XI'AN · HYPERSCALE FOR RADIX

WEEK 20 · 11 MAY 2026

11–17 May 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 3

A quiet opening week for Hyperscale: no code committed and no chat activity captured. The project has just kicked off its 18-month build, and this calm moment is a good chance to look at the road ahead.

0	+0 −0	0	0
-100% vs last wk	-100% vs last wk		
COMMITS	LINES CHANGED	FILES TOUCHED	DAY COMMIT STREAK

WHAT FLIGHTOFTHEFOX BUILT

A calm start to the build

flightofthefox's first reporting week closed with no commits and a quiet chat. The M1 work is just getting underway, and the public repository is empty so far.

'see "The System" on hyperscale.rs

The map is already drawn

The project website lays out the full design vocabulary in plain language, giving the community a shared mental model of what M1 will deliver — starting with the beacon that tracks validators.

'see "The Census" on hyperscale.rs

THE BOTTOM LINE

STANDOUT CHANGE

A quiet first week — no code has landed yet, but the project's 18-month build is officially underway.

WHAT'S NEXT

Likely next: flightofthefox will start landing the first commits for the beacon — the slow chain that tracks validators and the shard map.

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 System** — *Split the work, not the world.* With no code yet, this week is about the whole design coming into focus — the 'split the work, not the world' thesis that M1 will turn into running software.
>
> hyperscale.rs

JARGON, DECODED

Shard — A slice of the network that processes its own transactions in parallel with the others.

BFT — Short for Byzantine Fault Tolerant — a way for a group of computers to agree on something even if some of them are dishonest.

Beacon — A slow coordination chain that keeps track of who the validators are and how the shards are arranged.

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

WHERE THE WORK LANDED

A quiet week in the codebase.

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 3. 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.