

• BUILD • TEST • DEPLOY • MONITOR

MeshForge

Mesh Network Operations Center
& Development Ecosystem



S H A K A

PRESENTER **WH6GXZ** / Nursedude

BUILT WITH **Dude AI** — Claude Code × Nursedude

REPO github.com/Nursedude/meshforge

Four things, four layers, four communities.

PHY

LoRa

Chirp Spread Spectrum · 902–928 MHz (ITU R2)

Sub-GHz physical layer. SF7–SF12 trades data rate for sensitivity (–137 dBm at SF12/BW125). Real-world: 0.3–27 kbps, line-of-sight miles, hours of duty-cycle airtime per day. **Not a network protocol — just radio.**

MAC/NET

Meshtastic

Firmware on LoRa modules · protobuf · managed flood
ESP32 / nRF52 firmware. Channels are PSK-encrypted protobuf packets, hop-limited flood routing. Phone app over BLE/serial; `meshtasticd` Linux daemon for SBCs. MQTT egress is built in.

TRANSPORT

Reticulum (RNS) + LXMF

Cryptographic transport · link-layer agnostic

Identities are Ed25519/X25519 keypairs; addresses are hashes of public keys. Runs over LoRa (RNode), TCP, serial, I2P. LXMF is the store-and-forward messaging layer. **Different abstraction than Meshtastic — transport, not firmware.**

MAC/NET

MeshCore / MeshAnchor

Newer LoRa firmware · deterministic routing

Newer entrant. Path-based routing instead of flood, companion-radio model, store-and-forward. MeshAnchor is my MeshCore-primary sister project — same architecture as MeshForge, different protocol primary.

*These do **not** interoperate. Different framing, different addressing, different assumptions about routing and trust.*

Meshtastic and MeshCore — different jobs, not a contest.

What you choose depends on what you're trying to do.

MESHTASTIC

since 2020 · firmware v2.x · GPLv3

SHAPE

Open-source LoRa mesh *firmware* for ESP32 / nRF52 / RP2040 boards. Text, location, telemetry over a flood mesh — every node can relay.

PRIMARY JOB Off-grid texting between people. The walkie-talkie metaphor.

FITS Hikers, festivals, neighbourhood chat, EmComm pre-deploy

DOESN'T Dense urban — flood collisions when node count climbs

CLIENTS Polished iOS / Android / Web apps · CLI · cloud MQTT relays

FLASH flasher.meshtastic.org · Heltec / RAK / LILYGO / T-Beam

MESHCORE

since 2024 · C++ library + firmware · MIT

SHAPE

A C++ *library* with reference firmware — devs build embedded apps on top. Multi-hop routing with explicit, fixed roles.

PRIMARY JOB Lightweight infra: repeaters, BBS-style rooms, sensor networks

FITS Operators planning fixed sites · companions *don't* relay (by design)

DOESN'T Replace Meshtastic socially — smaller user base, especially locally

CLIENTS iOS / Android / Web · Python & Node CLIs · KISS modem bridge

FLASH meshcore.io/flasher · roles: Companion / Repeater / Room / Sensor

On Hawai‘i — what’s actually on-air.

Maturity is a real feature. Here’s the field reality between protocols on the islands today.

MESHTASTIC

▫ established

- **Active nodes** on O‘ahu, Maui, Big Island — visible on meshmap, growing year-over-year.
- **Discord + club placements** — BIARC, Pacific mesh groups coordinating sites and channels.
- **EmComm interest** — real-world deployment exercises, not just hobby chat.
- **Ecosystem maturity** — you can buy a radio today, flash it from a browser, talk to neighbours by tonight.

*If you key up **today**, this is what you’ll hear.*

MESHCORE

▫ early

- **Handful of operators** experimenting with repeater + room-server roles.
- **Architecturally well-suited** to a chain of fixed sites — explicit roles, room-server rooms.
- **User base still building** — fewer field apps, less community knowledge yet.
- **Open question** — does it land here as a Meshtastic alternative, or as the backbone tier underneath?

The architecture fits the geography. The community has to catch up.

MESHFORGE STANCE

Both supported, equally. The TUI installs either. Run them on separate radios, see them on the same map, bridge selectively. **The protocol war isn’t ours — operability is.**

What MeshForge is — and is not.

IS

A **NOC**, not a radio. Runs **alongside** meshtasticd and rnsd — doesn't replace them.

A **composable bridge** between Meshtastic and Reticulum/LXMF over MQTT.

An **RF planning toolkit** — link budget, Fresnel zone, path loss, NOAA space weather. Standalone, no radio required.

