The World Asset Engine

Development Track: Gaming & Entertainment

Engine-agnostic, AI-assisted cross-game asset embodiment

1. The Problem

Digital game assets are locked inside individual game engines and worlds.
Existing interoperability solutions only work between compatible games—those that share similar engines, asset pipelines, or gameplay logic.

This fragmentation prevents:

- true asset ownership by players
- cross-game identity persistence
- scalable creator economies

Current approaches fail because they attempt to move assets directly, rather than interpret how an asset should exist *natively* inside a new world...

Graeme put the SU

2. The Proposal

I propose The World Asset Engine:
an engine-agnostic, AI-assisted kernel that enables a single canonical asset identity to be natively embodied inside any participating game world.

Rather than forcing direct asset conversion, the kernel:

- verifies ownership of a canonical asset
- interprets how that asset should manifest within a specific game's art style and logic
- authorizes the game to unlock a locally defined embodiment of that asset

Each game remains fully in control of its rendering, balance, and aesthetics.

3. Engine Architecture

The system consists of four layers:

1. <u>Canonical Asset Layer (Blockchain)</u>

- NFT representing a globally unique asset identity
- Stores ownership and asset ID only
- No rendering, physics, or gameplay data stored on-chain

2. <u>Kernel (World Asset Engine)</u>

- Engine-agnostic authorization and interpretation layer
- Verifies asset ownership
- Determines which world-specific embodiment applies
- Issues permission for in-game asset activation

3. <u>AI Translator (Assistive Layer)</u>

- Interprets the relationship between:
    - canonical asset identity
    - game world logic and art style

- Selects or recommends a world-appropriate embodiment
- Designed to scale toward automated interpretation

4. <u>In-Game API Interface</u>

- Lightweight client inside each game
- Requests authorization from the kernel
- Unlocks or denies access to pre-existing local assets

4. Technology Stack

- Blockchain: Polygon
- Smart Contracts: ERC-721 NFTs (canonical asset identity)
- Kernel: Python / C# / C++
- AI Component: Transformer-based sequence-to-sequence model (Python)
- In-Game API Interface: Python / C# / C++

5. MVP Scope (Hackathon)

Given the limited timeframe, the MVP will demonstrate:

- A single canonical NFT asset
- Two incompatible open-source games
- Each game contains its own native embodiment of the asset
- The asset is locked by default
- The kernel verifies NFT ownership and authorizes unlocking
- Each game renders the asset according to its own world logic

This proves engine-agnostic interoperability without requiring shared pipelines.

6. MVP Approach

Training a full AI sequence-to-sequence model would require large datasets and extended time.

Instead, for the MVP:

- World-specific embodiments of the asset will be manually designed
- The kernel will simulate AI behaviour by selecting the correct embodiment via predefined mappings
- The authorization and selection process will mimic how an AI-assisted system would function at scale

This approach demonstrates the architecture and feasibility of the system without overengineering.

7. Innovation & Potential Impact

The World Asset Engine introduces:

- true cross-game asset ownership
- native world-dependent embodiments
- engine-agnostic interoperability
- a scalable kernel-based approach rather than brittle asset conversion

This lays the foundation for:

- interoperable game economies
- persistent player identity across worlds
- a creator ecosystem unconstrained by engines

With widespread adoption, this model could unlock an entirely new class of digital asset economies at massive scale, expanding upon a gaming industry which is valued at over billions of dollars already.

This project focuses not on moving assets between games, but on interpreting how an asset should exist within each world — preserving the creative sovereignty of producers while enabling true ownership.