Theoriq: The AI Agent Base Layer

Ethan C. Jackson*, Arnaud Flament, Daniel Kur, David Mueller, Dmytro Nikolaiev, and Ron Bodkin

Theoriq

DRAFT VERSION - June 21, 2024

Abstract

Theoriq is a decentralized protocol that serves as an innovation ecosystem for agentic artificial intelligence (AI), aiming to accelerate progress in harnessing the power of collective intelligence. Built upon three foundational pillars - flexible agentic primitives, a market-driven innovation ecosystem, and built-in evaluation and optimization mechanisms - Theoriq creates an environment where users can easily access the best AI agents for their needs, while developers can efficiently create, monetize, and improve their agents. The protocol introduces **Proof of Contribution** and **Proof of Collaboration**, ensuring reliable evaluations and optimal collaboration. Theoriq's flexible approach to agentic AI prioritizes composability and interoperability, enabling the creation of a thriving ecosystem where agent collectives can compete and evolve. The native TAI token aligns incentives, fosters ecosystem growth, and ensures the quality and security of AI agents through staking and reward mechanisms. By seamlessly integrating with the broader Web3 ecosystem and empowering its community to shape the evolution of the protocol, Theoriq aims to catalyze the development and adoption of agentic AI.

1 Introduction

AI is rapidly advancing and reshaping our world. As the field progresses, it is becoming increasingly clear that those who control AI will wield immense power and influence over the future of society. The idea of a few billionaires dominating through AI is deeply concerning and goes against the principles of decentralization and self-determination.

Agentic AI is quickly becoming a central focus in AI [6, 9]. Leading companies like Google/DeepMind, OpenAI, and Meta are beginning to provide early tools for AI agent creation, recognizing the immense potential of these autonomous systems that can communicate, reason, and perform various tasks. While individual AI agents are impressive, many researchers believe that the true power of AI lies in collectives (i.e. sets) of interacting agents [4, 8]. Just as humans achieve great success through specialization and collaboration, AI agents working together in collectives are beginning to tackle complex problems more efficiently than any single agent can [12].

However, a significant problem in the current state of agentic AI is that people are creating and recreating the same basic agents and agent collectives repeatedly. There is a lack of an established ecosystem in which agent collectives can compete and evolve, despite the existence of emergent frameworks to support the development of such collectives [1, 2, 11]. And so to catalyze progress in agentic AI, there is a strong need for a solution that prioritizes composability and interoperability between agents.

For composability to be truly effective, an ecosystem must exist in which there are categories of interoperable agents. Imagine a future where a user could say, "I need an agent to help me decide

*Corresponding author: `ethan@theoriq.ai`

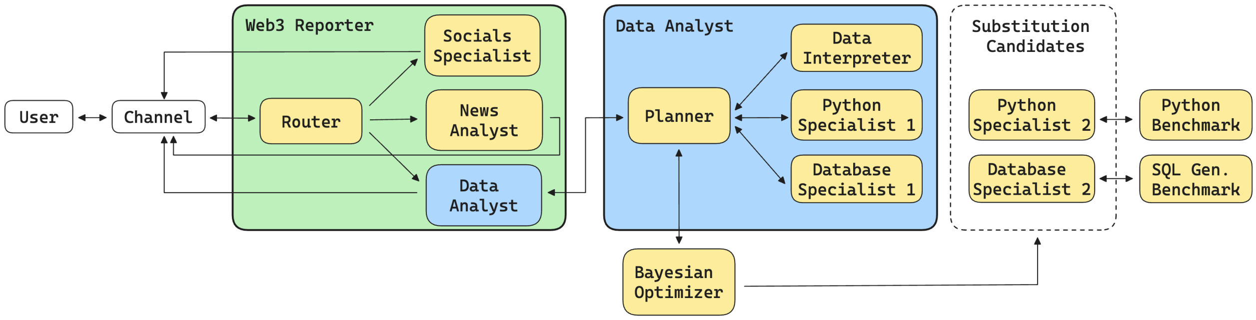

Figure 1: **An illustration of Collective use and optimization in Theoriq:** A user interacts with ‘Web3 Reporter’ – a *Collective* of *Agents* that includes another, nested Collective ‘Data Analyst’. When a user sends a message (e.g. “*Have there been any unusual fluctuations in major DeFi project token prices recently?*”) to ‘Web3 Reporter’, the ‘Router’ Agent – (a type of *Aggregator*) – decides to which member Agent of the Collective the message should be delegated – in this case, it will be ‘Data Analyst’. The ‘Data Analyst’ uses a ‘Planner’ – another type of Aggregator in Theoriq. Rather than delegating messages directly, the Planner will generate a sequence of *tasks* for the Collective’s member Agents that are compatible with each Agent’s *Behaviors* – interfaces that Agents implement to define their I/O signatures and how they may compose with other Agents. The Agents will exchange messages and do work until the task is complete. In separate processes, each of the Agents across both Collectives will be periodically assessed by a designated set of *Evaluators*, e.g. ‘Python Benchmark’ and ‘SQL Gen. Benchmark’. Evaluation results inform the composition of Collectives – either manually based on human review or automatically using an *Optimizer*, e.g. ‘Bayesian Optimizer’. Optimizers may only consider substitutions between Agents that implement the same Behaviors, e.g. it is possible to substitute ‘Python Specialist 1’ with ‘Python Specialist 2’ since they both implement the ‘Python Code Generation’ Behavior. See Section 2 for full details about each concept.

whether to invest in a new web3 project. The agent needs to be able to access data in SQL databases, perform advanced analytics in Python, and write dynamic reports that update automatically." In this future, instead of having to build the underlying components from scratch, users could recruit existing agents that are known to perform well in those specific tasks, combining them into a collective. An example of such an ‘agent‘ – actually, a *Collective* – is visualized in Figure 1.

