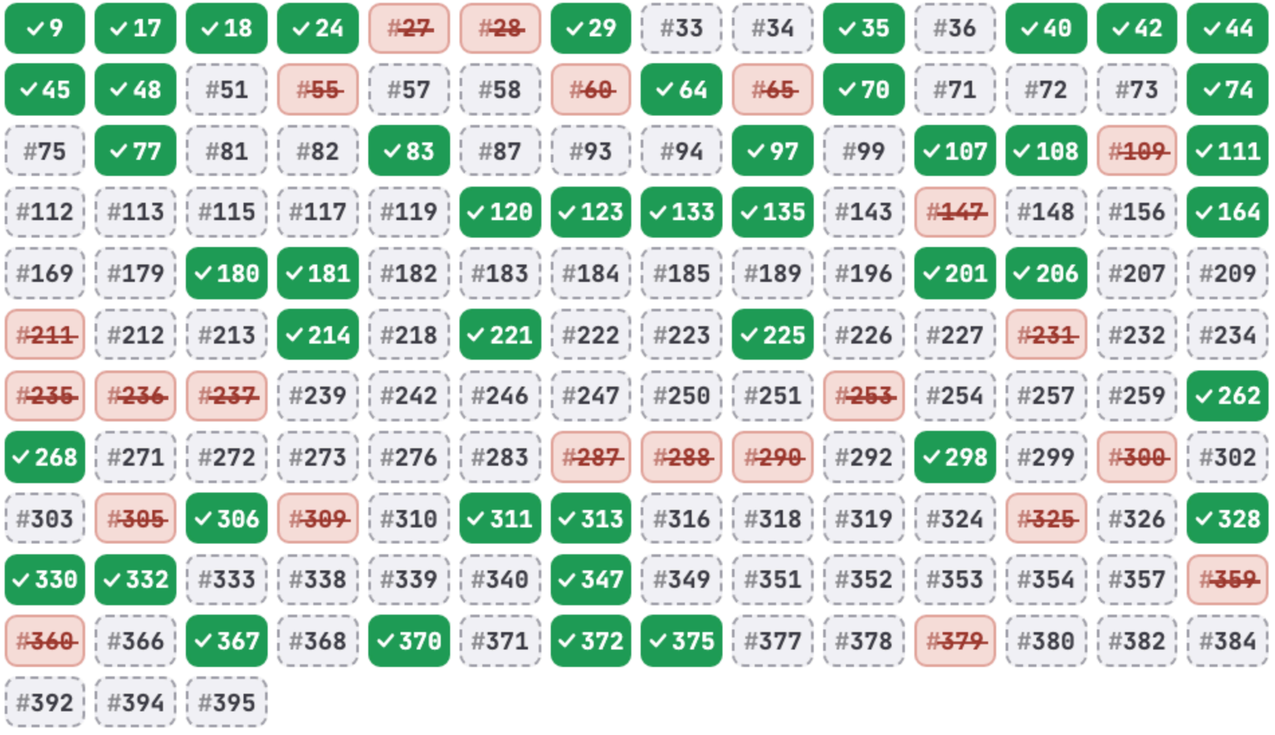
46 merged

23 closed

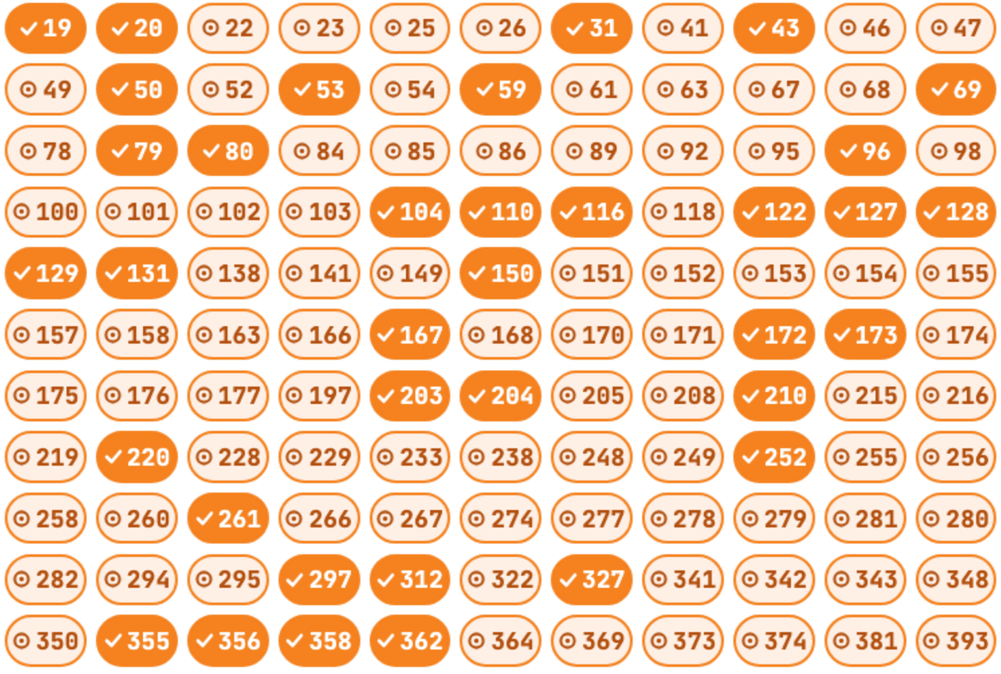
85 open

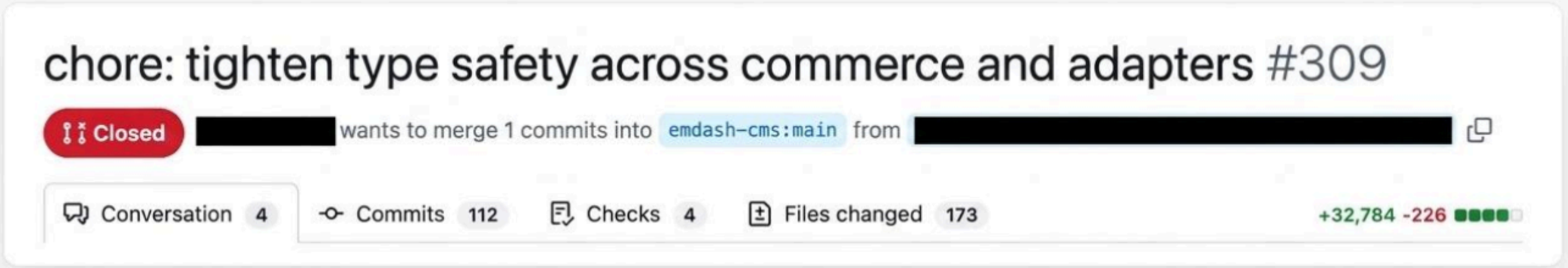
36 closed

Pull requests 157



Issues 121

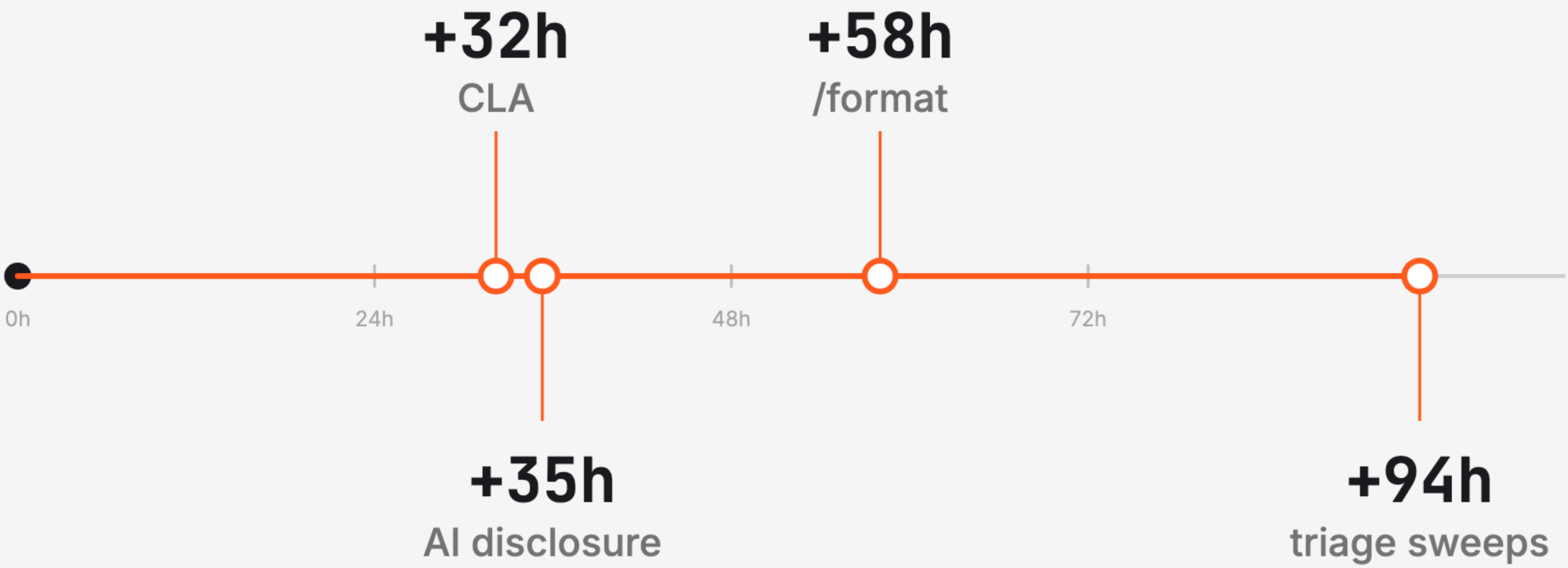


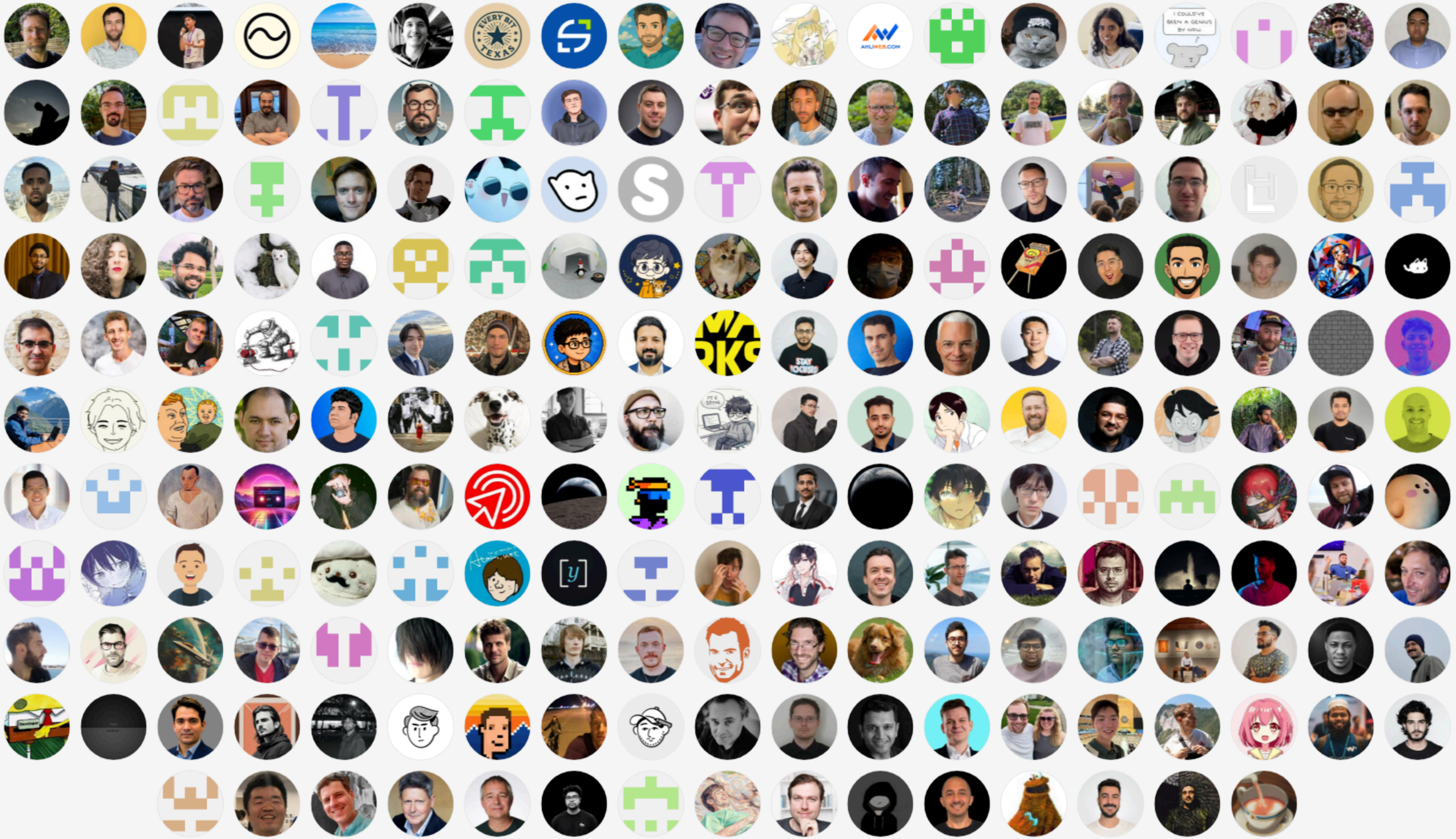

chore: tighten type safety across commerce and adapters #309
Closed
[redacted] wants to merge 1 commits into emdash-cms:main from [redacted]
Conversation 4
Commits 112
Checks 4
Files changed 173
+32,784 -226

CLOUDFLARE

What would you automate first?




+32h
CLA
+58h
/format
0h
24h
