MeshCore Governance and Community

MeshCore is an open-source project with a distributed community and a governance structure that changed significantly in April 2026. Understanding the project landscape helps you navigate firmware choices and community resources.

The April 2026 governance split

In April 2026, the MeshCore project underwent a governance transition:

- **Core team repository:** github.com/meshcore-dev/MeshCore - maintained by the core team. MeshCore was created by **Scott (Ripple Radios / ripplebiz)**, the project founder and lead firmware engineer; the wider core team includes Liam Cottle (app), Recrof (map/flasher), FDLamotte (Python/STM32), and Oltaco (bootloader). This is the canonical firmware for the full hardware range, the source for the official MeshCore app, and what most regional networks (CascadiaMesh, WCMesh, RegionMesh, NoDakMesh) recommend for interoperability.
- **Andy Kirby's fork (MeshOS):** meshcore.co.uk - a third-party community fork optimized for standalone keyboard devices (T-Deck, T-Deck Plus). Andy Kirby is a former contributor/promoter who controls the original .co.uk domain and Discord, but is not the project founder. Includes the MeshOS firmware and associated tooling.

Both projects share the same underlying protocol and are interoperable on the radio link. A node running MeshCore core team firmware and a node running MeshOS can communicate over the air. The split is about firmware features, hardware focus, and development direction - not protocol compatibility.

Which firmware should you use?

Scenario	Recommended firmware
Standard repeater or router node (Heltec, RAK4631, T-Echo, T-Beam)	Core team (github.com/meshcore-dev/MeshCore)

Scenario	Recommended firmware
T-Deck or T-Deck Plus standalone keyboard device	MeshOS (meshcore.co.uk) for best feature set; core team firmware also works
Joining a regional network (CascadiaMesh, RegionMesh, etc.)	Core team (most regional networks recommend core-team firmware for interoperability; check the specific network's docs)
Ultra-low power ESP32 optimization	EasySkyMesh (IoTThinks community fork) for additional ESP32 power-saving options

Community resources

Resource	URL	For
Core firmware source	github.com/meshcore-dev/MeshCore	Source code, issues, releases
MeshCore web flasher	flasher.meshcore.io	Flash firmware without local tooling
Web configuration	config.meshcore.io	Configure nodes via browser
MeshOS (Andy's fork)	meshcore.co.uk	Third-party community T-Deck standalone firmware
Python tooling (CLI)	github.com/fdlamotte/meshcore-cli	Python CLI/API for automation
CascadiaMesh (PNW)	cascadiamesh.org	Pacific Northwest community
WCMesh (West Coast Mesh)	wcmesh.com	West Coast network
RegionMesh (Central US)	regionmesh.com	Central US communities
NoDakMesh (Northern Plains)	nodakmesh.org	North Dakota & region

Contributing to MeshCore

The project welcomes contributions in several forms:

- **Bug reports:** If your device behaves unexpectedly, open an issue on GitHub with firmware version, hardware, and steps to reproduce. Detailed bug reports are the most immediately useful contribution.
- **Hardware compatibility:** Testing new devices and reporting what works (or doesn't) helps the project support more hardware. Especially valuable for less-common boards.
- **Documentation:** This wiki is the community's primary documentation resource. See the Contributing section for how to improve it.

- **Code:** Use the standard GitHub PR workflow. Open an issue first for significant changes to discuss the approach before writing code.

Revision #5
Created 2026-05-03 03:55:57 UTC by Mesh America Admin
Updated 2026-06-09 14:38:25 UTC by Mesh America Admin