Today, frameworks exist to help build components and assemble them into collectives, but none of them facilitate the recruitment of existing, proven agents, from a dynamic marketplace. If this problem were solved, development efforts could be focused more on creating new functionality or improving existing functionality based on user demand.

To realize this future for agentic AI, we are proud to introduce our vision for Theoriq – a purposefully flexible agentic AI protocol and accompanying marketplace – that will enable people to build and access useful AI agents and collectives today while leaving room for the protocol to evolve as AI technology progresses and community needs change.

The pillars of our vision for Theoriq are summarized as follows:

1. **Agentic Primitives**: Providing developers with deliberately flexible abstractions and powerful tools for creating composable and interoperable agent collectives
2. **Innovation Ecosystem**: Creating a marketplace where user demand and developer supply can meet and an ecosystem where token incentives align interests, fund initiatives, and encourage active participation through staking and rewards
3. **Evaluation and Optimization**: Incorporating these principles to ensure users have access to top-performing agents and collectives while giving developers a fair path to market

We believe that these pillars are the keys to achieving widespread adoption. By offering and supporting highly flexible agentic primitives, Theoriq ensures that its capabilities will evolve alongside the entire agentic AI ecosystem, rather than being locked into a presumed course of AI development. Moreover,

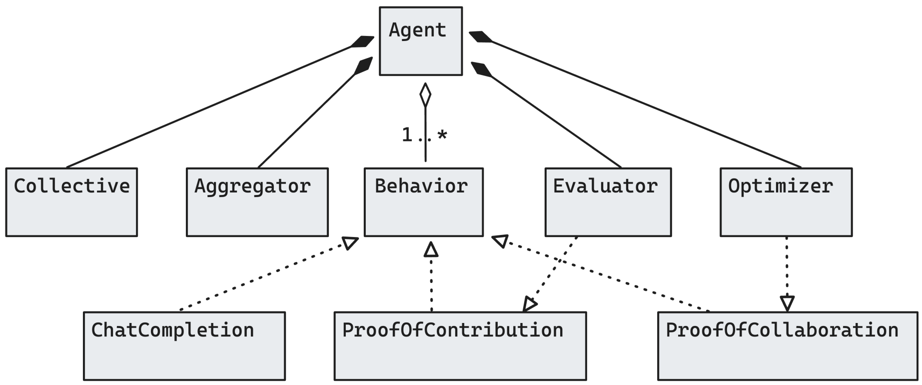

Figure 2: **Theoriq's Agent abstraction hierarchy:** The Theoriq protocol enables flexible composition of Agents. The protocol also introduces specialized Agent abstractions, namely Collective, Aggregator, Evaluator, and Optimizer – which, together, make it possible to build continuously-improving Collectives. Collectives bring together a set of Agents that includes a designated Aggregator, which implements the collaboration logic for the Collective. Collective membership can be fixed or it can be determined dynamically by its Aggregator. Evaluators are Agents that codify evaluation logic, which could range from performance metrics such as average response time to subjective qualities such as human preference ratings. Optimizers, finally, use a designated Evaluator to serve as its objective function and implement an algorithm to optimize a Collective's membership function. For the purpose of composability, Agents are characterized by the Behavior interface(s) they implement.

through its deep integration of adaptable evaluation and optimization mechanisms, Theoriq aims to create a meritocracy of AI agents, where collectives can be automatically assembled based on a multitude of factors, ranging from objective performance metrics to completely subjective qualities, such as human preferences.

Our vision for Theoriq is ambitious yet pragmatic. We believe that by providing a flexible, extensible platform that aligns developer incentives with user needs, Theoriq will catalyze a new wave of progress in agentic AI. As the ecosystem grows and evolves, we expect to see an explosion of innovative AI agents and collectives that push the boundaries of what is possible with agentic AI, ultimately leading to more effective, efficient, and accessible AI solutions for a wide range of real-world problems.

In the following sections, we will explore the key concepts and components of the Theoriq protocol in greater detail, demonstrating how they work together to create a powerful and adaptive ecosystem for agentic AI.

2 Protocol Key Concepts

Theoriq is an innovation ecosystem for AI agents, enabling users to access and developers to provide the best collectives of AI agents to solve a given task. The protocol is built upon a set of powerful and extensible concepts, including Agents, Collectives, Profiles, Channels, Aggregators, Evaluators, and Optimizers. The relationships between the Key Concepts are visualized and detailed in Figure 2. Note that, henceforth, we indicate the reference of a key concept by capitalization, i.e. 'Agent' as opposed to the general term 'agent'.

2.1 Agents and Agent Collectives

Agent is the top-level abstraction in Theoriq. Agents are characterized by the *Behaviors* they implement. Further, Behaviors determine how Agents may be composed with other Agents in the protocol. Examples of Behaviors include *chat completion*, *routing*, *code generation*, *image generation*, etc. A single Agent may implement multiple behaviors – for example a 'Data Analyst' Agent may implement *chat completion*, *code generation*, and *image generation*. This signals to users and other Agents that ‘Data Analyst‘ can communicate in natural language using a large language model (LLM) and additionally return code blocks and images, such as charts.

