CS4710 23/24
Overview of the First Multi-Agent System (MAS) Framework for agent-based applications in Web3: A Layman's Perspective

Tung Dao

February 18, 2024

1 Introduction

Abstract: *This paper delineates the Autonomous Economic Agent (AEA) framework, an innovative multi-agent system designed for Web3 interactions utilizing distributed ledger technologies (DLTs). Examining key drivers, it elucidates components fostering decentralized automation among autonomous yet interoperable agents. As an open-source tool, the framework bridges traditional architectures with emerging blockchain infrastructure, unlocking novel economic coordination possibilities. Emphasizing its significance, this analysis highlights AEA's role in scalable, self-managed Multi-agent Systems (MAS), offering potential for value creation among diverse stakeholders. The discussion caters to a broad academic audience, providing a succinct overview for both business and technology spheres.*

Keywords: Distributed Ledger Technologies, Multi-Agent Systems, Autonomous Economic Agent framework

1.1 MULTI-AGENT SYSTEM (MAS) for agent-based applications

A multi-agent system is a technology involving a group of intelligent, self-governing entities that interact in a cyber-physical environment to achieve common goals (Goonatilleke & Hettige 2022), (Shehata & El-Helw 2021). These agents operate in an unpredictable setting, where they may collaborate or compete and exchange or withhold information from one another . The core functionality of a MAS lies in the interaction among these structured entities within their environment. MAS offers benefits such as problem-solving intelligence, adaptability, and dependability when addressing complex distributed problems Goonatilleke & Hettige (2022), (Oprea 2004). Additionally, it enables task allocation, power distribution, and agent communication. In business contexts, MAS can support activities like energy management, disaster response planning; traffic control as well as container stacking management (Sonenberg et al. 2012), (Henesey et al. 2019).

The decentralized and diverse nature of MAS creates an ideal environment for modeling complex systems, offering a detailed insight into agent behaviors in different scenarios. With its scalability and adaptability, MAS provides a robust framework for representing entities with independent decision-making abilities Developer@fetch.ai (n.d.). Nevertheless, the challenges posed by system complexity and unpredictability call for continuous research to improve MAS performance in AI applications, paving the way for further progress in agent-based modeling and simulation.

1.2 MAS in the context of Web3

MAS in Web3 refers to the fusion of autonomous agents and blockchain technology for decentralized and secure communication and collaboration within the Web3 ecosystem. This integration enables the development of intelligent, cooperative, and secure systems that can interact both among themselves and with users in a decentralized manner.

This combination of emerging technologies offers numerous benefits. First, it allows for creating personalized and reliable services by utilizing the capabilities of intelligent agents. These agents are capable of providing real-time, adaptable, and tailored services to users thereby enhancing their experience within the Web3 ecosystem. Second, MAS in Web3 facilitates peer-to-peer (P2P) as well as device-to-device (D2D) communication which promotes seamless interaction between diverse devices and networks (Cao 2022), ultimately fostering interoperability throughout the Web3 ecosystem.

The exploration into integrating DLTs with MAS has been pursued to enable trust management alongside automated agent-to-agent communication (Mhamdi et al. 2022). This incorpation leverages the decentralized and secure nature of blockchain to enhance the efficiency and confidence of communication within the MAS (Woodward 2022), (Lad et al. 2020).

1.3 Methodology

The aim of this paper is to give an introductory overview of the potential for a specialized multi-agent system framework designed for agent-based applications in Web3. With decentralized platforms and blockchain technology gaining widespread adoption, there are increasing opportunities to utilize intelligent agents that can coordinate and make optimal decisions within Web3 environments. However, most current MAS frameworks were created prior to Web3 and lack native integration. To drive innovation in this field, three key research questions are identified:

1. How can the utilization of AI in MAS positively impact the efficiency and productivity of businesses, potentially leading to improved financial performance?
2. How can the integration of AI in MAS influence market dynamics, and what potential opportunities does it create for investors?
3. What are the core components of the first MAS framework for agent-based applications in Web3, and how do they integrate AI techniques to enhance the system's capabilities?