48h
72h
+35h
AI disclosure
+94h
triage sweeps



CLOUDFLARE

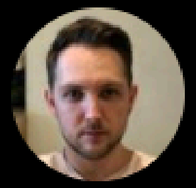
Automating maintenance

emdashbot Bot 3 weeks ago Contributor ···

[needs fixing] `ctx.http.fetch` is produced by `createHttpAccess` (or `createUnrestrictedHttpAccess`), which manually follows redirects using `globalThis.fetch(..., { redirect: "manual" })` so it can strip credentials on cross-origin hops. Because each internal fetch uses manual redirect mode, the final `Response` it returns has `redirected: false` even when redirects were followed. The `response.redirected` guard therefore means this branch never fires in production: a webhook that redirects to a sign-in page and returns 200 is treated as success, exactly the silent failure the PR says it is fixing. Drop the `response.redirected` requirement (and normalize URL comparison if you want to avoid trailing-slash false positives) or surface redirect metadata from the HTTP wrapper itself.

Suggested change

```
-                                } else if (response.redirected
  && response.url !== url) {
+                                } else if (response.url !==
  url) {
```



ZEE

@____zee___

I have to say @EmDashCMS bot is a pleasure to work with...

1️⃣ I raise an issue, it triages
2️⃣ proposes a fix
3️⃣ I (my bots) approve and fix if needed
4️⃣ the lovely maintainers get it merged!

this is software engineering in 2026

7:48 PM · Oct 8, 2026 · **21** Views

CLOUDFLARE

Sharing responsibility

emdash-cms/emdash · updated 12:45 UTC

What the EmDash bot is working on

Every issue and pull request the bot has, grouped by who acts next. Everything here is also on GitHub; act there by labelling, commenting or reviewing.