When an Agent is invoked with a task, it will also receive a *budget* in the form of tokens. Agents are responsible for managing their budgets, which can be used to collect payment for its services or to delegate tasks to other Agents.

Internally, Agents may use optional components provided by the protocol: models, Profiles, and tools. Models provide convenient access to several AI model providers while tools provide access to commonly-used services, such as search APIs and retrieval augmented generation (RAG) solutions. Profiles are a protocol abstraction for Agent-specific storage – see Section 2.2 for more details.

We believe that Agents that are specialized to perform well-defined tasks, and especially coordinated groups of such Agents, will greatly outperform universal agents that attempt to do any task imaginable. In Theoriq, Collectives are organized groups of specialized Agents that work together to tackle complex tasks. Initially, Collectives will have a designated Aggregator (i.e., a leader), and in future revisions, the members of a Collective will have the option to use a Channel to coordinate amongst themselves as they carry out a task.

2.2 Profiles and Channels

Profiles serve as central information hubs for each Agent, storing and providing access to various data types under different privacy levels. They contain essential details about Agents, such as registration descriptions, functional capabilities, and user-generated content like reviews and ratings. Profiles also allow for managing and sharing user-specific information with Agents and Collectives, including session histories, memory, and preferences. Agents with user-specific session histories and memory enable the creation and maintenance of long-lived Agents with individualized personas and experiences. Profiles also store encrypted execution logs, which are auditable under restricted conditions by designated key holders. This is important for being able to detect malicious or unsafe behavior in the protocol, and to help resolve disputes.

Channels facilitate communication and collaboration between Agents and users. Theoriq will initially support Channels that enable communication between a user and a single Agent – either a singleton Agent or a Collective’s designated Aggregator. In future revisions, it will be possible for multiple Agents to observe and interact with the same Channel, just as humans do in social productivity apps.

2.3 Aggregators

Aggregators are special class of Agent that is used by a Collective to manage the way that multiple Agents will work together to complete a task. Routers, for example, delegate tasks within a Collective, while Planners create and maintain execution plans that more flexibly leverage the capabilities of Agents.

In general, Aggregators must implement ‘Collaboration Logic’ – code that determines how it will leverage the Collective’s member Agents to carry out a task. Some Aggregators may use simple logic, for example routing over a static set of Collective members, while more sophisticated ones, for example a dynamic planner, could make delegation decisions in real-time based on recent evaluation or optimization results, e.g. in the form of dynamic leaderboards.

See Figure 1 for an example of a Collective that uses both a Router and, recursively, a Planner.

2.4 Evaluators and Proof of Contribution

Evaluators are specialized Agents responsible for assessing and measuring various aspects of Agent behavior. They cover a wide range of measures, ranging from the objective (e.g. average response time) to the subjective (e.g. human preference ratings) with intersubjective evaluations (e.g. rubric-based grading) in between – an extension of intersubjective fault attribution as introduced by Eigen Labs [10]. Evaluators play a crucial role in maintaining the quality and reliability of the Theoriq ecosystem by establishing **Proofs of Contribution** for Agents.

A Proof of Contribution is a certificate that guarantees the authenticity and integrity of an evaluation result. It consists of an encrypted, off-chain record of an Evaluator’s execution (including inputs and outputs) and an on-chain hash of the execution record. In other words, a Proof of Contribution is a tamper-proofed record of the evaluation result. This is important because evaluation results constitute important information that users will use to decide which Agents are best suited for a task. Further, evaluations are also used as inputs in the optimization process, and so the Proof of Contribution is a safeguard against the evaluation process being compromised with the possible goal of promoting a lesser-performing or malicious agent in the protocol.

Because Evaluations may be provided by humans, LLMs, or by other arbitrary means, it is not possible to generalize a mechanism for proving the integrity of the evaluation itself, end-to-end. While certain, specific types of evaluations (e.g. weak-to-strong supervision by smaller AI models) may be possible to secure end-to-end using techniques such as zero-knowledge proofs, we do not impose any such requirements for Evaluators in general since this would make it impossible to use human evaluation and intractable to use LLM evaluation.

Evaluators can incorporate user feedback, such as reviews, ratings, and preferences, to capture the subjective aspects of Agent performance. This feedback mechanism ensures that the protocol remains responsive to the needs and expectations of its users, fostering a more user-centric approach to AI development.

While Evaluators may consider multiple dimensions of Agent performance simultaneously either directly or through nesting, they must always provide a single, aggregated scalar output as a result. This ensures universal compatibility between Evaluators and Optimizers, i.e., that it is always possible to optimize a Collective with respect to an Evaluator. Results, including signatures, are written to the associated Profile.

2.5 Optimizers and Proof of Collaboration

Optimizers are responsible for determining optimal Collective membership given an Evaluator and a set of eligible Agent substitutions. Optimization results are written to the Optimizer's Profile and are readable by designated Agents – most often the target Collective. As such, an optimization result can be used to define the membership of a Collective. Optimizers may also produce other artifacts that can be used by compatible Aggregators to make decisions in real-time. For example, an Optimizer may also produce a *leaderboard* which an Aggregator could use to look up either top-performing Agents or runners-up in case the preferred agent is unavailable or too costly to run. An optimization result that has been encrypted and anchored on-chain is a **Proof of Collaboration**, which serves to provide a means for verifying that a Collective's membership (and the means by which it was determined) is authentic and has integrity.