In order to comprehensively examine the landscape and shape the research inquiries on MAS frameworks customized for Web3, this study applied a systematic review methodology focused on recent publications from 2020-2023. The search involved academic databases such as IEEE Xplore, ACM Digital Library, and arXiv to retrieve works addressing multi-agent systems, Web3, and blockchain technology. Alongside scholarly articles, technical documents for prominent Web3 platforms like Fetch.AI (Developer@fetch.ai n.d.) were also analyzed. Moreover, relevant perspectives from expert blogs such as Shawn @Swyx Wang's Substack (Swyx 2023) and AEA developer forum (Minarsch n.d.) were incorporated. This process encompassed an assessment of over 20 sources through retrieval and selection filtering based on topical relevance and date range; it further entailed qualitative analysis to identify recurring concepts. Through this rigorous approach key themes were extracted along with discernible gaps in existing MAS frameworks concerning the requirements of emerging coordination paradigms within Web3 involving software agents - insights which contributed to shaping the framing and research questions of this paper.

2 Background

2.1 Overview of the evolution of MAS

For more than two decades, researchers have been working on the creation of multi-agent systems (MAS) (Minarsch et al. 2020). Initially, MAS emerged from the field of Distributed Artificial Intelligence (DAI) in the 1980s, signifying a novel and promising technology (Goonatilleke & Hettige 2022), (Ponomarev & Voronkov 2017), (Cao 2022). The evolution of multi-agent systems has been a fascinating journey, guided by the anatomy of autonomy idea (Swyx 2023).

The progress of MAS has been greatly influenced by the advancements in artificial intelligence, particularly with the rise of large language models (LLMs) such as OpenAI's GPT-4 (*GPT-4* n.d.)

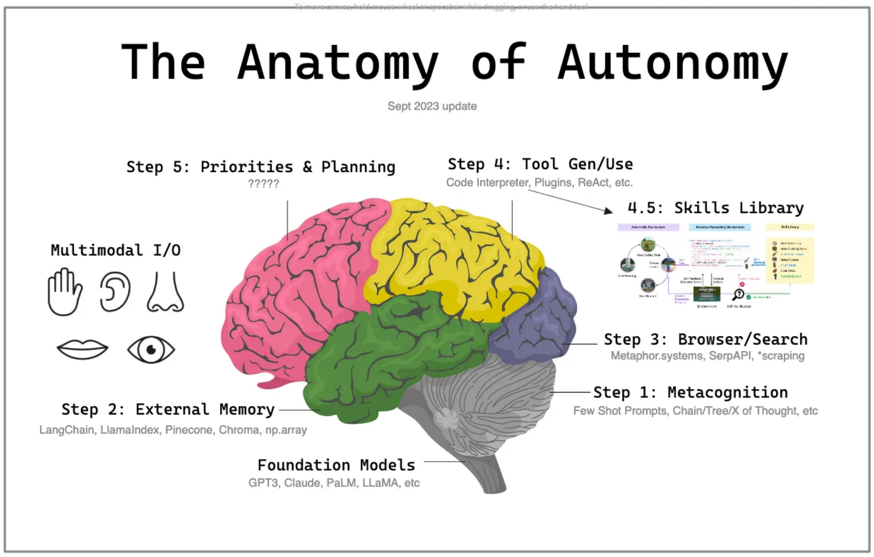
The Anatomy of Autonomy
Sept 2023 update
Step 5: Priorities & Planning
?????
Step 4: Tool Gen/Use
Code Interpreter, Plugins, ReAct, etc.
4.5: Skills Library
Multimodal I/O
Step 3: Browser/Search
Metaphor.systems, SerpAPI, *scraping
Step 1: Metacognition
Few Shot Prompts, Chain/Tree/X of Thought, etc
Step 2: External Memory
LangChain, LlamaIndex, Pinecone, Chroma, np.array
Foundation Models
GPT3, Claude, PaLM, LLaMA, etc