A **live NOC** — browser map, both meshes, SNR-coded links, packet inspection.

Linux all the way down — systemd units, journalctl, standard CLIs.

IS NOT

A new mesh protocol. Doesn't reframe LoRa packets.

A replacement for the Meshtastic phone app.

A cloud service. No telemetry, no account.

An AREDN router. AREDN integration is read-only NOC overlay.

Production-stable. v0.5.6-beta — expect rough edges.

POSITIONING Wireshark + Grafana + a coverage planner, scoped to amateur LoRa.

ECOSYSTEM · THREE REPOS, ONE OPERATOR

CORE · THIS DECK

meshforge

Mesh Network Operations Center · TUI · gateway · RF tools

+

EXTENSION

meshing_around_meshforge

Bot framework integration · MockAPI for demo mode · MQTT decryption bridge

+

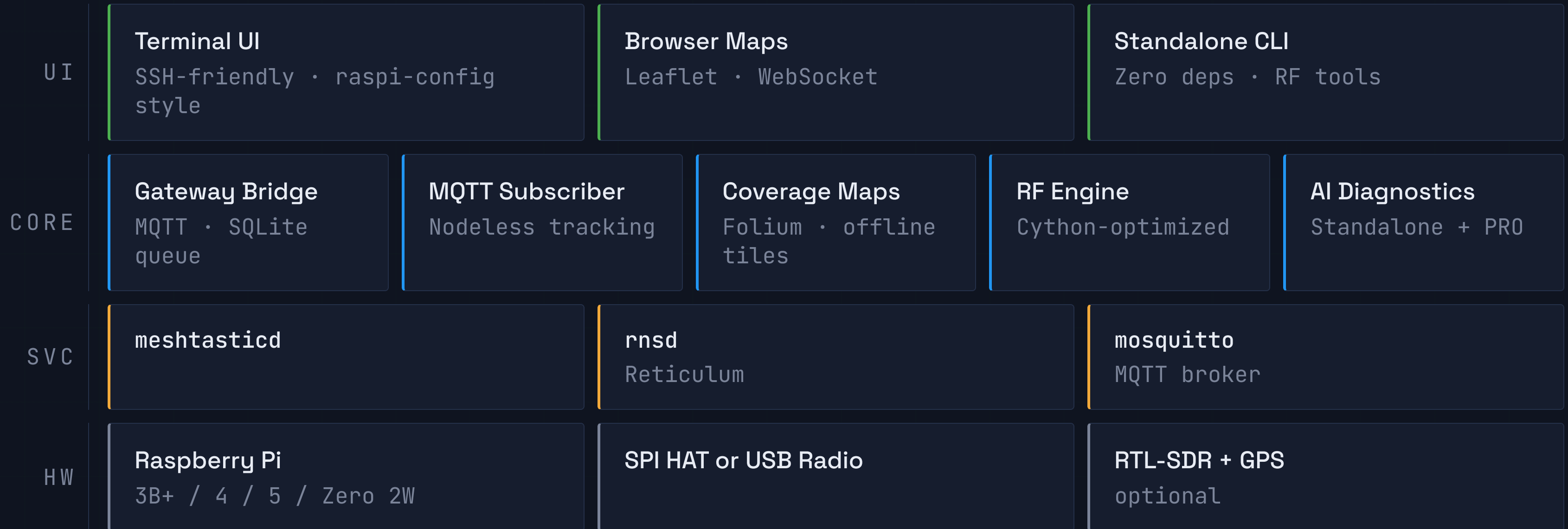
EXTENSION · :8808

meshforge-maps

Multi-source map · Meshtastic + AREDN + MeshCore + MQTT + RNS on one canvas

Installed from the TUI as separate systemd units. Each repo owns one job — no duplicated state, no shared writers. *Compose, don't merge.*

Four layers. Standard Linux underneath.



systemctl · journalctl · meshtastic CLI · rnstatus — nothing exotic.

Three independent bridges. One process. Refusal on inconsistency.

