

The Prologue

Split the work, not the world

Before the first weekly digest, there were 1,348 commits. This is where the machine came from.

1,348

COMMITTS BEFORE THE FIRST WEEKLY · SINCE 7 DECEMBER 2025

Every week, this digest follows flightofthefox — building Hyperscale in the open: a sharded, Rust-native rewrite of Radix designed to get faster as validators join, instead of grinding to a halt. But the story doesn't begin at Week #1. It begins on 7 December 2025, with a single commit whose message read, simply, “Fresh repo.”

In the five months that followed, flightofthefox wrote 1,348 commits — very nearly alone — and by the time the funded work on Milestone 1 formally began in May, the hard core of a sharded blockchain was already standing: a working consensus engine, transactions executing across shards, a deterministic crash simulator, and mathematical proofs of the parts that are never allowed to break.

Consider this the “previously, on Hyperscale” — the ground every weekly digest is built on, told in the language of hyperscale.rs. From here, Week #1 picks up the thread.

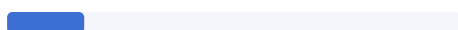
HOW IT BEGAN

Dec 2025

 434 commits

From “Fresh repo” to a running consensus core, cross-shard execution, and RocksDB-backed state. The skeleton went up fast.

Jan 2026

 77 commits

A quieter month spent on the deep plumbing — crash recovery, the peer sync protocol, node reputation. Fewer commits, harder problems.

Feb 2026

 42 commits

The design winter: formal models written in Quint, and the safety questions you think about far more than you type.

Mar 2026

 163 commits

The simulator finds its feet — partition tests, byte-for-byte replay, batched signature checks. Bugs start reproducing on demand.

Apr 2026

 450 commits

The big push. The busiest month yet, as every piece converges toward the start of Milestone 1.

1–10 May 2026

 182 commits

Milestone 1 — Adaptive Sharding — officially begins. The weekly digest starts here, at Week #1.

The foundations already standing

What was already built by the time the weekly digest began.

The Census

The beacon that tracks validators & topology

The registry of who's validating, how much they've staked, and which shard they sit on. It was among the first things to exist — you can't split the work until you know who's doing it.

The Generals

Cross-shard atomic commit — not two-phase commit

Making a transaction that touches two shards either fully happen or not at all — deliberately not the textbook approach. Cross-shard execution was wired into the runners in the very first weeks.

The Proof

Formal, mathematical verification

Where tests aren't enough, maths is. Quint specifications model the protocol's safety and liveness, so the core guarantees are proven — not merely observed to hold.

The Overlap

Provably fork-free

The guarantee that the chain can never split into two conflicting histories. View-change certificates and cross-shard verification landed early, because safety is the one thing you can't bolt on later.

The Crash Lab

Replay every bug byte-for-byte

A deterministic sandbox where any failure can be replayed exactly. Built so that when something breaks, flightofthefox can reproduce it on demand instead of guessing.

The Library

One giant binary merkle tree

All of global state held as a single binary merkle tree — the quiet trick that makes cleanly re-sharding the network possible at all.

WHAT'S IN THE BOX



That's the ground floor. Everything the weekly digest covers from here on — adaptive sharding, shards splitting and merging, validators reshuffling without dropping a transaction — is built on top of these foundations. Read on: Week #1 begins 11 May 2026.

The bigger picture

The full map to read along on hyperscale.rs, and the road ahead.

THE FULL PICTURE

The System

The whole sharded design at a glance.

The Census

The leaderless beacon of validators, stake and shards.

The Archive

Every published byte is preserved or provably expired.

The Lottery

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

The Crash Lab

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

The Journey

One transaction's trip through the network.

The Overlap

Why two conflicting blocks can never both commit.

The Library

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

The Triage

Under overload, urgent traffic never waits for bulk.

The Proof

Proving safety with maths, not just tests.

The Clock

Consensus-attested time across independent shards.

The Generals

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

The Will

How in-flight transactions settle when a shard dies.

The Governor

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

The Asterisks

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

THE ROAD TO MAINNET

M1

Adaptive Sharding
~4 mo

M2

Radix Engine
~5 mo

M3

Gateway & API
~3 mo

M4

Validator GUI
~3 mo

M5

Migration
~3 mo

M6

Live support
12 mo

Next up: Milestone 1 — Adaptive Sharding, beginning May 2026. Everything above is the ground it's built on — 1,348 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

[flighthofthefox](https://github.com/flighthofthefox)



scan for
hyperscale.rs

Questions? [flighthofthefox](https://t.me/hyperscale_rs) is always happy to discuss in the community Telegram — t.me/hyperscale_rs.