White Paper: Decentralized MLOps Governance on Forte

1. Abstract

This paper presents a decentralized MLOps governance platform built on **Forte's tooling**, leveraging its **Rules Engine** and **O2 Oracle API** to manage ML model lifecycle decisions in a secure, transparent, and automated way. The system enables performance metrics and metadata from cloud ML pipelines to be recorded on-chain. Smart contract rules—defined in Forte's Rules Engine—evaluate these metrics to determine model promotion or deprecation. These governance decisions are immutably logged on-chain and can trigger downstream automation. The platform also includes a model registry and monetization layer, empowering developers to share or commercialize ML models with full auditability and access control.

2. Problem Statement

MLOps today is hindered by centralized control, limited transparency, and fragmented tooling. These issues manifest in:

- Poor auditability of model lifecycle decisions and events
- Risky centralized governance mechanisms
- Minimal monetization paths for niche model developers

As AI becomes more critical to business and society, there is a need for robust, transparent, and decentralized governance of machine learning systems.

3. Proposed Solution

We propose a decentralized MLOps platform built atop **Forte**, where model lifecycle decisions are governed algorithmically through the **Rules Engine**, and interaction with off-chain ML pipelines is managed via the **O2 Oracle API**.

3.1 Rules Engine for Smart Governance

Forte's **Rules Engine** replaces traditional smart contracts with a policy-driven logic layer. Rules are defined to evaluate incoming model metrics (e.g., accuracy, precision, data drift scores) and compare them against predefined thresholds. These rules automatically determine:

- Whether a model should be promoted or deprecated

- Whether retraining is required
- Whether the model remains eligible for monetization

Decision outputs (e.g., Pass/Fail flags) are recorded as on-chain events.

3.2 O2 Oracle API for Metric Ingestion

The **O2 Oracle API** is used to securely transmit model performance metrics and metadata from cloud-based ML pipelines to the Forte blockchain. This API supports:

- **Metric submission** (e.g., validation accuracy, latency, drift)
- **Metadata logging** (e.g., training data info, model version, timestamps)
- **Decision flag retrieval**, allowing pipelines to react to governance outcomes

The Oracle layer ensures integrity and synchronization between ML environments and the blockchain.

3.3 Decentralized Model Registry

Every model submitted is registered on-chain, with entries controlled via policy. This registry:

- Tracks model metadata, version history, and decision outcomes
- Allows developers to publish models as open or private
- Creates an immutable audit trail across the full ML lifecycle

3.4 ML Universe & Monetization Layer

Models published as open source can participate in a **per-inference monetization system**. Users pay for model access using Forte's native cryptocurrency, rewarding developers for their contributions. This model supports general-purpose foundational models and niche applications such as:

- Domain-specific language models
- Specialized object detection models
- Open-source ML primitives

3.5 Permissioning & Access Control

Through Forte's policy framework, access to model data and events is fully customizable:

- Private metrics can be restricted to specific users or organizations
- Public flags (e.g., model approved/rejected) remain available for transparency
- Token-based or ACL-based controls can manage who sees and uses model information
- Encrypted metadata or external references can support privacy-sensitive use cases

3.6 ML Models as Real-World Assets (RWAs)

In the ML Universe, each trained machine learning model is treated as a **Real-World Asset (RWA)**—a tokenized digital representation of valuable intellectual property (IP). By anchoring these assets on-chain with governance metadata, access policies, and monetization flags, the platform:

- Provides **verifiable ownership and version history**
- Enables **secure IP rights management**
- Facilitates **on-chain licensing and inference-level revenue models**

This framing aligns the platform with broader trends in **RWA tokenization**, bridging the world of data science with decentralized finance and digital asset infrastructure.

4. Architecture Overview

- **MLOps Tasks (Cloud, local scripts)**: Train, monitor and promote models, then submit metrics and metadata via SDK
- **Python SDK**: Provides a simple interface for data scientists to send metrics through the O2 Oracle API
- **Forte O2 Oracle API**: Transfers data between ML systems and the blockchain
- **Forte Rules Engine**: Applies governance policies to on-chain model metrics
- **Blockchain Layer**: Stores model events, decisions, and registry entries
- **Consumers**: Access models via inference APIs and pay via native token

5. Use Cases

- **Enterprise AI Auditability**: Maintain compliance and track model decisions on-chain
- **Open-Source Model Monetization**: Developers earn from usage of shared models
- **AI Fairness and Transparency**: Expose and verify model evaluation logic
- **Federated LLM Management**: Track promotion and lifecycle of multiple evolving LLMs

6. Benefits

- **Built on Forte**: Utilizes powerful native tooling like the Rules Engine and O2 Oracles
- **Transparent Governance**: All decisions are algorithmic and recorded on-chain
- **Developer-Friendly SDK**: Data scientists don’t need blockchain expertise
- **Privacy-Aware**: Supports both open and private model governance
- **Economic Incentives**: Monetization aligned with contribution and usage
- **Models as Assets**: Treats ML models as IP-backed digital assets (RWAs)

7. Roadmap

- **Phase 1**: Build Forte Rules Engine logic and O2 Oracle-based metric ingestion
- **Phase 2**: Develop a user-friendly Python SDK that wraps O2 Oracle API calls for easy ML pipeline integration
- **Phase 3**: Develop Decentralized Model Registry
- **Phase 4**: Launch ML Universe monetization layer using a derivative of Forte's native token (wrapped token)

8. Platform Foundation: EVM Compatibility

The decentralized MLOps platform is **EVM-based**, ensuring compatibility with the Ethereum Virtual Machine and its broader ecosystem. Smart contract execution via **Forte's Rules Engine** occurs within this framework, offering:

- Interoperability with existing EVM tools and wallets
- Familiarity for developers in the Ethereum ecosystem
- Seamless integration with DeFi primitives for future extensions

9. Future Extensions: DeFi + AI

With an EVM foundation, the platform can evolve into a more dynamic economic layer for AI via DeFi primitives:

- **Staking**: Developers or users can stake tokens on a model's performance; stakes are slashed if models fail predefined thresholds, enforcing quality.
- **Insurance**: Models can be covered by decentralized insurance smart contracts, compensating users if models underperform or introduce risk.
- **Performance Bonds**: Developers can lock tokens in escrow, released only if models maintain strong performance over time.

These mechanisms tie **financial incentives** directly to **model reliability**, enabling trustless, market-driven ML ecosystems.

10. Conclusion

This platform brings together MLOps, decentralization, and Forte's blockchain tooling to create an automated, secure, and monetizable AI governance framework. By embedding lifecycle rules and metrics into the blockchain, it fosters transparency, incentivizes innovation, and makes machine learning infrastructure more trustworthy for all stakeholders.