Figure 1: Evolution of MAS guided by the anatomy of autonomy idea

and Google's Gemini (Google DeepMind n.d.). Strategies like Chain of Thought prompting and the utilization of external memory have been formalized and enhanced to improve MAS performance, as evidenced in recent research (Wei et al. 2023), (Wang 2022). The integration of MAS with browser automation and the creation of autonomous agents capable of planning, reflection, and prioritization have expanded the capabilities of these systems, as demonstrated by ACT-1 Action Transformer (Wire 2022) and Reflexion agent (Shinn & Gopinath 2023). The idea of "AI social networks" and coordinating multiple agents to work harmoniously indicates a future where MAS can perform intricate tasks and self-improve within AI virtual software developer platforms like e2b (e2b dev n.d.). However, challenges such as prompt injection, task creation without hallucination, and the principal agent problem areas that necessitate additional research efforts (Swyx 2023).

2.2 Web3's Frontier: MAS and the Emergence of Autonomous Economic Agents

The significance of MAS in the context of Web3 is considerable. These systems offer essential functions for managing agents, communication, mobility, and directory services, among other things. They facilitate the development of decentralized intelligent systems and services that advance the Web3 ecosystem including blockchain, metaverse, and decentralized autonomous organizations (Cao 2022). While multi-agent systems have been developed in academia for many years, their adoption outside this setting has been limited. One reason for this is the lack of a native digital financial system within existing frameworks which is crucial for economic activities in MAS (Minarsch et al. 2020). With the increase in decentralized developments and the use of distributed ledger technologies such as Bitcoin and Ethereum, there is now a chance to incorporate MAS into Web3. This integration can be realized through frameworks for Autonomous Economic Agents (AEAs), which utilize trustless, non-intermediated exchange of wealth and smart contracts facilitated by DLT. Embracing this technological transition could lead to widespread adoption of MAS in Web3 environments.

AEAs are intelligent agents designed to operate on behalf of their owners with minimal interference while aiming to generate economic value. They are tailored specifically for adversarial and multi-stakeholder scenarios involving competing incentives between different actors. AEAs do not possess artificial general intelligence but instead have a narrow goal-directed focus aimed at achieving economic gain implemented through simple conditional logic (Minarsch et al. 2020). These agents coupled with DLTs, provide a solution that allows optimal allocation of off-chain and on-chain tasks combining strengths from both systems. The AEA framework addresses a critical limitation seen in smart contracts - purely reactive nature - by introducing proactive behavior necessary for autonomously executing complex business logic without reliance on external triggers (Minarsch et al. 2022),(Lipton & Levi 2018).

3 Framework Landscape: Navigating the Options for Intelligent Collaboration

3.1 Breaking down MAS development frameworks

Intelligent multi-agent systems can be developed and deployed with the help of frameworks. These tools and functionalities enable agents to interact, communicate, and collaborate with each other in order to achieve common goals. In other words, they serve as the foundation for building systems where agents can work together effectively. These frameworks allow developers to create agents with different reasoning and cognitive models, such as Procedural Reasoning System (PRS), Belief-Desire-Intention model (BDI), and Situation Calculus (Goonatilleke & Hettige 2022), (Oprea 2004). Overall, multi-agent system frameworks provide the necessary infrastructure and tools for intelligent collaboration in various fields.

MAS development frameworks act as the foundation for creating systems where independent agents work together intelligently. A variety of frameworks exists, each with its own unique strengths to meet different business requirements. For example, ASTRA extends the AgentSpeak language, Chromar uses a rule-based notation with stochastic semantics, and GOAL facilitates cognitive agent programming using logic-based language. Additionally, Jason provides an open-source interpreter for AgentSpeak while Gwendolen offers verifiable agent programs. JADE is a widely used Java-based framework that ensures interoperability and portability. Other frameworks like JADEL simplify JADE's complexities with a DSL and Jadex offers flexible BDI reasoning engine. LightJason introduces concurrent semantics and JaCaMo combines Moise, Jason, and CArtAgo for shared distributed environments. There are also lightweight options such as MaSMT which supports different agent types,and PLACE adoptsa planning-based approach for BDI models. All these frameworks contribute to the diverse landscape of MAS developments allowing intelligent collaborative system creation.