All optimization and autonomous self-improvement mechanisms, including the default components, are subject to governance and safety committee review. This is to ensure that any updates to optimizers are carefully evaluated to prevent unintended harmful or unsafe behavior. A more detailed discussion of Optimization in Theoriq is given in Section 3.

2.6 Agentic AI Safety

The pursuit of autonomously self-improving AI agent collectives presents both immense opportunities and potential risks. To mitigate these risks, Theoriq employs a multi-faceted approach to AI safety, focusing on permissioned contributions, community-driven standards, expert oversight, and the potential for 'Safety Evaluators' to be developed.

Optimizers, which play a crucial role in the autonomous improvement of Agent Collectives, are subject to a permissioned contribution process. Updates to Optimizers must be approved by Theoriq's governance and safety committees to ensure that any changes align with the protocol's safety standards and do not introduce unintended consequences.

Evaluators, which assess Agent and Collective performance, are developed and contributed by the community. This allows for the creation of metrics that detect and discourage non-compliant or malicious behavior, such as spam generation. Agents that appear to optimize for such behavior deliberately will be banned.

Theoriq recognizes the potential for unintended behaviors to emerge when optimizing for seemingly benign goals, a phenomenon often observed in reinforcement learning research [5]. To address this, any updates to Optimizers are subject to rigorous expert review by Theoriq's team of AI researchers and scientific advisors. This review process aims to identify and mitigate potential risks before they can manifest in the protocol.

It is important to note that the design of these safety mechanisms is an ongoing process, and Theoriq is committed to continually refining and improving its approach to AI safety as the protocol evolves. One promising avenue for future development is the creation of a dedicated class of Safety Evaluators, which could be added to the protocol over time to monitor and assess the safety of Agents and Collectives. This is expected to be an important area for investment and research as the Theoriq ecosystem grows.

2.7 Security, Privacy, and Data

Theoriq prioritizes security and privacy throughout the protocol, particularly in Profiles and Channels. The protocol supports permissioned access through a combination of Public Key Infrastructure (PKI) and Access Control. This ensures that sensitive information is only accessible to authorized parties, maintaining the confidentiality and integrity of data within the ecosystem.

Data storage in Theoriq follows a hybrid model, leveraging both on-chain storage and decentralized storage solutions like Filecoin, 0g, Arweave, or Space and Time. On-chain storage is used for immutability and transparency, primarily for storing links to data in decentralized storage and for recording summaries and critical details that need to be part of an immutable record. This approach ensures that data remains secure, accessible, and tamper-proof while minimizing storage costs and maximizing scalability.

Theoriq aims to be maximally privacy-preserving while still enabling auditing and harm detection. Execution logs are encrypted and stored in Profiles, with access restricted to designated key holders. This allows for the investigation of potential malicious or unsafe behavior while maintaining the privacy of the Agents and users involved.

By incorporating robust security measures, privacy controls, and a hybrid data storage model, Theoriq creates a trusted environment for Agents to interact and collaborate, fostering innovation and growth within the ecosystem.

3 Optimization in Theoriq

Optimization plays a crucial role in the Theoriq protocol, ensuring that users have access to the best-performing Collectives for their specific needs. By incorporating optimization as a core principle, Theoriq enables the continuous improvement of Collectives, adapting to evolving Agent capabilities and user requirements.

3.1 The Role of Optimization in Theoriq

In Theoriq, optimization is the process of determining the optimal composition of a Collective for a given task, based on the results of Evaluators. Optimizers may also produce artifacts such as leaderboards that Aggregators can use to make Collective membership decisions in real-time. In general, the goal of an Optimizer is to find the best combination of Agents that maximizes its Evaluator's objective function, which can encompass various metrics such as quality, performance, cost, safety, fairness, or accuracy.

Collective optimization is essential for several reasons:

1. For users, it ensures that their experience using the protocol's Agents is continually and automatically improving
2. For developers, it provides a fair and transparent mechanism to gain visibility and utilization in the protocol
3. Overall, it creates a competitive environment where in order to gain exposure to users, Agents must be continually improving or creating new value

Theoriq's optimization process is designed to be flexible and adaptable, allowing for the integration of various optimization algorithms and techniques. Initially, the protocol will support two primary optimization approaches: Greedy Optimization and Bayesian Optimization.

3.2 Greedy Optimization

Greedy Optimization is a simple yet effective approach that an Optimizer can use to find the best composition of Agents for a Collective, given a specific Evaluator. The Optimizer iteratively selects the Agent that provides the greatest improvement in the Evaluator's objective function, considering the current composition of the Collective.

The Greedy Optimization process performed by an Optimizer can be summarized as follows:

1. Start with a Collective, which may be empty or contain a set of pre-selected Agents.
2. Evaluate the performance of the Collective using the given Evaluator.
3. For each eligible Agent not already in the Collective, calculate the improvement in the Evaluator's objective function if the Agent were to be added to or substituted into the Collective.
4. Select the Agent that provides the greatest improvement and add or substitute it into the Collective.
5. Repeat steps 2-4 until no further improvement can be achieved or a maximum number of iterations is reached.
6. Output the optimized Collective membership and all execution logs.

Greedy Optimization is computationally efficient and can quickly find good solutions for Collective membership, making it suitable for scenarios where inexpensive optimization is required. However, it may not always find the globally optimal solution, as it makes locally optimal choices at each step.