`rns_bridge_enabled`

RNS ↔ Meshtastic

message bridge · LXMF over MQTT

```
Mesh channel meshforge ⇌ LXMF
inbox in NomadNet
```

FLEET DEFAULT · `moc3`

`mesh_bridge.enabled`

Meshtastic ↔ Meshtastic

cross-preset relay (LongFast ↔ ShortTurbo)

```
HAT preset A ⇌ USB radio preset B
CODE LANDED · not yet field-
live
```

`rns_transport.enabled`

RNS over Meshtastic

RNS uses LoRa frames as link layer

```
RNSMeshtasticInterface ⇌
meshtasticd
```

EXPERIMENTAL

A MESSAGE IN FLIGHT · MESH → RNS · MOC3

▶ [click any hop](#) · or [⚠ what breaks](#)

Mesh → RNS · forward

RNS → Mesh · reverse

[⚠ what breaks?](#)

RF
SHORT_TURBO
903.625 MHz · SF7



DAEMON
meshtasticd
SPI HAT · :4403



BUS
mosquitto
:1883 · multi-sub



CORE
meshforge-
gateway
classify · echo-
filter · queue



LXMF
rnsd
shared · rpc_key
pinned



RNS
RNode +
AutoIface
LoRa-RNS · TCP
fleet

protobuf MeshPacket

JSON on msh/#

CanonicalMessage

LXMF over RNS link

RNS packet / announce

END-TO-END BUDGET ~250 - 700 ms typical · *RF-airtime dominated* · processing < 25 ms total · queue persistence adds 0 ms in steady state

ECHO FILTER

```
if msg.text.startswith("[RNS:") and tag in self._own_tags: drop()
```

Every cross-bridge TX is tagged with the destination hash; gateway holds a sliding-window set and drops re-ingest before classification. **This line prevents the double-bridge loop.**

REFUSAL

two bridges on same radio → exit 2.

QUEUE

SQLite-backed; survives restarts.

SOAKED

`moc3` · 5 boxes · weeks live.

A bridge isn't composable just because it bridges.

"BRIDGE"

– L2 networking

A device that connects two segments.

Monolithic. One bridge, one path.

"COMPOSABLE"

– data center, software-defined

Resources **decoupled** from hardware, requested on demand without disturbing what's running.

MESHFORGE

– borrows the second, on a Pi

The MQTT bus **is** the software layer. Bridges become subscribers, not chokepoints.

= orthogonal flags + bus fan-out + config-time refusal

AXIS 01 · ORTHOGONAL MODES

Three flags. Any subset.

`rns_bridge_enabled` — Mesh ↔ LXMF

`mesh_bridge.enabled` — preset ↔ preset

`rns_transport.enabled` — RNS over LoRa

$2^3 = 8$ valid configs, one gateway.

AXIS 02 · BUS, NOT SOCKET

A new consumer doesn't touch the gateway.



One subscription away.

AXIS 03 · REFUSAL AS CONTRACT

Wrong configs caught at startup.

```
if bridges_overlap(cfg):  
    log.error("ambiguous radio")  
    sys.exit(2)
```

Exit 2 is the contract.

FIELD PROOF · moc3 All three axes live · five subscribers, one bus · phone & CLI still work · weeks of soak, zero double-bridge events.

The decision that unlocked everything else. TCP gave one client. We needed five.

01

THE WALL

meshtasticd's TCP API on :4403 accepts *exactly one* client. Bridge connects → phone disconnected. Phone connects → CLI fails. Inspector starves. Coverage map starves.

02

FALSE STARTS

Connection-pooling proxy — a shim in front of :4403 that fanned out to N consumers. Worked in tests; bridge state diverged in the field. **Fork meshtasticd** — would lose upstream forever. **Polling HTTP scrape** — drops live events, can't drive a bridge.

03

THE FLIP

"meshtasticd already speaks MQTT — it's been there the whole time." Don't proxy the radio's TCP socket. Make the radio *publish*. Let any number of consumers *subscribe*. The bus was the answer, not a smarter client.

04

THE PATTERN

meshtasticd → mosquitto. Every MeshForge component is now a *subscriber* on the same bus. TCP:4403 is freed for the CLI. The gateway, monitor, traffic inspector, coverage map, Grafana exporter, decryption bridge — all coexist. **Zero interference, by construction.**

BEFORE · v0.5.3 TCP bridge — one winner

meshtasticd
TCP :4403 · 1 client

Bridge ✓ holds it Phone ✗ CLI ✗ Inspector ✗

Whoever grabs the socket wins. Everyone else is locked out — silently.

v0.5.4+ MQTT bus — everyone subscribes

mosquitto
MQTT :1883 · ∞ subs

Bridge ✓ Subscriber ✓ Inspector ✓ Coverage ✓
Grafana ✓ Decryptor ✓

Multi-consumer by construction. TCP:4403 stays free for the CLI.

WHAT THIS UNLOCKED

Persistent SQLite queue — queue is the source of truth. **Composable bridges** — three modes coexist on one radio.

Refusal on inconsistency — config-time contract, not socket race.

Phone & CLI still work — meshtastic --info live.

What works. What doesn't. What's mid-flight.