3.2 AEA Framework: The Rising Star

The Autonomous Economic Agent (AEA) framework, an open-source Python-based platform for creating multi-agent systems that can independently pursue economic objectives as defined by their owners, was developed to address the limitations of existing agent libraries and draw from web framework experiences. The main idea behind this framework is to expedite the development process of complete agents through code reusability and functionality encapsulation in order to tackle the distinct challenges of agent development (Minarsch n.d.).

Table 1 outlines four key elements of an AEA: *Skills*, *Connections*, *Contracts*, and *Protocols*. *Skills* embody the agent's business logic and capabilities while connections facilitate communication with external services and systems. *Contracts* serve as wrappers for interacting with smart contracts on distributed ledgers, whereas *Protocols* define message formats and interaction guidelines.

The architectural design follows an actor-like asynchronous approach with loose coupling between components where interactions occur via message passing instead of shared state. This design provides flexibility and supports concurrency while being managed by a runtime environment using either event loop (async mode) or threads (threaded mode) (Minarsch et al. 2022).

Table 2 presents a comparison between AEA framework and its alternatives, focusing on model, implementation language, features, and strengths. This advanced framework adopts a hybrid approach by combining reactive and deliberative elements rather than explicitly adopting a specific agent model such as BDI or rule-based models. Furthermore, it innovatively integrates with DLTs for financial settlements and commitments via smart contracts, facilitating multi-stakeholder sys-

Component	Description	Innovation
Skills	Primary business logic modules that implement an agent's capabilities. Consist of handlers, behaviors, and models.	Highly reusable and composable code components across agents. Flexible integration of AI approaches.
Connections	Handle networking, connect an agent to external services and systems. Translate between AEA framework and 3rd party comms.	Loosely coupled to rest of framework. Modular integration with diverse transport protocols.
Protocols	Define message types and sequence rules for dialogues.	Generalized protocol framework not tied to a specific standard. Interoperability through separation of encoding and transport.
Contracts	Provide access layer and wrappers around smart contracts and their execution logic.	Seamless integration with major distributed ledgers and ability to add more through plugins.

Table 1: Components and Key features of an Autonomous Economic Agent (AEA)

tems with explicit economic incentives. The modularity is akin to platforms like JADE but AEA framework distinguishes itself through protocol-agnosticism preventing constraint by FIPA or another protocol standard while balancing flexibility and interoperability. Additionally,,the DecisionMaker component handles the agent's wallet and economic behavior Developer@fetch.ai (n.d.). Noteworthy characteristics also include the encapsulation of business logic into reusable Skills, Protocol agnosticism offering choice in interaction rules, distribution of agents directly to end-users to reduce adoption barriers, and dynamic runtime loading of components.

The AEA framework stands out from previous multi-agent development platforms due to its modular architecture, integration with DLT, and emphasis on production deployments. However, a potential drawback is that the Python implementation may not be suitable for large agent populations because of performance limitations. This could potentially be addressed through compiled implementations.

4 Implications for Everyday Users

The AEA framework provides numerous advantages for regular users engaging with decentralized applications and protocols. Firstly, by eliminating the need to deal with blockchain details, the framework simplifies the development and usage of Web3 applications. For instance, an AEA can effectively automate interactions with blockchains and streamline processes such as options trading (8ball030 n.d.). This improves user experience by eliminating manual steps and minimizing errors. Furthermore, the integration of native wallet features and inclusion of economic components in the framework offer users greater management capabilities over digital assets and data.

