XI'AN · HYPERSCALE FOR RADIX

WEEK 32 · 3 AUG 2026

3–9 Aug 2026

Hyperscale Weekly

Split the work, not the world

CURRENT MILESTONE

Milestone 2 · Radix Engine

Porting the Radix Engine so Scrypto runs across shards

Week 1

Previously: flightofthefox had wired in make-before-break committee rotation, overhauled the beacon's randomness, and drawn enthusiastic community reaction with a WASM browser demo of the full node.

This week straddled the finish line of Milestone 1 and the opening days of Milestone 2. flightofthefox confirmed adaptive sharding complete, then dove straight into the Radix Engine work — and the first act was a massive cleanup: merging to one engine, dropping the last Radix library dependencies, and reshaping cross-shard settlement around a unified 'tick' model. It was the biggest single week of refactoring yet.

246
+26% vs last wk
COMMITS

+23k −9.4k
-5% vs last wk
LINES CHANGED

559
FILES TOUCHED

34+
DAY COMMIT STREAK

COMMITS / DAY



peak 53/day · 7/7 days active

WHAT FLIGHTOFTHEFOX BUILT

One engine, no Radix leftovers

flightofthefox merged the two-engine setup into a single engine crate, adopted a new binary encoding called HBOR, and dropped the last external Radix libraries. The codebase now stands on its own feet.

'see "The System" on hyperscale.rs

Cross-shard settlement finds its final shape

Dozens of commits unified how cross-shard transaction batches — now called 'ticks' — are certified, settled, and fenced when a shard terminates. The outcome of any payment is decided by content, not by committee identity.

'see "The Generals" on hyperscale.rs

Fees and gas budgeting arrive

Transactions now carry a signed gas budget, the network meters work rather than just counting transactions, and fee burns land in each payer's own receipt. This is the fee system the sharded engine needs to run for real.

'see "The Journey" on hyperscale.rs

THE BOTTOM LINE

STANDOUT CHANGE

The codebase now runs on one engine with zero external Radix dependencies — a clean foundation for everything Milestone 2 will build.

WHAT'S NEXT

Likely next: pushing deeper into fee-system wiring and porting more Babylon feature parity onto the now-unified engine.

Written by AI from public commits, the community chat and hyperscale.rs — may contain mistakes.

From the community

What people are saying, and the words behind the work.

HEARD IN THE CHAT

The mood: Lively community discussion about hardware requirements — Raspberry Pi nodes, mobile devices, backup connections — plus a deep dive into validator penalties and staking economics.

> *"i implement impounding, not slashing. I.E. your stake becomes locked for 90 days... it's just stopping a node runner from being able to repeat attacks, without actually slashing their money permanently"*
>
> — FLIGHTOFTHEFOX, IN THE COMMUNITY TELEGRAM

THIS WEEK'S CONCEPT

The System — *Split the work, not the world.* The engine consolidation and Radix dependency removal is the foundational Milestone 2 move — everything else this week rides on top of having one clean engine instead of two inherited ones.

hyperscale.rs

JARGON, DECODED

Tick — A batch of transactions that a shard processes and certifies together — the unit of cross-shard execution.

HBOR — Hyperscale's own binary encoding format, replacing the Radix library's encoding so the codebase has no external dependencies.

Impounding — Locking a misbehaving validator's stake for a set period rather than permanently destroying it — a gentler alternative to slashing.

Written by AI from public commits, the community chat and hyperscale.rs — may contain mistakes.

ZOOM OUT

The bigger picture

Where the work landed, and the map to read along on hyperscale.rs.

MOMENTUM · LINES CHANGED / WEEK



last 8 weeks · newest right

THIS WEEK'S WORK



79% Rust · 19% Config · 2% Docs

WHERE THE WORK LANDED

node · The System
engine merge and Radix dependency removal

execution · The Generals
tick, wave, finalization and settlement

vm/hbor · The System
new binary encoding replacing Radix types

THE FULL PICTURE

The System
The whole sharded design at a glance.

The Journey
One transaction's trip through the network.

The Clock
Consensus-attested time across independent shards.

The Census
The leaderless beacon of validators, stake and shards.

The Overlap
Why two conflicting blocks can never both commit.

The Generals
All-or-nothing cross-shard commits, computed not voted.

The Archive
Every published byte is preserved or provably expired.

The Library
All state in one merkle tree; a shard is a subtree.

The Will
How in-flight transactions settle when a shard dies.

The Lottery
Random, ever-moving committees; proven cheats are jailed.

The Triage
Under overload, urgent traffic never waits for bulk.

The Governor
Validator entry is re-priced every epoch, like a market.

The Crash Lab
The simulator that replays any failure byte-for-byte.

The Proof
Proving safety with maths, not just tests.

The Asterisks
Every design's trade-offs — including Hyperscale's own.

THE ROAD TO MAINNET

M1	M2	M3	M4	M5	M6
Adaptive Sharding	Radix Engine	Gateway & API	Validator GUI	Migration	Live support
done	~5 mo	~3 mo	~3 mo	~3 mo	12 mo

Now: Milestone 2 · Radix Engine — Week 1. Since day one: 2,968 commits — one developer, fully in public.

LINKS

WEBSITE
hyperscale.rs

DISCUSS THIS ISSUE ON X
x.com/i/status/2086679275067875696

CODE (PUBLIC)
github.com/hyperscalers/hyperscale-rs

MILESTONE PROPOSAL (PDF)
hyperscale.rs/xian_proposal.pdf

COMMUNITY (TELEGRAM)
t.me/hyperscale_rs

AUTHOR
flightofthefox

scan for hyperscale.rs

Questions? flightofthefox is always happy to discuss in the community Telegram — t.me/hyperscale_rs.

Written by AI from public commits, the community chat and hyperscale.rs — may contain mistakes.