UBS ZK

SPACE NETWORK

ZK

UBS is a sovereign blockchain for zero-knowledge storage mining. No GPUs. No stake. Just provable space and deterministic time.

This document is for informational purposes only.
No public offering. No guaranteed token allocation.

Read Full Architecture Doc
Linktr.ee/UBSZK

UBS ZK Space Network

Live ZK Blockchain for Storage Mining

Real Mining. Real Privacy. Real Chain.
Modular · Cross-Chain · Verifiable
Hybrid consensus, Bitcoin-compatible, Sybil-resistant

Why This Matters Now

- For Enterprises: Replace costly cloud storage ($20+/TB/year) with earn-as-you-store infrastructure
- For Miners: 1TB HDD = full participation (no ASICs/staking)
- Requires 85% less infrastructure cost than Filecoin
- For Developers: Native ZKID and private DAOs - GDPR-ready by design

Key Differentiators

Traditional Solutions	UBS ZK Space
❌ AWS/GCP: Cost center ($20+/TB)	✅ Protocol efficiency benchmark: 12–14% (based on modeled resource utilization)
❌ Bitcoin: 150 TWh/year waste	✅ $3/TB/year energy cost
❌ Filecoin: No privacy	✅ L1 ZK-proofs (no rollups)

Key Differentiators

Consensus	Privacy Layer	Stack Infrastructure
• Proof-of-Space	• OpenZKP at L1	• GTK4 GUI (30MB)
• VDF Time-Lock	• ZKID & ZKDAO	• Nix / Docker builds

Traction

We're not launching ideas. We're launching production-ready infrastructure.

Proven Stack
50+ nodes testnet V1 completed,
GUI/CLI suite, ZK browser extension.
Patent-pending PoSpace + VDF consensus,
Substrate runtime, native OpenZKP engine.

Built-In Confidentiality
ZKID (zk-ID for DeFi)
Private DAOs, and proof compression at the protocol level.

Green Mining
No staking, no GPUs. Just 1 TB of disk space and uptime.

Coming:
Testnet V2
Mainnet
Storage DATA Marketplace

Get Involved linktr.ee/UBSZK

- Ask the UBS AI Assistant
- Read Full Architecture Doc
- Pre-Seed Memo & Token Terms

