XI'AN · HYPERSCALE FOR RADIX

WEEK 25 · 15 JUN 2026

15–21 Jun 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 8

Previously: flightofthefox had just landed the settled-waves fence for straddlers, the first end-to-end shard split and merge in the simulator, and per-shard boundary QCs anchoring crossings in the beacon — work moving from parts t

A heads-down, no-chat week focused on closing the gap between the simulator and the real network. flightofthefox made the sim mirror production at the same epoch length and with the same follower pools, multiplied the end-to-end tests on the production side, and quietly introduced the first on-chain knob validators can vote to retune. The foundations are starting to feel production-shaped.

78	+10k −4.1k	159	30
-52% vs last wk	-26% vs last wk		
COMMITS	LINES CHANGED	FILES TOUCHED	DAY COMMIT STREAK

COMMITS / DAY



peak 16/day · 7/7 days active

WHAT FLIGHTOFTHEFOX BUILT

Validators get a live tuning knob

flightofthefox added NetworkParams to the beacon at genesis and a way for stake-pool votes to retune them at runtime. It's the first self-governing dial in Hyperscale — a thermostat the community can turn, not a setting frozen by the developers.

'see "The Governor" on hyperscale.rs

The simulator catches up to production

flightofthefox rebuilt the sim around the same epoch length and follower pool the real network uses, made message iteration deterministic, and added request retries and simulated latency. The crash lab is now a faithful rehearsal space, not a smaller parallel world.

'see "The Crash Lab" on hyperscale.rs

Reshape gets production-grade tests

End-to-end tests for the split, the merge, and straddler settlement across the reshape boundary all turned on in production. The pacemaker now starts the moment a new shard is seated, cohort seeds lock at split time, and a merge parent composes cleanly from its child terminals.

'see "The Will" on hyperscale.rs

THE BOTTOM LINE

STANDOUT CHANGE

The simulator and the production test suite are now running the same epochs, the same follower pools, and deterministic iteration — the first realistic dress rehearsal for a public testnet.

WHAT'S NEXT

Likely next: stabilise the production reshape test suite under fault scenarios before moving on to the lottery-style validator shuffling also promised in M1.

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.

THIS WEEK'S CONCEPT

> **The Crash Lab** — *Replay every bug byte-for-byte.* This week's biggest theme was the simulator and production test suite converging onto the same epochs, the same follower pools, and deterministic iteration — turning the crash lab into a realistic dress rehearsal for a public testnet.
>
> hyperscale.rs

JARGON, DECODED

NetworkParams — The on-chain settings that describe how the network runs — like timeouts and committee sizes — stored in the beacon so validators can vote t

Stake pool — A group of XRD holders who delegate their stake to a single validator so they can vote and earn rewards together.

Cohort seed — A random number that locks in exactly which validators get assigned to a brand-new shard the moment it splits off.

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



98% Rust 2% Config

WHERE THE WORK LANDED

simulation · The Crash Lab
mirrors production epochs and follower pool

node · The Census
beacon-follower pool and route gate

beacon · The Governor
NetworkParams and stake-pool voting

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 8. Since day one: 2,324 commits — one developer, fully in public.

LINKS

WEBSITE
hyperscale.rs

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.