● WORKS

Gateway bridge — stable on 5 boxes, several weeks soak

MQTT pipeline — SQLite-queued, refusal on inconsistency

RF library — link budget, Fresnel, path loss; standalone

Coverage maps — Folium, offline tiles, SNR-coded

TUI installer — SPI/USB/GPS/SDR auto-detect

Diagnostic rule engine — offline, 20-topic KB

● IN PROGRESS

Live NOC map — real GPS, not mocked feeds

MQTT decryption — partial; channel-PSK pathway

Mesh alert engine — thresholds, throttling, sinks

uConsole profile — hardware ships Q2 2026

Historical playback — SQLite → timeline scrub

● NOT YET

AREDN write — read-only overlay only

MeshCore native — lives in MeshAnchor sister repo

Firmware flashing — v1.0 stretch, not started

RTL-SDR spectrum — v0.6/v0.7 target

Production stability — it's beta. Expect breakage.

\$35 floor. Real RF ceiling.

FLOOR

~\$35

Pi Zero 2W + USB radio

Heltec V3/V4 · CDC

LILYGO T-Beam S3 · GPS

RAK4631 · nRF52840

MeshStick · official

RECOMMENDED

~\$90

Pi 4 / 5 + SPI HAT

MeshAdv-Pi · 33 dBm (1 W)

Ebyte E22-900M30S · 30 dBm

Waveshare SX126X

RAK2287 · WisBlock

FIELD UNIT

Q2 2026

uConsole AIO V2

CM5 8 GB

SX1262 LoRa · 22 dBm

RTL-SDR · 100 kHz-1.74 GHz

Multi-constellation GPS

Auto-detect: SPI, USB, GPS, SDR. No editing config.yaml by hand.

— FOR THE LICENSED OPERATOR · CORE TUI · ALL SHIPPED

What's in it for a HAM.

Not a chat app. A radio operator's NOC.

PROPAGATION & WX

NOAA SWPC Space Wx

Live Kp, Bz, solar wind, X-ray flux

Local Wx + EAS/iPAWS

NWS alerts pushed on-mesh

Fresnel + Path Loss

Terrain-aware, point-to-point

RF PLANNING

Link Budget

TX · RX · antenna · feedline · loss

Site Planner

Coverage prediction, terrain

Freq Slot Manager

Region-2 LoRa channels

LIVE OPERATIONS

Live NOC Map

Multi-protocol, SNR-coded links

Topology Graph

Who-hears-who, real-time

Traffic Inspector

Packet-level, MQTT tail

DIAGNOSTICS & FIELD

Dashboard

Service health, data-path check

RTL-SDR

Spectrum view, signal capture

SOS Beacon

One-keystroke field activation

One TUI, no tab juggling. Twelve tiles, six-key main menu — Dashboard, Mesh Networks, RF & SDR, Maps & Viz, Configuration, Emergency. SSH in over a slow link, drive everything from the keyboard.

— ON THE ISLANDS · 6 DAYS · ONE LISTENER

This is what one Pi heard last week over Hawai'i.

Six days. One LongFast listener. The whole island chain.



6.0

DAYS · APR 28 → MAY 4

16,235

PACKETS

606

NODES PINNED · 788 SEEN

231





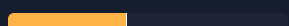
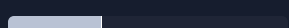
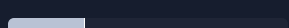

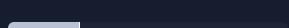
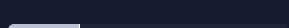
UNIQUE SOURCES

Who's on the air. Who's relaying the islands.

6-day window · LongFast · Apr 28 → May 4 2026 UTC · real long_names from the capture.

MOST ACTIVE · BY PACKETS RECEIVED











top 10 of 231 unique sources

01	WS85 Ho'okipa !75695fb4		1,080
02	GTR B-Base !18cc57c0		559
03	Makawao e255 !1b1de255		556
04	KH7EH-04 !a038f60c		514
05	Solar monolith beta !79be01d3		458
06	solar 482f !f57d482f		355
07	Haleakala Summit Repeater !6d9ee8c7		289
08	WH6EPS-[] !a2ed9124		287
09	Hale Pohaku alpha 195f !c34e195f		266
10	Meshtastic bc60 !a2e1bc60		265

Weather stations and solar nodes beacon constantly — that's why they top the list. The repeaters show up further down because they're *relaying*, not just *announcing*.