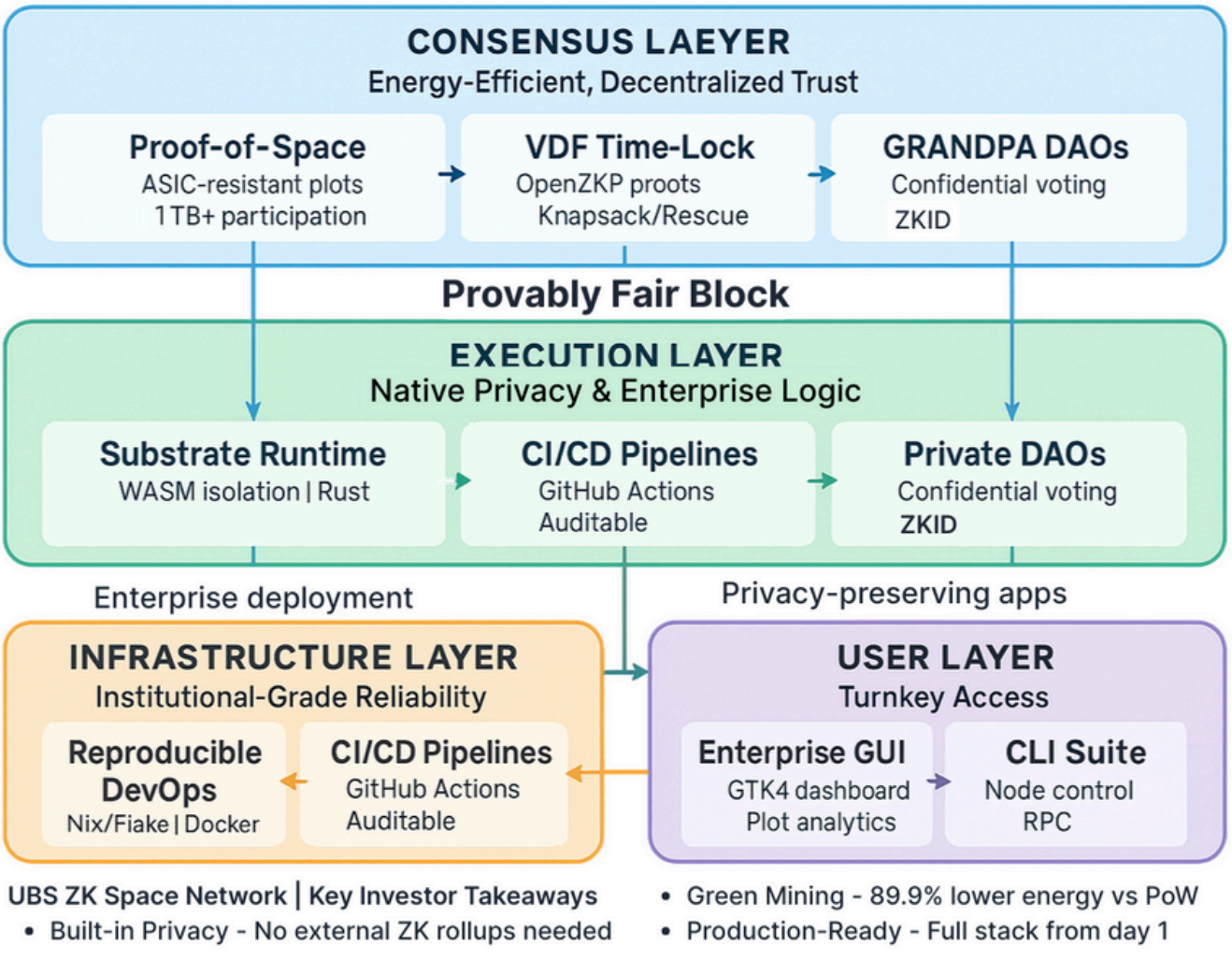
UBS ZK Space Architecture

UBS is engineered as a modular, verifiable, and reproducible system. Each layer is isolated, cryptographically sound, and deployable independently or as a unified vertical stack.

Core Stack Layers:

1. **Consensus Layer**: Proof-of-Space + VDF + GRANDPA Finality
2. **Execution Layer**: Substrate Runtime with native ZK support
3. **Infrastructure Layer**: Reproducible DevOps (Nix, Docker), CI/CD, telemetry
4. **Interface Layer**: GTK4 Desktop GUI, CLI Suite, Web3 Extension

The architecture is built to scale — from local farming nodes to integration with multi-chain ecosystems like Cosmos IBC, Celestia DA, and ZK-Rollup frameworks.



The general architecture scheme of the UBS ZK Space Network, which includes the consensus, execution, infrastructure, and user layers.

Consensus:
PoSpace + VDF + GRANDPA Finality

At the core of UBS lies a three-phase consensus mechanism that eliminates the need for staking, expensive hardware, or centralized validators. The system ensures honest participation using time, space, and verifiable computation.

Proof-of-Space (PoSpace)

Instead of tokens or compute power, UBS uses disk space as the resource of truth. Farmers pre-generate plot files (chiapos v2.3) and participate by allocating real storage.

- Low entry threshold: from 1 TB
- Near-zero energy usage (no GPU/ASIC load)
- Geographically distributed farming infrastructure

Verifiable Delay Function (VDF)

Once a valid quality string is found in a plot, the farmer must compute a time-locked VDF proof using haraka-rs. This ensures that winners are determined fairly and deterministically, regardless of hardware speed.

- Eliminates race conditions and front-running
- Levels the playing field between home users and data centers
- Guarantees time honesty in block selection

GRANDPA Finality

UBS finalizes blocks using the BFT-based GRANDPA protocol, ensuring deterministic finality and enabling slashing conditions for validator misbehavior.

💡 UBS does not require tokens to mine, nor megawatts to participate.
Just storage — and provable time.

ZK Execution Layer: Privacy by Design

While most chains add privacy as a patch or via rollups, UBS ZK Space was built from the ground up as a ZK-native blockchain — where privacy is a foundation, not an option.

OpenZKP Runtime (Integrated)

Instead of external ZK-SDKs, UBS embeds a modular proof engine directly in the runtime layer, leveraging Knapsack, Rescue, and Haraka hash functions.