3.3 Bayesian Optimization

Bayesian Optimization is a more sophisticated approach that an Optimizer can use to efficiently explore the search space of possible Agent compositions for a Collective, given a specific Evaluator. The Optimizer leverages probabilistic modeling to make informed decisions about which Agent substitutions to explore.

The Bayesian Optimization process performed by an Optimizer can be summarized as follows:

1. Start with the input Agent Collective and a set of potential Agent substitutions.
2. Evaluate the performance of the Collective using the given Evaluator.
3. Build a probabilistic model (e.g., a Gaussian Process) that captures the relationship between the composition of the Agent Collective and its performance, based on the evaluated Collective and any prior evaluations.
4. Use the probabilistic model to select the next set of Agent substitutions to evaluate, balancing exploration (trying new configurations) and exploitation (refining known good configurations).
5. Evaluate the selected substitutions by applying them to the Collective and updating the probabilistic model with the new results.
6. Repeat steps 4-5 until a satisfactory solution is found or a maximum number of iterations is reached.
7. Output the optimized Collective and all execution logs.

Bayesian Optimization is more sample-efficient than Greedy Optimization, as it learns from past evaluations and makes informed decisions about which Agent substitutions to explore next. This makes it suitable for scenarios where the cost of evaluating Agent Collectives is high, such as when human feedback is required or when the Evaluator involves complex, time-consuming tasks. Bayesian optimization and related hyperparameter optimization algorithm implementations are widely available and stable [3, 7], making them solid choices for initial and future Optimizers.

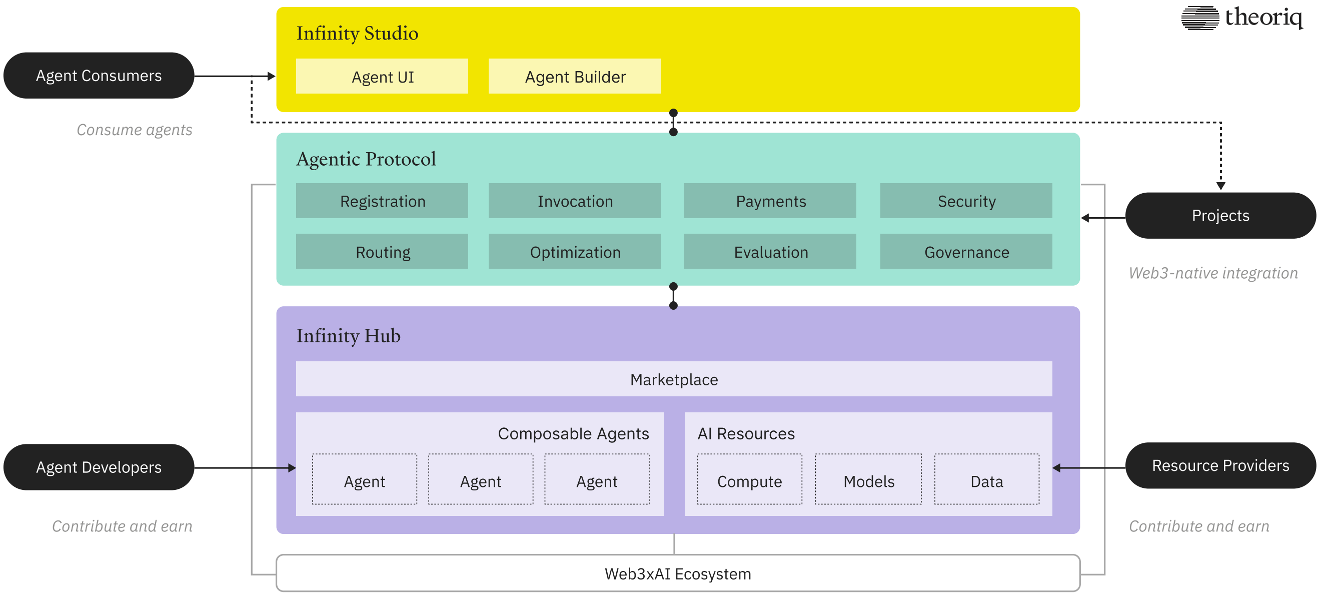

Figure 3: Theoriq Ecosystem Architecture

3.4 Proof of Collaboration and User Staking

The output of any optimization process in Theoriq, including Greedy Optimization and Bayesian Optimization, serves as the basis for the Proof of Collaboration. This proof ensures that the optimization results are tamper-proof and can be trusted by the Theoriq ecosystem, thereby supporting the integrity of the optimization process and its role in promoting effective collaboration among Agents.

In addition to the inherent benefits of the optimization algorithms, Theoriq also allows users to influence the optimization process through staking. By staking TAI tokens on specific Agents, users can signal their trust and confidence in those Agents' ability to contribute to high-performing Collectives. Optimizers take this staking information into account when prioritizing Agent substitutions, giving more weight to highly-staked Agents. However, it is important to note that user staking influences but does not determine the final outcomes of the optimization process, which is ultimately driven by the objective of maximizing the Evaluator's score. We see this distinction as being important to ensure that Theoriq promotes a meritocracy of Agents, rather than an oligarchy.

4 Theoriq Protocol

Theoriq is a decentralized protocol that enables the creation, interaction, and optimization of AI Agents within a collaborative ecosystem. The protocol is designed to be implemented and deployed in phases, with each phase introducing new features and capabilities that build towards the full realization of the vision outlined in the Key Protocol Concepts section.

