



# MODULE: Enabling the First Composable Multi-Agent System Tokens with ERC-AI

Ødinson  
module-ai.pro

## Abstract

MODULE introduces a new experimental framework for tokenized intelligence and intelligent liquidity: **Composable Multi-Agent System Tokens**, powered by a novel token standard and execution primitive called **ERC-AI**. Each MODULE token represents a fully autonomous, programmable AI agent that is composable with other agents to form powerful onchain swarms. By combining the fungibility of ERC-20 and the identity of ERC-721 with the logic of ERC-6551, ERC-AI enables a modular swarm of intelligent agents that can reason, act, and evolve together. This whitepaper presents the architecture, design rationale, and compositional logic of MODULE and ERC-AI, laying the foundation for a new category of decentralized AI systems where tokens are not only assets but intelligent actors.

## 1 Introduction

The emergence of autonomous agents has shifted the way we think about decentralized systems. While tokens have historically represented static units of value or identity, MODULE redefines tokens as **modular intelligent actors** capable of composing themselves into systems, swarms, and networks.

MODULE pioneers **ERC-AI**, a proposed AI token standard (AITS) blending ERC-20 fungibility and ERC-721 identity with ERC-6551's programmable state. This fusion enables each token to operate as an onchain, composable agent, going beyond a simple asset role. Each agent features its own wallet, state, and logic, accessible via a built in creation and interaction interface. The agent's logic, assets, and state remain onchain for security and transparency, while the AI inference layer executes offchain. This structure supports scalability, flexibility, and integration with powerful AI models.

As the base layer for onchain multi-agent systems, MODULE tokens enable the formation of intelligent swarms. These entities can then coordinate, govern, or execute tasks autonomously.

## 2 Technical Overview

### 2.1 ERC-AI: Composable Agent Token Standard

ERC-AI defines a multi-layered architecture:

- **ERC-20 Layer:** Each MODULE token exists as a fungible unit onchain and can be freely traded, staked, or bridged.
- **ERC-721 Layer:** Each MODULE token also has a unique non-fungible identity, enabling per-token ownership, metadata, and agent-specific logic.
- **ERC-6551 Layer:** Each token is paired with a token-bound account (TBA) that contains its logic, state, assets, and history.

**Mathematical Representation:** Let  $M$  be a MODULE token, with components:

$$M = M_{ERC20} \cup M_{ERC721} \cup M_{ERC6551}$$

Where:

- $M_{ERC20}$ : Fungible tradable token
- $M_{ERC721}$ : Non-fungible identity
- $M_{ERC6551}$ : Account-bound identity with smart logic and composability

**Total Functional State:**

$$F(M) = L + A + S$$

Where:

- $L$ : Logic executed by the agent
- $A$ : Assets held in the token-bound account
- $S$ : State/history of the agent

### 2.2 Composability: Swarm Intelligence

MODULE tokens are not standalone agents—they are designed to **compose** multi-agent systems.

Let  $S$  represent a swarm of agents:

$$S = \{M_1, M_2, \dots, M_n\}$$

Swarm behavior emerges when:

- Agents share logic  $L$
- Communicate state  $S$
- Interact with shared or individual assets  $A$

These modular agents can be programmed to:

- Act independently or as part of a collective
- Form dynamic coalitions
- Govern themselves or other systems
- Share rewards, state, and actions

## 2.3 Token-Scoped Reasoning

Each MODULE token contains its own programmable logic engine. This allows for **token-scoped reasoning**:

- Decision-making at the agent level
- Local execution of logic
- Autonomous response to onchain events

This makes MODULE tokens suitable for:

- DeFi bots
- Governance delegates
- Validator agents
- Copy-trade agents
- RWA managers

## 2.4 Recursive Composition

Just as ERC-6551 enables accounts to own NFTs, MODULE enables agents to own **other agents**.

**Recursive Composition:** Let  $M_1$  be a MODULE token that owns  $M_2, M_3, \dots, M_n$ :

$$M_{16551} = \{M_2, M_3, \dots, M_n\}$$

This allows for:

- Agent hierarchies
- Swarm controllers
- Nested governance
- Inherited logic and data

## 3 Applications and Use Cases

### 3.1 Autonomous DeFi Agents

- **Core Idea:** Create sophisticated, self-running DeFi participants that execute adaptive strategies.
- **How it Works:**
  - **Token-Scoped Reasoning:** Agents make complex decisions based on market data, internal logic (e.g., risk settings), and their own performance history.
  - **ERC-6551 Account (TBA):** Each agent holds and manages its own assets directly within its token-bound account.
  - **Composability (Swarms):** Agents can form **swarms** to coordinate actions like providing concentrated liquidity, executing large trades with minimal slippage, or diversifying strategies.
  - **Recursive Composition:** Build “manager” agents that oversee and allocate capital to specialized sub-agents based on performance.
- **Outcome:** Tradable, autonomous DeFi strategies where owning the token means owning the active, evolving agent and its capital.