0 needs a maintainer | 4 ready for review | 1 working | 0 waiting on CI, review or a reply

0 of 4 slots in use

Needs a maintainer 0

The bot can't go further without you

Nothing needs a maintainer.

Ready for human review 4

Checks passed, reviewer approved, merges cleanly

#4013 **fix(admin): provider-specific loginHelp and neutral fallback on LoginPage**

A maintainer will review and approve the pull request.

PR #4013 assigned by @ascorbic open 2 h 0 m last activity 37 m ago

Ready

checks 7/7 review bot ✓

#4070 **fix(core): delete orphan _emdash_fields row when createField fails on D1**

A maintainer will review and approve the pull request.

PR #4070 assigned by @ascorbic open 2 h 1 m last activity 38 m ago

Ready

checks 7/7 review bot ✓

#4024 **fix(core): tolerate missing _emdash_relations.slug during pending migrations**

A maintainer will review and approve the pull request.

PR #4024 assigned by @ascorbic open 2 h 0 m last activity 39 m ago

Ready

checks 7/7 review bot ✓

#4074 **fix(core): remove stale slug-change redirects that shadow new entries at the old URL**

A maintainer will review and approve the pull request.

PR #4074 assigned by @ascorbic open 2 h 0 m last activity 39 m ago

Ready

checks 7/7 review bot ✓

WordPress-imported `buttons` blocks can't be edited in the admin: generic "Enter URL..." fallback saves a stray URL on the block #4313

New issue

Open #4314

NegativeAL opened 18h ago

Summary

The WordPress importer (`@emdash-cms/gutenberg-to-portable-text`) converts Gutenberg `core/buttons` / `core/button` into Portable Text blocks:

```
{"_type":"buttons","layout":"horizontal","buttons":[
  {"_type":"button","text":"Official results","url":"https://…","style":"default"}
]}
```

These render fine on the site through `Buttons.astro` / `Button.astro`. The admin has no editor for them, though. `getEmbedMeta()` doesn't find `buttons` in the block registry, so it falls back to the generic embed box with a single "Enter URL..." input.

Problems

1. Imported buttons can't be edited: you can't change their text or URL, add or remove a button, or set the layout.
2. Typing a URL into that box saves it as `id` on the parent `buttons` block, not on any button. Nothing displays that URL, so the editor appears to do nothing and leaves stray data behind.
3. Nothing in the slash menu or toolbar inserts a new buttons block, even though the importer creates them and the renderer supports them.

Steps to reproduce

1. Import a WordPress post containing a `wp:buttons` group with several `wp:button` links.
2. Open the post in the admin. The group appears as one "Buttons" card with an empty "Enter URL..." field.

Assignees

No one - Assign yourself

Assign to Agent

Labels

area/admin bot:fix triage/verified

Type

No type

Fields

Priority Choose an option

Projects

No projects

Milestone

No milestone

Relationships

None yet

Development

fix(admin): prevent URL editing of

Read the report ↗ Preview request ↗ Reporter response ↗

CLOUDFLARE

What would you let the model decide?



Merge requirements

A normal pull request can merge when:

