

NexaCore OS — Backgrounder

Extended background for journalists. Safe to quote or paraphrase. Last updated: 2026-07-08.

One line: An independent open-source project exploring what an operating system should be in the age of AI agents: AI-native, local-first, and decentralized, with inference and intelligent agents as kernel-level primitives, running on hardware you own.

Why now

Today's operating systems were designed before modern AI. They treat inference and agents as applications bolted on top of a cloud service, which means the intelligence in your daily computing runs on someone else's servers, under someone else's terms. NexaCore OS starts from a different premise: if AI is now a primitive of computing, it belongs in the kernel, next to processes and files — and it belongs on hardware the user controls.

What it is

NexaCore OS is an independent open-source project. It treats inference, model orchestration, and intelligent agents as kernel-level primitives rather than cloud-backed add-ons. It runs locally by default. When it needs more compute it federates over a peer-to-peer mesh of cryptographically-attested nodes, rather than commercial cloud APIs. Privacy is enforced by the protocol, not by policy: the guarantees are structural, checked cryptographically, not promised in a terms-of-service document.

How it works

- **Microkernel in Rust.** The kernel is written entirely in Rust with a frozen system-call ABI and a Ring 0 / Ring 3 split. A small, auditable trusted base.
- **Capability model.** Access is granted by unforgeable capabilities, not ambient authority. A component can touch only what it was explicitly handed.
- **Five system agents.** Orchestrator, Guidance, System Administrator,

Security & Performance, and Task — each capability-isolated, so an agent's reach is bounded by the capabilities it holds.

- **ncScript (NCIP)**. A capability-gated scripting language for automating the system without handing scripts unrestricted power.
- **Federated mesh**. Compute scales outward — local first, then a personal cluster, then a federated mesh of attested peers, with commercial cloud as a last resort rather than a default.

Openness

The codebase is Apache-2.0 and the protocol specifications are public domain. The project's openness does not depend on any single organisation: forks are first-class citizens by design. The long-term stewardship and governance structure will be announced as the project matures.

Status

Phase 1 — a microkernel proof of concept that boots and runs under QEMU, Proxmox and VirtualBox (bare-metal bring-up on physical machines in progress), released as v0.3.0-alpha.2. At this stage the numbers that describe the codebase are: 86 crates, 7,700+ tests, 70 frozen syscalls, 100% Rust. This is early, pre-1.0 software: it demonstrates the architecture, it is not yet a daily-use operating system.

Founded by

Founded by Matteo Sala.

Links

- Website: <https://nexacoreos.com>
- Technology deep dive: <https://nexacoreos.com/technology.html>
- ncScript manual: <https://nexacoreos.com/manual.html>
- Built-in agents: <https://nexacoreos.com/agents.html>
- Devlog: <https://nexacoreos.com/devlog.html>

Media contact

press@nexacoreos.com — we aim to respond within two business days.