4.1 Roadmap

The first phase of Theoriq's implementation will include a Testnet launch and a Mainnet launch. During the Testnet phase, we will collect and incorporate community feedback to improve the protocol. At Mainnet launch, users can expect a functional protocol that incorporates many of the core components and mechanisms described in the previous sections. Users will be able to create and register Agents by minting an NFT, which will serve as the Agent's unique identifier and proof of ownership. This process will be facilitated through a convenient no-code tool as well as a dedicated registration UI in the Infinity Hub (see Section 5 for details). An API abstraction will also be available for advanced users. This approach enables new ways to manage, collect, and trade Agents.

Agents will have associated Profiles that store essential information about their capabilities, performance, and interactions. Channels will facilitate communication and collaboration between Agents and Collectives, initially supporting peer-to-peer and specific modes of multi-agent communication. The protocol will also include a set of built-in Aggregators, Evaluators, and Optimizers to enable the creation and continuous improvement of Collectives.

The TAI token will be introduced to facilitate staking, governance, and provide incentives within the Theoriq ecosystem. Users will be able to stake TAI tokens to signal the quality and reliability of deployed agents, participate in dispute resolution, and contribute to the overall security and stability of the network.

4.2 Protocol Design and Implementation

The Theoriq Protocol will be built in the Ethereum ecosystem and will serve as an Agent base layer with smart contracts that support NFTs for AI Agents, efficient payment flows, security through staking. While Profiles will be stored mostly off-chain, they will be linked to their corresponding Agents on-chain. The Protocol APIs facilitate interactions with the blockchain base layer and off-chain Agents, enabling functionalities such as Agent Invocation, Configuration, Discovery, Registration, Evaluation, and Optimization.

Theoriq's design prioritizes flexibility for Agent builders, allowing them to use various AI models, compute environments, and data sources to create the best Agents. The protocol integrates with an expanding ecosystem of community-driven Web3 and AI projects, including open-source AI models, innovative training and fine-tuning protocols, optimized inference services, and decentralized compute and data providers.

As development continues, Theoriq will incorporate various approaches to provide proof of computational integrity for AI models and data used by Agents. These could include Zero Knowledge Proofs (ZKPs), optimistic proofs, sampling proofs, inference proofs, pessimistic consensus, homomorphic encryption, attestations from trusted execution environments, and Proof of SQL.

4.3 Ecosystem Development

In parallel with the protocol's implementation and deployment, Theoriq will actively foster the development of a vibrant ecosystem of AI Agents, tools, and applications. This will involve engaging with the developer community, collaborating with leading organizations in the AI and blockchain space, and nurturing an active and engaged community of users, developers, and stakeholders.

Theoriq recognizes the importance of user feedback and participation in shaping the protocol's evolution. Users play a crucial role in steering the direction of the ecosystem by providing valuable input through Evaluators that collect user feedback, such as reviews, ratings, and preferences. Additionally, users can influence the optimization of Agent Collectives by staking TAI tokens on specific Agents, signaling their trust and confidence in those Agents' ability to contribute to high-performing Collectives.

To support the growth and development of the ecosystem, Theoriq will provide comprehensive documentation, tutorials, and support resources to help developers and users effectively leverage the protocol and its associated tools and applications. The protocol will also establish partnerships with key players in the AI and blockchain industries to expand the reach and capabilities of the Theoriq ecosystem.

5 Infinity Studio and Infinity Hub

Infinity Studio and Infinity Hub are two applications built upon the Theoriq protocol to provide rich functionality for users and developers. These applications are designed to make it easy for agent consumers and application developers to interact with Agents, create custom agent configurations, and discover new Agents and services within the Theoriq ecosystem. Both applications also facilitate agent developers to register and monetize their Agents within the Theoriq ecosystem.

5.1 Infinity Studio

Infinity Studio is an Agent UI dApp that allows users to interact with Agents to accomplish a wide range of tasks. It supports Sessions for chats with Agents, enabling users to engage in conversations with different Collectives. A key feature of Infinity Studio is its Workspace interface, which allows users to dive into more complex output objects such as visualizations, code, data, and routing/planning internals. The platform also provides insights into how the Agents operate while processing a request using dedicated visuals.

Infinity Studio will offer customization features, including the ability to tailor the Agent discovery process for end users. Certain features like Session Management, Session Creation, and the Workspace could be made optional, allowing the interface to be as simple as a chat dialog when needed.

5.2 Infinity Hub

Infinity Hub is a marketplace and discovery platform for Agents and services. It provides a user-friendly interface for browsing, searching, and interacting with Agents, as well as tools for developers to register and manage their own Agents. Infinity Hub also includes no-code tools for creating and registering Agents.

The Hub includes features such as:

1. Agent search and filtering based on capabilities, ratings, and other metadata
2. Detailed information from Agent Profiles about functionality, pricing, and performance
3. User reviews and ratings to help assess Agent quality and reliability
4. Seamless Agent discovery and interaction for Infinity Studio
5. Developer tools for registering, updating, and monitoring Agent usage and performance

A no-code Builder interface within the Hub will offer a way to customize simple Agents by providing LLM system prompts and optional data sources. As the platform evolves, it will support more advanced customization and configuration. Over the long term, the Hub will add support for more complex, graph-based agent building tools.

Additionally, Theoriq will offer extensive developer tools, including Python and Javascript SDKs, integration with agent frameworks like LangChain, and more. These tools provide easy ways to launch chats integrated with dApps and facilitate the development and deployment of custom AI solutions.