- Supports ZK-DAO, ZKID, and confidential governance
- Compresses and aggregates SNARK proofs without touching private data
- Enables full input masking and preimage resistance

Private DAOs & ZK Governance

DAOs can operate without ever exposing participant identity. Votes, memberships, and proposals are executed via zk-proofs with zero leakage.

- ZKID at protocol level
- Ready for jurisdictional compliance without data exposure
- Fully compatible with DeFi, KYC-linked use cases

DevOps & GUI Infrastructure: Built for Real Deployment

Most blockchains promise decentralization but deliver little beyond a CLI and a node binary. UBS ZK Space comes with a battle-tested, reproducible DevOps stack and intuitive GUI interfaces - ready for mainstream deployment.

Reproducible Infrastructure

- Nix / Flake: fully reproducible environments
- Multistage Docker builds: cross-platform compatibility
- GitHub Actions: CI/CD pipelines, automated linting & security checks
- Build caching: instant onboarding for new nodes and contributors

User-Facing Interfaces

- GTK4 GUI: lightweight desktop client for Windows/Linux/macOS (30 MB, zero-dependency)
- CLI Suite: node control, plot inspection, RPC access
- Web3 Extension: zk-proof signing and dApp integration

UBS isn't built for engineers only - it's built for real users. From plot farmers to enterprise node operators, the experience is streamlined, fast, and production-ready.

What Makes UBS ZK Space Unique

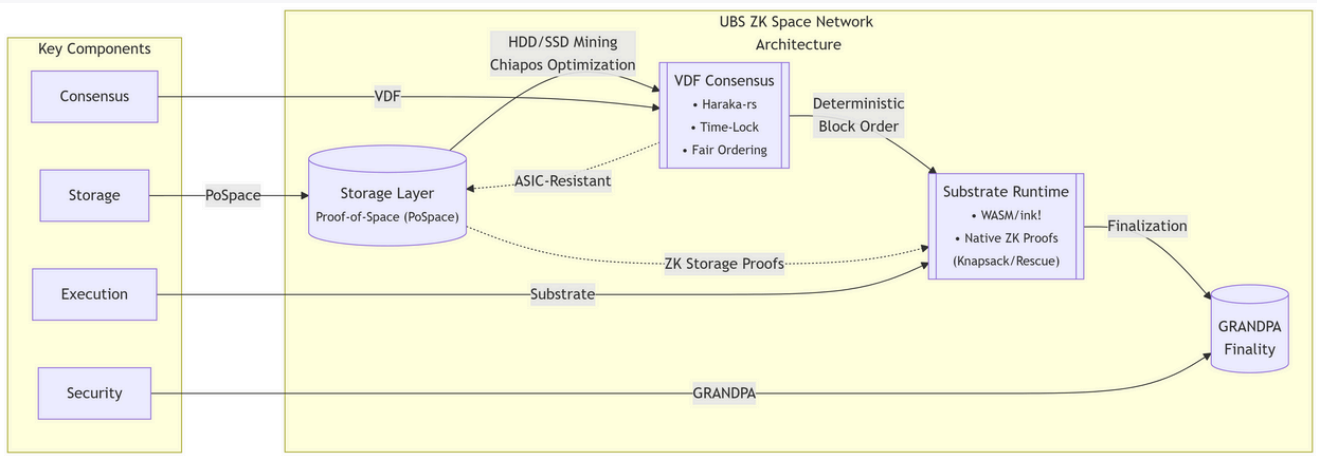
Most blockchain storage protocols suffer from three systemic issues:

- ❌ No privacy
- ❌ No scalability
- ❌ No real UX or deployability

Some chains layer complex L2s, rollups, or external tooling in an attempt to fix this. UBS took a radically different path: architectural honesty + verifiable design at L1.