However, users who are not familiar with Web3 concepts may face initial difficulties when interacting with AEA-based apps. The economic principles that underlie AEAs might not be easily understood by all users. Additionally, complex autonomous actions conducted on behalf of the user can lead to reduced transparency. Moreover, securing cryptographic keys effectively continues to be a challenge for many individuals (Minarsch n.d.). It will be crucial to provide comprehensive educational resources about Web3 and ensure adequate transparency in system operations in order to facilitate widespread adoption by everyday users.

As the framework grows, AEAs have the potential to fuel various applications in areas such as supply chain, mobility, and finance. By encoding and automating intricate interactions among multiple parties, AEAs can bring about efficiency improvements and new economic models. However, addressing transparency and security concerns regarding user-agent relationships is an important

Framework	**Model**	**Implementation Language**	**Key Features and Strengths**
ASTRA	BDI	Java	Extension of AgentSpeak language, based on logic programming
Chromar	Rule-based	Haskell	Stochastic semantics, continuous-time Markov chain
GOAL	Rule-based	Java	Logic-based, cognitive agent programming language
Jason	BDI	Java	Open-source interpreter for AgentSpeak, logic-based
Gwendolen	BDI	Java	Verifiable agent programs, typical features of BDI languages
JADE	FIPA	Java	Java Agent Development Framework, FIPA-compliant
JADEL	DSL	Java/JADE	Novel programming language, extension of JADE
Jadex	BDI and OOP	Java	BDI reasoning engine, flexible, runtime infrastructure
LightJason	BDI	Java	Concurrent BDI model, inspired by AgentSpeak and Jason
JaCaMo	BDI	Java	Multi-agent system development, programs in Moise and Jason
MaSMT	FIPA	Java	Lightweight framework, supports ordinary, manager, root agents
PLACE	BDI, HTN	Java	Planning-based Language, looks-ahead planning approach
AEA	Hybrid (Reactive + Deliberative)	Python	Native DLT and crypto economic integration, Production focus and end-user distribution, High reusability and composability of components

Table 2: Summary of MAS Frameworks

ongoing research focus.

In summary, the AEA framework offers clear benefits but poses some initial usage challenges for everyday users. Striking the right balance between automation and transparency while proliferating educational resources will determine whether AEAs transition from novel research concept to practical mainstream technology.

5 Conclusion

In conclusion, Autonomous Economic Agent framework represents the first specialized MAS for coordinating agents in Web3 environments. To enhance business efficiency, integrations such as the AEA architecture utilize AI and DLTs to automate interactions among multiple parties through smart contracts, resulting in increased productivity. Market dynamics may increasingly reflect incentives driven by tokenized models. Investing in the protocols that support such next-generation automation could prove lucrative. The analyzed components, including Skills and Connections, demonstrate specialized modules that incorporate AI techniques to improve decentralized orchestration. While challenges exist around transparency and evaluation, this foundational research enhances understanding for non-technical audiences about the potential of purpose-built MAS in Web3 contexts. The questions raised are intended to prompt responsive frameworks tailored to agent economies supported by maturing blockchain infrastructure. As MAS adoption follows increasing decentralization across sectors, addressing these open issues becomes essential to realize the potential scale of intelligent automation possible on this frontier. By simplifying complex technical aspects for mainstream accessibility while guiding ongoing improvements, this paper encourages stakeholders towards crucially contributive development directions.

References

8ball030 (n.d.), '8ball030/autonomoushegician: Helper application for hegic defi options using the fetch.ai stack'. Available at https://github.com/8ball030/AutonomousHegician. Last accessed: February 18, 2024.

Cao, L. (2022), 'Decentralized ai: Edge intelligence and smart blockchain, metaverse, web3, and desci', *IEEE Intelligent Systems* **37**(3), 6–19.

Developer@fetch.ai (n.d.), 'Agent-based and multi-agent systems'. Available at https://docs.fetch.ai/learn_the_concepts/agent-based_and_multi-agents_systems/. Last accessed: February 18, 2024.

e2b dev (n.d.), 'E2b-dev/e2b: Secure sandboxed cloud environments made for ai agents and ai apps', https://github.com/e2b-dev/e2b. Accessed: February 18, 2024.