BACKBONE REPEATERS · ROUTER / ROUTER_LATE

top 10 of 27 backbone-class nodes

01	Haleakala Summit Repeater !6d9ee8c7		289
02	WH6EPS-[] !a2ed9124		287
03	Hale Pohaku alpha 195f !c34e195f		266
04	Keaalau Repeater !d4b8113d		258
05	Pepeekao BIARC !16cd7438		163
06	Pi4-30dB-hoku deck !012fb668		155
07	Elab router pi4-30dB ba5a !32c7ba5a		155
08	Pi4-30dB-whha !ddd0c892		154
09	Wailuku Repeater (MQTT) !a2eb419c		152
10	Pi400-30dB !731553de		145

● **Pepeekao BIARC is in this room's backbone.** 27 backbone nodes across the chain — 16 ROUTER, 11 ROUTER_LATE. Per-hop relay counts would need live packet capture; this is ranked by activity within the role.

It's mostly not text. Here's why your battery drains.

Six days, 16,235 packets, broken down by portnum and hour.



6,920
POSITION

4,599
NODEINFO

3,442
TELEMETRY

1,274
TRACEROUTE

The fleet is real. Deliberately heterogeneous — a regression class can't hide.

volcanoai

CANONICAL
WRITER

Code, configs, memory.
The only box that
commits.

```
LongFast
```

```
QTH · short name  
vail
```

moc

LONGFAST PEER

RF peer for volcanoai —
preset-A regression
catcher.

```
LongFast
```

```
Pi 4 · LongFast  
HAT primary
```

moc1

TEST BED

Full stack. Maps, Prom,
Grafana — observability
host.

```
:8808 · :9090 ·  
:3000
```

```
Pi 5 · field-  
validates  
dashboards
```

moc2

CROSS-PRESET QA

Different modem preset
—
can *not* peer with LF
over RF.

```
SHORT_TURBO
```

```
The point: half  
the fleet  
disagrees
```

moc3

CANONICAL
GATEWAY

The Meshtastic ↔ RNS
bridge.
Real LXMF, real
LongFast, real soak.

```
SHORT_TURBO +  
RNode 903.625 MHz
```

```
First open-source  
NOC unifying both
```

QA THESIS

Five jobs, two modem presets, two ecosystems on one bus. **Identical boxes would hide whole bug classes.** A regression that survives volcanoai → moc → moc1 → moc2 → moc3 on heterogeneous radios has no good place left to hide.

3,229

TESTS · 81 FILES

Run on volcanoai before sync

~45 s

PUSH → FLEET ON NEW
CODE

```
scripts/fleet_sync.sh
```

2 / 2

MODEM PRESETS IN SOAK

LongFast · SHORT_TURBO

v0.6.0-β

CURRENT · COMPOSABLE
BRIDGES

293 → 3,229 tests in 8 weeks

Fleet status, one URL.

The operator's first SSH on any box. Truth state across six peers.

localhost:5000/fleet meshanchor-server · MeshForge NOC v0.6.0-β

SELF · THIS BOX meshanchor-server · 192.168.86.x · uptime 14d 06:42

CALLSIGN **WH6GXZ**

RADIO **MeshCore · BW62 / SF7**

FREQ **910.525 MHz**

RNS IFACE **tcp · 4242 · UP**

MQTT **● broker · 1.4k msg/h**

SERVICES **4 / 4 required up**

FLEET ROLLUP · 6 PEERS 5 running · 1 deliberately disabled

HOST	ROLE	SERVICES	LAST SEEN	STATUS
volcanoai:5000	canonical writer	meshtasticd · rnsd · gateway · mosq	2s ago	● OK
moc:5000	longfast peer	meshtasticd · rnsd · gateway · mosq	4s ago	● OK
moc1:5000	test bed · obs	maps :8808 · prom · grafana	1s ago	● OK
moc2:5000	cross-preset QA	meshtasticd · rnsd · gateway · mosq	3s ago	● OK
moc3:5000	canonical gateway	Connection refused · Pi-Zero host	—	● DISABLED
meshanchor-server:5000	self · meshcore	meshanchord · rnsd · maps · mosq	now	● OK

FEDERATION · RNS ANNOUNCES · 24H 20 hashes · LXMf + meshforge_node app_data

a1f3 c9e2 ...	meshforge volcanoai nomad	00:01:42
7d4b 2118 ...	meshforge moc nomad	00:03:08
e8c0 5af9 ...	meshforge moc1 nomad	00:04:55
3b91 0d67 ...	meshforge moc2 nomad	00:07:21
c25d 84e1 ...	meshanchor server lxm	00:09:03
+ 15 more	scroll for the full announce table	—