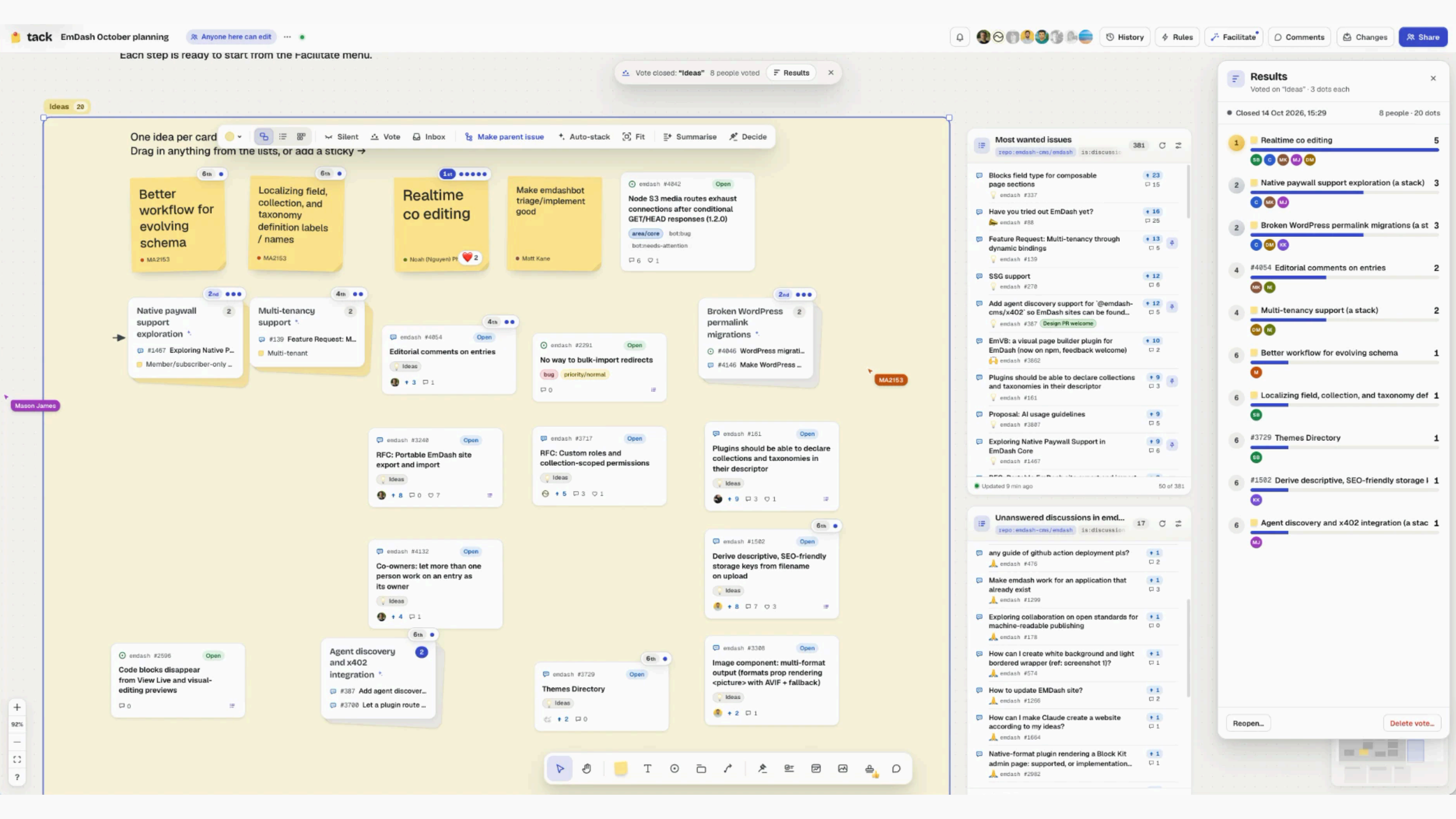
- required checks pass;
- blocking review threads are resolved;
- at least one human Maintainer other than the author has approved the current revision; and
- the change follows an accepted design when the contribution process requires one.

These requirements apply to the Project Lead, Cloudflare employees, external Maintainers, and other contributors. Routine self-merge is not permitted.

The author is the pull request's main author. A Maintainer who has made substantial changes to someone else's pull request, including an EmDashBot pull request, also counts as an author and cannot provide its approval. Automation only excludes the account that opened the pull request, so Maintainers apply this rule themselves.

An approval applies to the reviewed revision. A later substantive commit requires another approval. A merge commit that only updates the branch from `main` does not invalidate an approval.

