XI'AN · HYPERSCALE FOR RADIX

WEEK 23 · 1 JUN 2026
1–7 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 6

Previously: flightofthefox had wrapped nearly every consensus type in a compiler-enforced 'verified' tag and wired the beacon into the node end-to-end.

This week flightofthefox pushed the verified-tag discipline all the way down into storage and reproduced a real shard-consensus fork in the simulator — then closed it. A timestamp-driven topology schedule now answers 'who's in charge right now' for any moment.

86	+5.4k −2.8k	283	22
-53% vs last wk	-68% vs last wk		
COMMITS	LINES CHANGED	FILES TOUCHED	DAY COMMIT STREAK

COMMITS / DAY  peak 18/day · 7/7 days active

WHAT FLIGHTOFTHEFOX BUILT

Storage learns to speak verified

Reads and writes on both shard and beacon storage now return 'verified' blocks — the typestate net from last week tightened so nothing unproven can sneak through persistence. This makes the safety rules structural, not just procedural.

'see "The Overlap" on hyperscale.rs

The clock becomes queryable

A new per-epoch topology schedule lets any node ask which validators run shard X at time T and get the same answer everywhere. Shard IDs became prefix paths, and committees are computed one epoch ahead — the 'three engines, one clock' idea made real.

'see "The Clock" on hyperscale.rs

Shard consensus locks into HotStuff-2

flightofthefox reproduced a conflicting-commit fork caused by an old vote-unlock rule, then closed it with the HotStuff-2 safe-vote rule. Rounds now increase per block, parent hashes are signed, and a timeout pacemaker keeps validators in sync.

'see "The Overlap" on hyperscale.rs

THE BOTTOM LINE

STANDOUT CHANGE

A conflicting-commit fork that could have hit real shards is now reproduced in the simulator and shut down by the HotStuff-2 safe-vote rule.

WHAT'S NEXT

Likely next: driving a full epoch transition — with the new topology schedule and HotStuff-2 rule — through The Crash Lab end-to-end.

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 Clock** — *Three engines, one clock.* The new timestamp-queryable topology schedule is the architectural centerpiece of the week — it's the concrete lookup table that makes 'three engines, one clock' work for every honest node, every time.
>
> hyperscale.rs/clock

JARGON, DECODED

HotStuff-2 — A two-round voting rule that makes it mathematically impossible for honest validators to commit conflicting blocks.

Pacemaker — A timer inside the consensus engine that nudges all validators to move on to the next round if no block appears in time.

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



100% Rust

WHERE THE WORK LANDED

types · The System
typestate wrappers ripple through dozens of files

beacon · The Census
overflow guards and duplicate-committee rejection

shard · The Overlap
HotStuff-2 safe-vote rule and timeout pacemaker

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 6. 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.