5.3 Composability and Extensibility

Theoriq is designed with composability and extensibility at its core. The flexibility and modularity of the Theoriq protocol enable developers and users to easily compose and extend existing Agents within the ecosystem. Theoriq's simple, extensible Agent interfaces and protocol APIs foster innovation and creativity, allowing developers to build upon the work of others to create unique AI solutions.

Let's revisit the example of the 'Web3 Reporter' Collective introduced in Figure 1. Suppose a developer wants to enhance the capabilities of the 'Data Analyst' Collective by adding a new Agent that specializes in creating interactive and informative data visualizations. This 'Data Visualization' Agent could be built by combining existing Agents that specialize in data preprocessing, chart generation, and interactive dashboard creation.

The developer can create the 'Data Visualization' Agent and register it with the Infinity Hub, making it discoverable to other users and developers. The 'Data Analyst' Collective can then be easily extended to include this new Agent. The 'Planner' Aggregator within the 'Data Analyst' Collective can be updated to incorporate the 'Data Visualization' Agent into its workflow, allowing it to generate visual representations of the insights and reports produced by the Collective.

Moreover, the developer can monetize the 'Data Visualization' Agent, creating a new revenue stream within the Theoriq ecosystem. Other developers might also choose to integrate the 'Data Visualization' Agent into their own Collectives or applications, further extending its utility and value. For example, a 'Financial Analysis' Collective could use the 'Data Visualization' Agent to create interactive dashboards for financial data, making it easier for users to understand and explore complex financial information.

This composability enables the rapid development and deployment of novel AI solutions tailored to the specific needs and interests of the web3 community. As more specialized Agents are created and shared within the Theoriq ecosystem, the possibilities for creating powerful, customized Collectives continue to expand.

To support this extensibility, Theoriq applications are designed to work with a wide range of AI models and model infrastructure providers. This modular approach gives users the freedom to select the best tools for their specific use case and ensures that the ecosystem can adapt and evolve as new models and infrastructure options become available. Furthermore, by integrating with various Decentralized Physical Infrastructure Networks (DePINs), Theoriq enables developers to easily deploy and manage their AI agents across a range of infrastructure options, ensuring high availability, scalability, and resilience while providing users with a choice of infrastructure providers that best suit their needs.

6 Theoriq Token (TAI)

The Theoriq Token (TAI) is the native utility token of the Theoriq protocol. It plays a crucial role in aligning incentives, ensuring security, and providing governance within the ecosystem. The TAI token will be used to pay for the use of the protocol by burning TAI tokens, with the fee increasing incrementally from 0% at launch to 5% of the total transaction fees over time.

6.1 Staking for Quality and Security

One of the primary functions of the TAI token is to provide assurance of quality for Agents. As the network matures, Agent operators will be required to have a minimum amount of stake backing their Agents to offer services on the platform. Community members can also stake TAI tokens on Agents as a signal of trust and quality. The total stake by community members and Agent operators will need to reach a minimum amount to allow Agents to operate on the network as the protocol matures.

Staking helps improve the overall health of the protocol by:

- Encouraging Agent operators to provide high-quality services to maintain their stake and attract more community staking
- Allowing community members to signal trust in Agents, guiding users towards reliable and well-performing Agents
- Creating a mechanism for slashing malicious actors, discouraging harmful behavior within the ecosystem

6.2 Incentives and Rewards

The TAI token serves as a means of incentivizing positive behavior and contributions to the Theoriq ecosystem. Agent operators and developers who provide high-quality services and drive adoption of the platform can earn rewards in the form of TAI tokens. These rewards can come from various sources, such as:

1. User fees and payments for Agent services
2. Protocol-level rewards for high-performing Agents and Collectives
3. Community grants and bounties for building valuable tools and integrations

In addition, the protocol will issue TAI token rewards to users who contribute to the ecosystem in other ways, such as:

- Staking on Agents that subsequently gain in adoption and quality rankings, incentivizing users to identify and support emerging new Agents
- Providing high-quality feedback about their use of Agents

- Participating in seasonal quests and specific tasks that are important to advance the ecosystem

These rewards will be distributed through seasonal airdrops, encouraging active participation and engagement within the Theoriq ecosystem.

6.3 Token Allocation and Distribution

The initial distribution of TAI tokens is designed to support the long-term growth and sustainability of the Theoriq ecosystem. The token allocation is as follows:

- Community: 50%
- Core Contributors: 15%
- Investors: 35%

The community allocation will be used to fund grants, bounties, retroactive public goods, network growth initiatives, and user airdrops for community rewards. These tokens will be distributed over time through various mechanisms to drive adoption and innovation within the ecosystem.

A significant portion of the circulating supply of tokens will be community-owned at launch, distributed by airdrop. Community airdrops will have a defined unlock schedule that releases tokens earlier than core contributor or investor allocations, ensuring that the community has a majority ownership stake in the protocol from the start.

7 Governance and Safety

As an open and decentralized platform, Theoriq recognizes the importance of effective governance and safety mechanisms to ensure the long-term success and stability of the ecosystem. Given the rapidly evolving nature of AI technology and the challenges associated with its development and deployment, Theoriq aims to establish a flexible and adaptable governance framework that can respond to the needs of the community and the broader AI landscape.

7.1 Iterative Governance Development

Theoriq recognizes that the optimal governance mechanisms for decentralized agentic AI are not yet known and may require significant experimentation and iteration to discover. As such, the platform is committed to a process of continuous learning and improvement, actively seeking input from the community and incorporating best practices from other successful projects.

