XI'AN · HYPERSCALE FOR RADIX

WEEK 31 · 27 JUL 2026

27 Jul – 2 Aug 2026

Hyperscale Weekly

Split the work, not the world

CURRENT MILESTONE

Milestone 1 · Adaptive Sharding

Teaching the network to split and merge its shards on its own

Week 14

Previously: flightofthefox had completed the cut-over system for shard splits and merges, plus a full fork-detection and jailing pipeline, with validator-shuffling integration flagged as the clear next frontier.

This week flightofthefox delivered on that promise: committee rotation — the one-seat-at-a-time shuffling that keeps shards live — landed as a full make-before-break system wired into the beacon. He also rebuilt how the beacon produces its randomness and validates witness records, and topped it off with a stunning browser demo running the entire node in a tab.

78	+9.8k −2.9k	154	0
-38% vs last wk	-4% vs last wk		
COMMITS	LINES CHANGED	FILES TOUCHED	DAY COMMIT STREAK

COMMITS / DAY  peak 38/day · 4/7 days active

WHAT FLIGHTOFTHEFOX BUILT

Committee shuffling goes live

flightofthefox built the full make-before-break rotation: committees rotate one seat at a time, capped by sync headroom, so shards never lose quorum mid-shuffle. Merges cancel pending rotations, keeping topology changes safe.

'see "The Lottery" on hyperscale.rs

Beacon randomness, rebuilt

A per-epoch randomness reveal chain now folds into beacon randomness, with formal invariants registered and old witness types cleaned up. This is the lottery's fuel — the unpredictable seed that stops anyone rigging committee assignments.

'see "The Census" on hyperscale.rs

The whole node in a browser

flightofthefox compiled the full node to WASM and ran an eight-node, multi-shard network in a browser tab with simulated latency. The community loved it — a tangible, visual proof that the architecture works.

'see "The Crash Lab" on hyperscale.rs

THE BOTTOM LINE

STANDOUT CHANGE

Committee rotation — the one-seat-at-a-time shuffling at the core of adaptive sharding — is now wired in and running make-before-break.

WHAT'S NEXT

Likely next: hardening the shuffle against edge cases like concurrent merges, then pushing the demo toward community-runnable testnet readiness.

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: Huge excitement around the browser demo; thoughtful Q&A about beacon-chain resilience and cross-shard wave-cert safety from foxylady and others.

> *"what happens when you compile the whole node to WASM and slap a multi-shard network in a browser tab 😅"*
>
> — FLIGHTOFTHEFOX, IN THE COMMUNITY TELEGRAM

THIS WEEK'S CONCEPT

The Lottery — *A jury you can't bribe.* Committee rotation is the heart of validator shuffling — the mechanism that disperses corruption one seat at a time while never leaving a shard without enough honest validators to function.

hyperscale.rs/kleroterion

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



88% Rust · 6% Config · 3% Docs · 3% Specs

WHERE THE WORK LANDED

beacon · The Census
randomness reveal chain and witness proof overhaul

types · The Lottery
committee rotation and shuffle records added

demo · The Crash Lab
browser multi-shard simulation with cross-shard visuals

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
~4 mo	~5 mo	~3 mo	~3 mo	~3 mo	12 mo

Now: Milestone 1 · Adaptive Sharding — Week 14. Since day one: 2,605 commits — one developer, fully in public.

LINKS

WEBSITE
hyperscale.rs

DISCUSS THIS ISSUE ON X
x.com/i/status/2084142297801748618

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.