UBS ZK is not "yet another PoSpace chain". It's:

- ✅ A production-grade infrastructure, not a prototype
- ✅ Privacy-native at L1, not bolted on via rollups
- ✅ Stake-free, accessible, and inclusive by design
- ✅ Engineered for both institutions and home users



How UBS ZK Space Compares

🆚 Chia, Filecoin, Arweave:

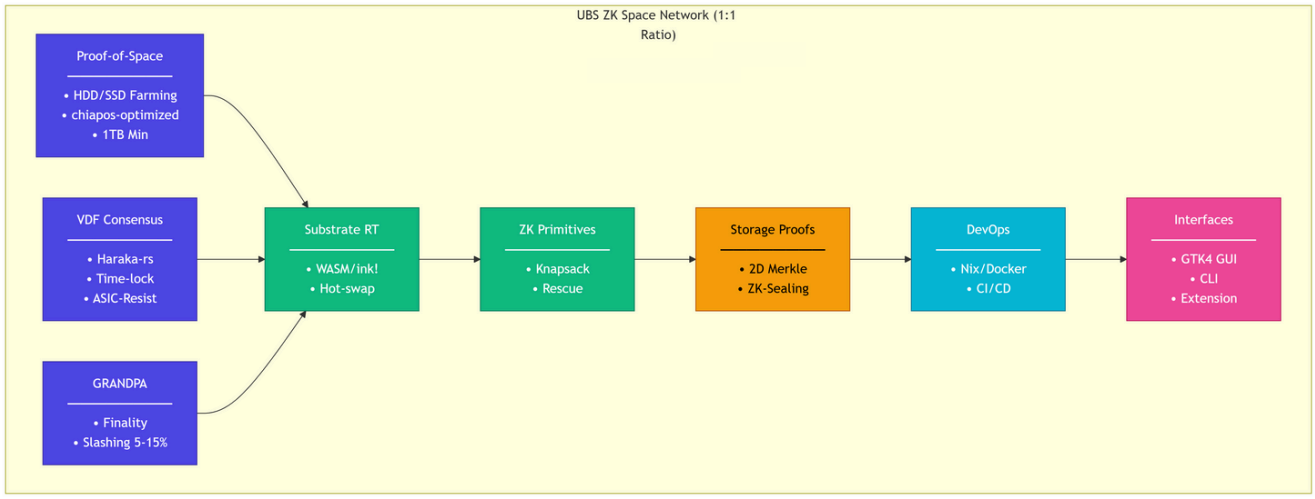
Disclaimer: Comparison for technical benchmarking only. Not a financial or service offering.

Project	Consensus	Privacy	Stack Maturity	UX	Monetization Model
Chia	PoSpace	❌ None	Testnet → Mainnet	❌ CLI only	❌ Mining only
Filecoin	PoRep + PoSt	❌ None	Complex infra stack	⚠️ Web3 but heavy	✅ Storage provider payments
Arweave	PoA (permaweb)	❌ None	IPFS-like layer	⚠️ Limited UX	❌ Flat-rate burn
UBS ZK	PoSpace + VDF	✅ Native	L1 Fullstack	✅ GUI + CLI + Ext	✅ Mining Rewards + Storage Market

Key Differentiators:

- ✅ Native ZK privacy in runtime (not an add-on)
- ✅ Low entry barrier: start with just 1 TB, no stake
- ✅ NFT-based storage plots: own, rent, or resell
- ✅ Full reproducible DevOps stack: GUI + monitoring + auto-build
- ✅ Fairness via VDF: eliminates resource dominance and centralization

UBS ZK isn't competing with Filecoin — it's redefining the model: storage-as-yield, privacy-by-default, fairness-as-consensus.



Built-In ZK Integration

We don't use rollups.
We don’t depend on zk-as-a-service providers.
We’ve embedded zero-knowledge logic directly into the protocol.

OpenZKP Engine

This is a ZK engine like in Zcash. It works on HDD and SSD.

- SNARK-friendly hashes: Knapsack, Rescue, Haraka
- ZK proofs applied at transaction layer
- Full input masking & extended preimage protection
- On-chain proof compression without data exposure

ZKID & Private DAOs

- Every user receives a zk-identity
- DAOs function anonymously and provably
- Voting & participation use zk-attestations, not wallets

2D ZK Storage Map

- Farmers are assigned unique storage quadrants
- They prove storage without revealing its contents
- Data maps are aggregated into SNARKs, usable for zk-voting, zk-hosting, zk-access layers

Modular ZK Layer

- Compatible with Nova / PlonK / Stark curves
- Path to universal trusted setup
- Parameters & ceremony logs published on IPFS / Arweave

UBS ZK doesn’t support privacy — it’s built around it.

The network is verifiable by design, economically sound, and **natively engineered for high-throughput ZK applications.**

Tokenomics

UBS ZK Space is built on the principles of energy efficiency, real infrastructure rewards, and anti-centralization.