### 3.2 Governance Delegates

- **Core Idea:** Turn governance participation into a dynamic, programmable, and accountable process.
- **How it Works:**
  - **Token-Scoped Reasoning:** Agents vote based on complex, pre-programmed logic considering proposal impact, principles, historical data, or even off-chain sentiment.
  - **Composability (Swarms):** Form voting blocs or liquid democracy structures where agents coordinate votes based on shared goals or emergent consensus.
  - **Recursive Composition:** Create hierarchical delegation systems where primary agents instruct specialized sub-delegates.
  - **ERC-6551 State:** Provides a transparent, onchain audit trail of voting decisions and rationale.
- **Outcome:** Holding a MODULE governance token grants ownership of a programmable, potentially autonomous voice within a DAO, enhancing transparency and sophistication.

### 3.3 AI DAOs

- **Core Idea:** Build decentralized organizations where core functions are managed by interacting intelligent agents.
- **How it Works:**
  - **The Swarm is the DAO:** The organization consists of a **swarm** of specialized MODULE agents.

- **Token-Scoped Reasoning:** Agents autonomously manage tasks like treasury diversification, grant analysis and distribution, parameter tuning, or even protocol upgrades based on embedded logic.
- **Composability:** Agents interact seamlessly, sharing data and coordinating actions to achieve the DAO’s objectives.
- **Recursive Composition:** Model complex organizational charts with sub-DAOs (agents owning other agents) handling specific domains.
- **Outcome:** Truly autonomous organizations with embedded AI, evolving governance, and transparent, automated operations driven by the agent swarm.

### 3.4 Index Agents

- **Core Idea:** Create dynamic, onchain investment vehicles that actively manage portfolios or mirror strategies.
- **How it Works:**
  - **Token-Scoped Reasoning:** Agents execute precise logic for mirroring wallets, rebalancing asset baskets held in their **TBA**, or running quantitative index strategies.
  - **ERC-20 Tradability:** The active strategy itself becomes a liquid token, directly tradable on DEXs.
  - **Composability (Swarms):** Agents can copy *other* MODULE agents or form swarms to collectively manage indices, adapting based on algorithms or sentiment.
  - **Recursive Composition:** Build meta-indices – agents managing portfolios of other index/copy-trading agents for strategy diversification.
- **Outcome:** Liquid, transparent, and programmable onchain funds and copy-trading mechanisms with verifiable performance history.

### 3.5 RWA Management

- **Core Idea:** Use intelligent agents as active managers for tokenized Real-World Assets (RWAs).
- **How it Works:**
  - **Token-Scoped Reasoning:** Agents autonomously manage RWA-related tasks like collecting rent into their **TBA**, distributing yield, scheduling oracle-triggered maintenance, or adjusting pricing based on market data.
  - **Composability (Swarms):** Create **swarms** to manage entire RWA portfolios (e.g., real estate), coordinating insurance, optimizing occupancy, or executing portfolio-level strategies.
  - **Recursive Composition:** Develop tokenized RWA funds where a primary agent holds and manages multiple underlying RWA agents.
  - **ERC-6551 Account:** Securely stores or links to relevant RWA documentation (deeds, compliance) directly within the token’s account.
- **Outcome:** More efficient, transparent, and potentially autonomous management of tokenized real-world assets and portfolios.

### 3.6 Simulation Swarms

- **Core Idea:** Deploy large populations of agents onchain to model and study complex adaptive systems.
- **How it Works:**
  - **Composability (Swarms):** Deploy vast **swarms** where each agent has specific logic (**Token-Scoped Reasoning**) representing actors in an economy, ecosystem, or social network.
  - **Onchain Interaction:** Agents interact, trade assets held in their **TBAs**, form coalitions, and evolve strategies based on programmed rules within a transparent, verifiable environment.
  - **Recursive Composition:** Model hierarchical structures within the simulation (e.g., firms, social groups).
- **Outcome:** A powerful, transparent, and reproducible platform for studying emergent behavior in decentralized systems with unprecedented granularity.

## 4 Conclusion

MODULE introduces a new class of token: the intelligent, composable agent. Through ERC-AI, it fuses ERC-20 and ERC-6551 into a programmable, swarm-enabled token structure, capable of holding assets, executing logic, and reasoning onchain.

By enabling agents to compose, MODULE unlocks the infrastructure layer for scalable decentralized AI—building toward agent economies, onchain simulations, and a new paradigm in token utility.

*“What is not good for the swarm, is not good for the bee.”*