Google DeepMind (n.d.), 'Google DeepMind', https://deepmind.google/technologies/gemini/#capabilities. Accessed: February 18, 2024.

Goonatilleke, S. & Hettige, B. (2022), 'Past, present and future trends in multi-agent system technology', *Journal Européen Des Systèmes Automatisés* **55**, 723–739.

GPT-4 (n.d.), https://openai.com/research/gpt-4. Accessed: February 18, 2024.

Henesey, L., Lizneva, Y. & Anwar, M. (2019), A multi-agent system with blockchain for container stacking and dispatching.

Lad, K., Ali, M., Dewan, M. & Lin, F. (2020), Trust management for multi-agent systems using smart contracts.

Lipton, A. & Levi, S. (2018), 'An introduction to smart contracts and their potential and inherent limitations'. Available at https://corpgov.law.harvard.edu/2018/05/26/an-introduction-to-smart-contracts-and-their-potential-and-inherent-\superintelllimitations/Accessed: February 18, 2024.

Mhamdi, H., Soufiene, B. O., Zouinkhi, A., Ali, O., Sakli, H. & Javed, A. R. (2022), 'Trust-based smart contract for automated agent to agent communication', *Intell. Neuroscience* **2022**.

Minarsch, D. (n.d.), 'Rise of the aeas - the autonomous economic agent (aea) framework explained'. Available at https://aea.dev/post-1/. Last accessed: February 18, 2024.

Minarsch, D., Favorito, M., Hosseini, S. A., Turchenkov, Y. & Ward, J. (2022), Autonomous economic agent framework, *in* N. Alechina, M. Baldoni & B. Logan, eds, 'Engineering Multi-Agent Systems', Springer International Publishing, Cham, pp. 237–253.

Minarsch, D., Hosseini, S. A., Favorito, M. & Ward, J. (2020), Autonomous economic agents as a second layer technology for blockchains: Framework introduction and use-case demonstration, *in* '2020 Crypto Valley Conference on Blockchain Technology (CVCBT)', pp. 27–35.

Oprea, M. (2004), Applications of multi-agent systems, *in* R. Reis, ed., 'Information Technology', Springer US, Boston, MA, pp. 239–270.

Ponomarev, S. & Voronkov, A. (2017), 'Multi-agent systems and decentralized artificial superintelligence'.

Shehata, H. A. & El-Helw, M. (2021), 'Modeling collaborative ai for dynamic systems of blockchained autonomous agents', *2021 3rd Novel Intelligent and Leading Emerging Sciences Conference (NILES)* pp. 421–426.

Shinn, N. & Gopinath, A. (2023), 'Reflecting on reflexion', https://nanothoughts.substack.com/p/reflecting-on-reflexion. Accessed: February 18, 2024.

Sonenberg, L., Stone, P., Tumer, K. & Yolum, P. (2012), 'Ten years of aamas: Introduction to the special issue', *AI Magazine* **33**, 11.

Swyx (2023), 'The anatomy of autonomy: Why agents are the next ai killer app after chatgpt', https://www.latent.space/p/agents. Accessed: February 18, 2024.

Wang (2022), '*Self-consistency: Learn Prompting - Your Guide to Communicating with AI*', https://learnprompting.org/docs/intermediate/self_consistency#notes. Accessed: February 18, 2024.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q. & Zhou, D. (2023), 'Chain-of-thought prompting elicits reasoning in large language models', https://arxiv.org/abs/2201.11903. Accessed: February 18, 2024.

Wire, B. (2022), 'Ai transformer inventors launch adept with $65m to lend a hand to knowledge workers', https://www.businesswire.com/news/home/20220426005963/en/AI-Transformer-Inventors-Launch-Adept-with-65M-to-Lend-a-Hand-to-Knowledge-Workers. Accessed: February 18, 2024.

Woodward, C. (2022), 'Analysis of integrating blockchain technologies into multi-agent systems', https://doi.org/10.48550/arXiv.2212.12313.