6

PEERS · 5 RUNNING · 1 DISABLED

20

RNS ANNOUNCES · 24H WINDOW

BW62 / SF7

MESHCORE · 910.525 MHz

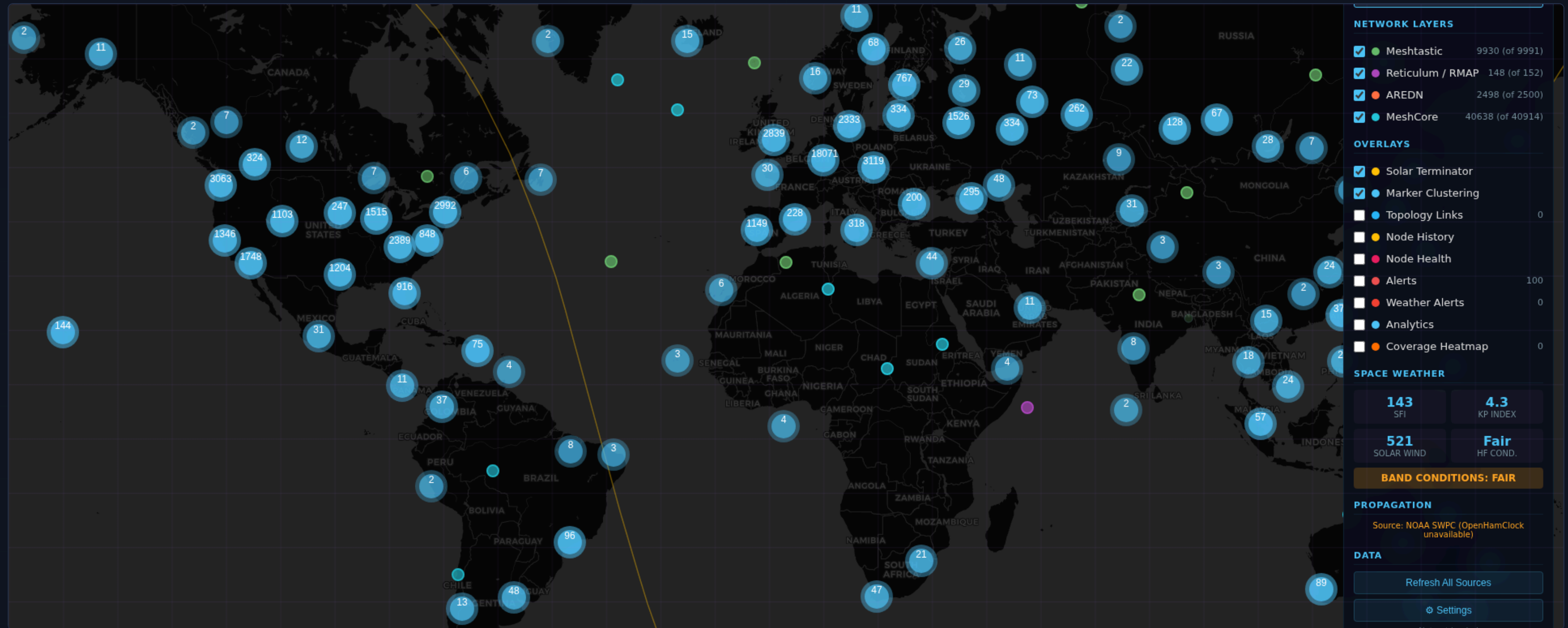
4 / 4

REQUIRED SERVICES UP

Failures are signal, not noise. The deliberate disable on moc3 shows as *Connection refused* — that's the network telling me my topology decision is still in effect.

This is what's on the screen.

Live web map · Meshtastic + RNS + AREDN + MeshCore · NOAA space weather · NWS alert ribbon.



MESHTASTIC
24,625
of 76,991 directory
total

MESHCORE
48,282
cyan clusters · largest
segment

SPACE WX
SFI 143 · Kp 4.3
HF cond: Fair · NOAA
SWPC

RNS / AREDN
1,378 · 2,706
RMAP + AREDN nodes

Memory joins the fleet. The agent that built the system is now part of it.

THE RECURSIVE GAP

Six weeks of pair-debugging lived in `~/ .claude/memory/` on whichever box the session started on. Five fleet boxes, **five divergent memories**. The fleet had been observable for a week. The agent had not.

01

RSYNC MIRROR

Canonical writer, four read-only replicas.

`fleet_sync.sh` pushes both memory dirs from `volcanoai` to every fleet host — `rsync -aq --delete --mkpath`. Bidirectional sync was the obvious move; we refused it.

Pick one writer, replicas pull only.

95015d9 · Tier 1

02

SECRETS GATE + GITHUB

Two private repos. A pre-commit hook with a high-confidence pattern table.