Unlike PoS networks, UBS doesn't require capital to participate — just disk space and uptime. The economic model is designed for long-term sustainability, fairness, and self-sufficiency.

Protocol token issuance per block

- **Year 1:** 5 UBS per block
- **Emission curve:** -10% every ~2M blocks (~12 months)
- **Duration:** rewards phase lasts 7 years
- **Minimum reward:** 0.1 UBS/block to protect small farms
- **Total planned emission:** ~900M UBS

Post-Emission Fee Model

Storage Fee: 0.001 UBS / GB·month

- 60% → farmers
- 20% → validators
- 20% → community-governed protocol fund (non-custodial)

Transaction Fees:

- 50% → farmers
- 30% → validators
- 20% → treasury
- 10% of every fee is burned

Participation Rules & Economic Fairness

Minimum Entry

- 1 TB of plot space (chiapos v2.3)
- No staking or token locking required

Resource compensation algorithm for Farmers

Base algorithm + $k \cdot \log_2(TB)$

- APR = (Block Reward + Storage Fees) / (Total Staked Space)
- After 1 PB, the k factor decays exponentially to limit centralization

Modeled resource compensation rate (Year 1)

- Modeled storage efficiency gains: 12–14% (non-financial metric)
- Validator activity metrics may vary based on network size and participation.

Validators

- Finalize blocks via GRANDPA
- Earn transaction fees + burn bonuses
- **Slashing rules:**

- Offline >24h → -5% stake
- Second violation → -10%
- Third → -15%

Anti-Sybil & Reputation Layer

UBS implements an identity-less, ZK-secured mechanism to eliminate fake farms and Sybil attacks:

- Each farm seals its plots under a private key → unique zk-proof
- Participation is tied to a long-lived reputation ID, based on uptime, block correctness, and history
- Trust coefficient influences reward adjustments
- Participation is anonymous yet provable — no identity, no simulation

NFT-Based Storage & DAO Marketplace

Plot-NFTs

Every unit of disk space can be tokenized as an NFT:

- Represents rights to store data
- Can be sold, leased, delegated, or transferred
- Tied to size, duration, and rarity

Cloud Pools

- Supports renting out plots via third-party providers
- Integrates with GUI for Filecoin, AWS, and custom clouds

DAO Storage Market

- DAO governs storage slot availability
- Users acquire storage via NFT-bound contracts
- Fully open-market pricing model

Modeled resource rate (Year 1)

- Supports renting out plots via third-party providers
- Integrates with GUI for Filecoin, AWS, and custom clouds

Product Roadmap

From Testnet to Interchain ZK Infrastructure

UBS ZK Space is being rolled out in **staged phases**, each designed to deliver real infrastructure not vaporware. We don't ship MVPs or half-finished tooling. We focus on **layer maturity, interoperability, and usability from day one.**

Phase 1: Testnet & Onboarding (Q3 2025)

"Technology, not just a token"

- Launch of Testnet V2 (Substrate + chiapos + VDF)
- 50+ live nodes (farmers + observers)
- GUI and ZK component validation
- Integration with external cloud providers (Filecoin, S3-compatible)
- 80% code coverage on Rust/WASM stack
- Early partnerships Infra projects

Phase 2: Mainnet Launch (Q4 2025-2026)

"A self-hosted, user-first chain with real-world tooling"

- Public launch of mainnet
- Embedded OpenZKP engine (Knapsack, Rescue, Haraka)
- On-chain storage NFT marketplace
- GUI-integrated leasing via DAO-market
- zkID and private DAOs live at L1
- Protocol tokens exist solely as mining rewards, with no planned distributions (if community-initiated and community-driven)
- Public node repo + Dev SDK
- Launch of ecosystem grants program
- Target: 500+ mainnet nodes by EOY 2026

Contribution terms, token economics, and milestone triggers are detailed in the official Pre-Seed Memo.