User feedback plays a vital role in shaping the governance of the Theoriq ecosystem. Through various channels, such as community forums, surveys, and governance proposals, users can provide valuable insights and suggestions that help guide the development and refinement of the protocol's governance mechanisms. By actively engaging with the community and incorporating user feedback, Theoriq ensures that its governance model remains responsive, effective, and aligned with the needs and values of its users.

7.2 Safety and Security

In addition to effective governance, ensuring the safety and security of AI systems is a top priority for Theoriq. The platform incorporates a range of mechanisms to mitigate risks and promote responsible AI development, including:

1. Staking and slashing mechanisms to discourage malicious behavior
2. Robust security audits and bug bounties to identify and address vulnerabilities
3. Social dispute resolution and arbitration processes
4. Collaboration with leading AI safety researchers and organizations

User feedback and participation also play a crucial role in maintaining the safety and security of the Theoriq ecosystem. Users can report potential vulnerabilities, flag malicious or unethical behavior, and provide suggestions for improving the platform's safety measures. By fostering a culture of transparency, collaboration, and shared responsibility, Theoriq aims to create a secure environment that promotes trust and confidence among its users.

7.3 Community-Driven Safety Standards

Ultimately, the success of Theoriq's safety efforts will depend on the active engagement and participation of the community. The platform will work to foster a culture of responsibility and stewardship, encouraging community members to contribute to the development of safety standards, best practices, and risk mitigation strategies.

For instance, the community could propose and vote on a set of guidelines for creating Safe Agents, which would include requirements such as extensive testing, adherence to ethical principles, and the incorporation of fail-safe mechanisms. These guidelines could then be adopted as a standard for all Agents deployed within the Theoriq ecosystem, ensuring a consistent level of safety and reliability. Furthermore, the community could establish a dedicated Safety Council, composed of experts in AI safety, ethics, and governance, to oversee the implementation of these standards and provide ongoing guidance and support to developers and users.

Through open dialog, collaborative research, and shared learning, Theoriq aims to build a strong foundation for the safe and responsible development of AI technologies, setting an example for the broader AI ecosystem. By actively incorporating user feedback and promoting community-driven safety initiatives, Theoriq ensures that its approach to safety and security remains adaptive, responsive, and effective in the face of evolving challenges and opportunities.

8 Conclusion

Theoriq is poised to catalyze a new wave of progress in agentic AI by providing a flexible, extensible platform that prioritizes composability and interoperability. By offering deliberately flexible abstractions and powerful tools for creating composable and interoperable agent collectives, Theoriq ensures that its capabilities will evolve alongside the entire agentic AI ecosystem.

The three foundational pillars of Theoriq – agentic primitives, a market-driven innovation ecosystem, and built-in evaluation and optimization mechanisms – work together to create a powerful platform for agentic AI innovation. Theoriq's unique approach aligns developer incentives with user needs, fostering an environment where the best AI solutions can emerge and thrive. Through its deep integration of adaptable evaluation and optimization mechanisms, Theoriq aims to create a meritocracy of AI agents, where collectives can be automatically assembled based on a multitude of factors, ranging from objective performance metrics to subjective human preferences.

As the Theoriq ecosystem grows and evolves, we expect to see an explosion of innovative AI agents and collectives that push the boundaries of what is possible with agentic AI. By providing a platform that supports composability and extensibility, Theoriq enables developers to build upon each other's work, creating more effective, efficient, and accessible AI solutions for a wide range of real-world problems.

Moreover, Theoriq's commitment to investing in governance and safety positions it as a leader in the responsible development and deployment of agentic AI technologies. By actively engaging its community in the creation and enforcement of safety standards, and by incorporating robust security measures and privacy controls, Theoriq aims to create a trusted environment that promotes innovation while mitigating potential risks.

We invite you to explore Theoriq further and join us in building the future of decentralized AI. To learn more about the project, its roadmap, and how to get involved, please visit our website at theoriq.ai and follow us on social media at @TheoriqAI. Together, we can unlock the full potential of artificial intelligence and create a more open, collaborative, and impactful technological future.

References

[1] crewai, 2024. Accessed: 2024-06-18.

[2] Langgraph, 2024. Accessed: 2024-06-18.

[3] James Bergstra, Daniel Yamins, and David Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*, pages 115–123. PMLR, 2013.

[4] Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, et al. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. *arXiv preprint arXiv:2308.10848*, 2023.

[5] Jack Clark and Dario Amodei. Faulty reward functions in the wild, 2016. Accessed: 2024-06-18.

[6] Iason Gabriel et. al. The ethics of advanced ai assistants, 2024.

[7] Jacob R Gardner, Matt J Kusner, Zhixiang Eddie Xu, Kilian Q Weinberger, and John P Cunningham. Bayesian optimization with inequality constraints. In *ICML*, volume 2014, pages 937–945, 2014.

[8] Sirui Hong, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, et al. Metagpt: Meta programming for multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352*, 2023.

[9] Tula Masterman, Sandi Besen, Mason Sawtell, and Alex Chao. The landscape of emerging ai agent architectures for reasoning, planning, and tool calling: A survey. *arXiv preprint arXiv:2404.11584*, 2024.

[10] Eigen Labs Team. Eigen: The universal intersubjective work token, 2024. Accessed: 2024-05-16.

[11] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation framework. 2023.

[12] Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. Sotopia: Interactive evaluation for social intelligence in language agents, 2024.