

## MLCP Development Strategy Canvas — Coordinated Build Plan

**Objective** To deploy the Meta-layer Coordination Protocol (MLCP) in progressive, open-source phases—balancing experimental vibe coding, secure engineering, community integration, and ethical governance. This canvas outlines the development trajectory, contributor structure, and tactical sequencing needed to launch and sustain the MLCP substrate.

---

**PHASE 0: Substrate Design & Framing (Foundational)** - Define MLCP as an extension of MCP (Model Context Protocol) for interface-level coordination. - Design MCP interface schema: overlays, presence, consent, tags, bridges, agents. - Establish identity primitives: DID, presence roles, consent tokens. - Draft visual overlays, tag flows, and minimal trusted consent stack. - Draft context capsules and manifest prompts for AI/vibe coders.

**Actors:** Core design team, architects, AI-aligned frontend prototypers

---

**PHASE 1: Reference Overlay & Runtime Stack (Minimal Viable Substrate)** - Launch reference overlay app with: - Presence engine - Tagging system - Consent prompts - Identity indicators - Implement MLCP API surface (modular hooks, invocation schema) - Log overlay sessions, traffic metadata (URL+hash), interaction trails - Anchor actor profiles on Holochain with private web trail ledger - Scaffold federated + wallet-based auth module

**Actors:** AI copilots, frontend devs, backend substrate engineers, UI/UX mappers

---

**PHASE 1.5: Data Sovereignty + Community Plugins (Ethical/Scalable Layer)** - Add Data Sovereignty module (revocation, export, access control UI) - Launch Plugin Registry for overlays, curators, validators - Enable native tokens for overlay unlocks, tips, and agent payouts - Build first semantic filter builder (community-safe context filters) - Establish starter meta-community templates (governance, reputation)

**Actors:** Web3 ecosystem partners, governance prototypers, token designers, consent auditors

---

**PHASE 2: Distributed Deployment & Reflexive Governance** - Enable overlays to run as decentralized nodes or browser extensions - Incorporate runtime governance via composable stacks (agent rulesets, policy layers) - Integrate runtime moderation & reflexive observatories (containment tools) - Support multi-domain trust paths, decentralized hosting (e.g., Holochain, IPFS)

**Actors:** Hackathon contributors, civic-tech groups, federated infra operators, moderation system designers

---

**Cross-Cutting Tactics - Vibe Coders:** Activate early to design rituals, flows, symbols, UI feeling tones - **Hackathons:** Focused on filters, semantic bridges, meta-presence visualizations - **Token Incentives:** Earn tokens via overlay contributions, moderation, validators - **AI Co-devs:** Use AI agents as copilots for

scaffolding, testing, documentation - **Secure Dev:** Build in containment assumptions, enforce TEE-based consent execution

**Long-Term Outcome:** A composable, extensible trust substrate for the Metaweb that is actor-sovereign, agent-coordinated, and community-enforceable above any webpage or domain.