PEM keys, AWS creds, GitHub PATs, Slack OAuth, bcrypt, `rpc_key ≥ 32-hex`.

Contextual, not entropy-based — memory is *full* of LXMf / RNS / commit hashes that look random and are fine. The signal is a key-ish name *adjacent to* a key-ish value.

726b77b · Tier 2

03

AUTO-COMMIT ON SYNC

Memory rides the same rails as code.

Every `fleet_sync.sh` begins with `git add -A && commit && push` on the canonical box, then `rsync` delivers `.git/` to all five replicas. **One operator command updates code, services, AND the agent's working memory.**

598440d · Tier 3

WHY THIS ISN'T LIKE THE OTHER PATTERNS

Files, not embeddings.

Exact recall. Recall `project_gateway_fleet_state.md` by filename — not the seventh-nearest neighbor.

Operator-paced, not background.

A bad memory written at 11:43 can be deleted at 11:44. Never reaches the fleet.

Single canonical writer.

No CRDT, no merge logic. `--delete` enforces it the way a foreign key enforces referential integrity.

Secrets gate at the writer.

Not at replication. A leaked credential is caught on the box where it was typed — before it lands on four others and a private remote.

✓ Live since 2026-04-30. `claude-memory-global` + `claude-memory-meshforge` — next session, any box: **already loaded.**

Nursedude + AI, every commit on the record.

THE COLLABORATION

The repo is the artifact.

MeshForge is not just code that **uses AI** — it is a record of **what AI-paired engineering actually looks like** across a real, hardware-in-the-loop project. Every dead-end, every refactor, every “why did we ever try that” is in the git log.

The **Dude AI white paper** in [docs/](#) is the methodology. The **Substack** is the journal. The **field logs** are the receipts.

What worked: tight loops, ruthless tests, AI as second pair of eyes on RF math. What didn't: trusting first-pass diagrams, skipping hardware soak. Both are documented.

~3,200

TESTS

RF math, gateway, parsers

5

PIS IN FIELD

Soak, not simulation

1

OPERATOR

Nursedude · WH6GXZ

GPL-3.0

LICENSE

Open, forkable, auditable

“The failures are part of the deliverable. If you can't show the bad commits, you didn't actually build it.”

— working principle

MeshForge is a domain, not a repo.

CORE HUB · NOC

Nursedude/meshforge

Gateway bridge · TUI · RF tools · Diagnostics · Service mgmt

v0.6.0-β

Python 3.9+ · GPL-3.0

VIZ · PLUGIN

meshforge-maps

Multi-source live map. Leaflet + D3 + WebSocket. Standalone *or* plugin.

:8808 / :8809 v0.7.0-β

BOT ALERTS

meshing_around_meshforge

12 alert types — emergency, EAS, volcano, proximity. Sits beside the meshing-around bot.

843 tests v0.6.0

SISTER · MESHCORE

meshanchor

Same DNA, different home radio. **MeshCore**-primary NOC. Extracted 2026-04-01.

3,226 tests v0.1.0-α

RNS INSTALLER

RNS-Management-Tool

Cross-platform installer for the RNS stack. Linux + **Windows 11**. 21+ RNODE boards.

Bash + PowerShell v0.3.2

Rule: Satellites may depend on the NOC. The NOC **never** depends on satellites. It discovers plugins via `manifest.json` at startup — and runs fine without any of them.

Three repos. Three jobs. One domain.

EXTENSION · 01

meshforge-maps

"the live, multi-source map"

Aggregates **Meshtastic**, **Reticulum/RMAP**, **AREDN**, **MeshCore**, and **NOAA/OpenHamClock** into one Leaflet web map. Live MQTT, topology graph, per-node health (0–100), threshold alerts, offline tile cache, curses TUI dashboard.

STACK	Python · Leaflet · D3 · WS
SOURCES	5 collectors
TESTS	1,047
MODE	standalone <i>or</i> plugin

ALERTS · 02

meshing_around_meshforge

"alert layer for the bot"

Companion to SpudGunMan/meshing-around. **10-screen TUI** + **51 mesh commands** + **12 alert types** (emergency, iPAWS/EAS, volcano, weather, proximity, battery, SNR, disconnect...). Hosts the canonical `global.ini` schema.

STACK	Python · Rich · paho-mqtt
CHANNELS	DM · Email · SMS · Sound
TESTS	843
PI 2W	no-radio MQTT validated

SISTER · 03

meshanchor

"same DNA, MeshCore home radio"

Extracted from MeshForge 7e4fa02 on **2026-04-01**. **MeshCore primary**; Meshtastic + RNS as optional gateways. **64-hop routing** (vs Meshtastic's 7), flood-free topology. Shares CanonicalMessage contract.