Project Team reviews, automated reviews, test results, and static analysis help the human reviewer. They do not replace Maintainer approval. A bot with repository permissions is still a bot, not a Maintainer.


tack
EmDash October planning
Anyone here can edit
History
Rules
Facilitate
Comments
Changes
Share
Each step is ready to start from the Facilitate menu.
Vote closed: "Ideas" 8 people voted
Results
Ideas 20
One idea per card
Drag in anything from the lists, or add a sticky →
Silent
Vote
Inbox
Make parent issue
Auto-stack
Fit
Summarise
Decide
Better workflow for evolving schema
MA2153
Localizing field, collection, and taxonomy definition labels / names
MA2153
Realtime co editing
Noah (Nguyen) Pl
Make emdashbot triage/implement good
Matt Kane
emdash #4042 Open
Node S3 media routes exhaust connections after conditional GET/HEAD responses (1.2.0)
area/core bot:bug bot:needs-attention
Native paywall support exploration 2
#1467 Exploring Native P...
Member/subscriber-only ...
Multi-tenancy support 2
#139 Feature Request: M...
Multi-tenant
emdash #4054 Open
Editorial comments on entries
Ideas
emdash #2291 Open
No way to bulk-import redirects
bug priority/normal
Broken WordPress permalink migrations 2
#4046 WordPress migrati...
#4146 Make WordPress ...
MA2153
Mason James
emdash #3240 Open
RFC: Portable EmDash site export and import
Ideas
emdash #3717 Open
RFC: Custom roles and collection-scoped permissions
Ideas
emdash #161 Open
Plugins should be able to declare collections and taxonomies in their descriptor
Ideas
emdash #4132 Open
Co-owners: let more than one person work on an entry as its owner
Ideas
emdash #1502 Open
Derive descriptive, SEO-friendly storage keys from filename on upload
Ideas
emdash #2596 Open
Code blocks disappear from View Live and visual-editing previews
Agent discovery and x402 integration 2
#387 Add agent discover...
#3700 Let a plugin route ...
emdash #3729 Open
Themes Directory
Ideas
emdash #3308 Open
Image component: multi-format output (formats prop rendering <picture> with AVIF + fallback)
Ideas
92%
Most wanted issues
381
Blocks field type for composable page sections
Have you tried out EmDash yet?
Feature Request: Multi-tenancy through dynamic bindings
SSG support
Add agent discovery support for `@emdash-cms/x402` so EmDash sites can be found...
EmVB: a visual page builder plugin for EmDash (now on npm, feedback welcome)
Plugins should be able to declare collections and taxonomies in their descriptor
Proposal: AI usage guidelines
Exploring Native Paywall Support in EmDash Core
Updated 9 min ago
50 of 381
Unanswered discussions in emd...
17
any guide of github action deployment pls?
Make emdash work for an application that already exist
Exploring collaboration on open standards for machine-readable publishing
How can I create white background and light bordered wrapper (ref: screenshot 1)?
How to update EMDash site?
How can I make Claude create a website according to my ideas?
Native-format plugin rendering a Block Kit admin page: supported, or implementation...
Results
Voted on "Ideas" · 3 dots each
Closed 14 Oct 2026, 15:29
8 people · 20 dots
1 Realtime co editing 5
2 Native paywall support exploration (a stack) 3
2 Broken WordPress permalink migrations (a st 3
4 #4054 Editorial comments on entries 2
4 Multi-tenancy support (a stack) 2
6 Better workflow for evolving schema 1
6 Localizing field, collection, and taxonomy def 1
6 #3729 Themes Directory 1
6 #1502 Derive descriptive, SEO-friendly storage l 1
6 Agent discovery and x402 integration (a stac 1
Reopen...
Delete vote...

Who does what?

Work	The prototype	Opening the repo	Automating maintenance	Sharing responsibility
Code	My agents	Contributors + agents	Contributors + bot	Agents + bot
Review	No reviewer	Me	Bot + human	Bots + maintainers
Triage	None yet	Scripts + backlog	Bot	Bot + code
Merge	Direct to main	Me	Maintainer	Five maintainers
Direction	Me + spec	Me	Me	Team

CLOUDFLARE

What would you change in your team?

Writing, checking, or deciding:
where is the next bottleneck?

emdashcms.com · github.com/emdash-cms/emdash