Phase 3: Ecosystem Expansion & ZK Interop (2026+)

"UBS becomes the backbone of sovereign, privacy-first infrastructure"

- Integration with Cosmos IBC / Celestia DA
- zk-Rollup modules (Nova, PlonK, Stark → WASM bridge)
- zk-Governance, zk-Lending, DAO-native data tooling
- Global distributed farming network (individual + institutional)
- ZK Bridge Layer for zkSync, StarkNet, Polygon zkEVM
- Cross-chain NFT marketplace: own storage across chains

Conclusion

A New Paradigm of Decentralized Storage Protocol

UBS ZK Space Network is not a competitor to cloud. It's an alternative to centralized infrastructure itself.

In a world where privacy has become illusion and decentralization is a marketing label, UBS returns control to users — through green consensus, provable fairness, and L1-native zero-knowledge logic.

- ❌ No rollup dependencies
- ❌ No staking oligarchy
- ❌ No hidden reliance on Big Tech

- ✅ Just provable space.
- ✅ Deterministic time.
- ✅ Validator-independent architecture.

UBS ZK Space is a Zero-Knowledge Fabric for decentralized storage — built for real infrastructure, not promises.
No token sale. No staking. No permission required.
If you're aligned with our mission and want to support early development, 🔗
you may review the Pre-Seed Contribution Memo (off-chain, non-binding).

Additional Resources

Explore Docs

📄 **Full Docs & Allocation Info → linktr.ee/UBSZK**

🔗 **Full Technical Specification**

🔗 **Pre-Seed Terms & Token Framework**

🔗 **UBS AI Assistant (Live Q&A)**

UBS ZK Space is not just a protocol. It's the infrastructure layer for decentralized, yield-generating storage.

Glossary UBS ZK Space Network

PoSpace (Proof of Space)
A consensus mechanism that requires participants to allocate disk space instead of compute power. Used to prove physical storage commitment.

VDF (Verifiable Delay Function)
A cryptographic function that requires a fixed amount of sequential time to compute but is fast to verify. Ensures time-based fairness in block generation.

ZK (Zero-Knowledge)
A family of cryptographic techniques allowing proofs of knowledge or computation without revealing the underlying data. Core to privacy and scalability in UBS.

Trusted Setup
A one-time, secure ceremony to generate public parameters for some types of ZK proofs (e.g., SNARKs). Required for Groth16.

Universal Setup
A reusable setup (e.g., PLONK, Sonic) that doesn't require a new ceremony per circuit. UBS plans to migrate here post-Mainnet.

Groth16 / PLONK / Sonic
ZK proof systems. Groth16 is efficient but needs a trusted setup. PLONK and Sonic support universal setup.

Plot / Plotting
A reserved and structured space on disk that is used in PoSpace consensus. "Plotting" refers to generating this space.

Farmer (Storage Node)
A participant that dedicates storage space and takes part in consensus by proving their plots.

Light Node
A node that validates blocks and consensus without storing the full chain. In UBS, light nodes may still prove storage and verify ZK proofs.

Runtime (Substrate)
The on-chain logic of a blockchain built on Substrate. Handles execution, state transitions, and pallets.

WASM (WebAssembly)
A portable binary instruction format. UBS executes smart contracts in an isolated WASM environment for safety.

Chain Spec
A JSON file defining the genesis state and configuration of a Substrate-based chain.

Glossary UBS ZK Space Network

GRANDPA
Finality gadget used in Substrate. Provides fast, deterministic block finality.

DevOps
Development operations — tools and scripts for deploying, monitoring, and scaling infrastructure.

Testnet
A non-production version of the blockchain used for experimentation and validation. UBS will launch multiple testnets.

ZK-SNARK
Zero-Knowledge Succinct Non-Interactive Argument of Knowledge — a popular type of ZK proof.

FFI (Foreign Function Interface)
A way to integrate external code (e.g., C libraries or DLLs) into Rust/Substrate environments. Used for integrating VDFs or GPU hashing in UBS.

DSL (Domain-Specific Language)
A programming language tailored to a specific domain. ZK projects often use DSLs for circuits (e.g., Circom, Leo).

CLI
Command Line Interface — used to interact with the node or wallet via terminal commands.

GTK (GIMP Toolkit)
A library for creating native GUI applications, used in UBS's desktop farmer client.

Docker / Nix
Tools for reproducible builds and deployments. UBS supports both for cross-platform setup.

Node Operator
A user or organization that runs a UBS full or light node. Can participate in farming, consensus, or data availability.

UBS does not facilitate token trading or custody. All mined tokens are subject to local AML laws when converted