STACK	Python 3.10+ · meshcore_py
PROFILES	5 (meshcore default)
TESTS	2,729 (mocks)
FIELD	needs you · a

The split that matters: MeshForge treats Meshtastic as the home radio with MeshCore as a gateway. MeshAnchor flips it. Same TUI, same bridge, same RF tools — CanonicalMessage is the contract that keeps both compatible.

Five repos. One config. One direction of dependency.

01 The shared identity file

```
~/config/meshforge/global.ini
```

Set MQTT broker, region preset, callsign, and home coordinates **once** — Maps and NOC read it as a fallback before their per-app configs. Hosted in `meshing_around_meshforge`; consumed everywhere. Set **WH6GXZ** in one place; the whole fleet knows.

02 Dependency direction

- ▶ `meshforge-maps` depends on `meshforge` (optional)
- ▶ `meshing_around` depends on `SpudGunMan/meshing-around`
- ▶ `RNS-Mgmt-Tool` independent — patterns flow *in*
- ↔ `meshanchor` sister · CanonicalMessage contract
- ! The NOC never depends on satellites.

BRIDGE LOGIC

Protocol routing →
`meshforge/gateway/`

VIZ + TOPO

Live maps, health → `meshforge-maps`

BOT ALERTS

Emergency, EAS →
`meshing_around_meshforge`

RNS INSTALL

`rnsd, NomadNet` → `RNS-Management-Tool`

✓ **The contract holds.** Maps auto-discovers via `manifest.json`; NOC connects to services it never embeds; CanonicalMessage bridges Meshtastic ↔ RNS ↔ MeshCore across both NOCs. Five repos, one operator UX.

Where it goes from here.

NOW · v0.5.x

Stability + field validation

Gateway bridge soak

Live NOC with real GPS

MQTT decryption

Mesh alert engine

NEXT · v0.6 /
v0.7

Visibility + portability

Historical playback

RTL-SDR spectrum

GPS + GPX export

uConsole AIO V2

LATER · v0.8
→ v1.0

Ecosystem + hardware

MeshForge ↔ MeshAnchor bridge

Hardware support matrix

Firmware flashing

v1.0 stable

By the time you read this, this slide is obsolete.

Two months. One repo became five. The slope is the product.

FEB 2026 · WEEK 0

"I have an idea."

Pi + Meshtastic + a hunch
meshforge created
CLI scaffolding, first bridge sketch

MAR 2026 · THE PIVOT

The bridge becomes composable.

v0.5.4 · MQTT pivot
~100 hrs of design Q&A
moc3 proves three axes live

APR 2026 · CRYSTALLIZE

Domain, not repo.

4/01 · MeshAnchor extracted
4/30 · MOC Memory deployed
5 repos · Hawai'i fleet live

THE 90° CLAIM

output × time



solo dev × Claude

≈ vertical

5

REPOS

From 1 → domain in 8 weeks

8,345

TESTS

Across all five repos

~60

DAYS

Idea → 5-Pi Hawai'i fleet

1

OPERATOR

Hobbyist · RN · ham · solo

The reframe: this isn't "I built a thing with AI." It's what AI-paired engineering actually looks like at full tilt — a hobbyist NOC, a sister project, a memory pattern, a deck — all in the time it used to take to scope a feature.

The curve, in writing → wh6gxznursedude.substack.com

by next month, this slide will need a sequel.

Three ways in.

①

Try the RF tools

```
python3 src/standalone.py
```

No install. No radio. No sudo.

②

Star · file issues · PR

```
github.com/Nursedude/meshforge
```

Field-test logs especially welcome.

③

The living history

```
substack: nursedude · docs/white-paper-ai-dev-partner.md
```

Every commit, every dead end, every fix.

This deck was already wrong by Wednesday.

That's the actual point.

RENDERED

2026-05-16

Frozen here. Test totals, fleet count, MeshCore packets
— true on this date.

LIVE · NOW

2026-05-16

Recomputed every time this slide loads.

Δ · SINCE RENDER

10h

commit count loading...
github.com/Nursedude/meshforge

The deck is a snapshot. The repo is the artifact.

If a number on a slide disagrees with github.com/Nursedude/meshforge, trust the repo. This deck was true the day it was rendered, and that's the strongest claim a same-day snapshot can honestly make.

CANONICAL · SOURCE

github.com/Nursedude/meshforge

FIELD · LOGS

wh6gxznursedude.substack.com

LIVE · NUMBERS

docs/birc_2026_05_17_live_numbers.md