Factlink: A Scalable & Self-Sustainable Truth Machine for Solana

Draft V0.2

June, 2025

Abstract

The foundational promise of blockchain technology—a single, global, permissionless financial system—remains constrained by the intrinsic inability of smart contracts to access and trust real-world data. While the optimistic oracle, pioneered by Universal Market Access (UMA)[1], has demonstrated a viable model for decentralized truth verification, its implementation on a high-cost, low-throughput blockchain introduces fundamental barriers to scalability and economic sustainability. The operational costs of dispute resolution on such networks often result in a net-negative financial function, necessitating external subsidies. This creates a protocol that is neither self-sufficient nor capable of scaling to meet the demands of a future global financial system.

This paper introduces Factlink, a next-generation optimistic oracle and Data Verification Mechanism (DVM) architected on Solana. Factlink is designed from first principles to solve the critical scalability and economic challenges that have limited its predecessors. By leveraging Solana's hyper-efficient transaction processing and a novel, state-optimized smart contract design, Factlink transforms the dispute resolution process from an unsustainable cost center into a profit function. This cash-flow positive economic model enables the creation of a protocol-owned treasury, which provides a sustainable foundation for progressively enhancing the oracle's economic security. Additionally, for ultra-high-value applications, Factlink plans to introduce optional identity verification for voters, ensuring the "Profit from Corruption < Cost of Corruption" (PfC < CoC) principle is practically enforceable.

1 The Unfulfilled Vison

1.1 The Blockchain Vision

The advent of blockchain technology heralded more than just a new form of digital currency; it presented the blueprint for a revolutionary financial and social infrastructure. The core proposition was, and remains: a single, global, and permissionless ledger for value, accessible to anyone with an internet connection. This technology promised to dismantle the arcane, siloed, and friction-laden rails of the traditional financial system, replacing them with a transparent, programmable, and universally accessible substrate.

The introduction of smart contracts on platforms like Ethereum elevated this vision from a simple peer-to-peer payment system to a world computer. Smart contracts—immutable, self-executing code on the blockchain—were poised to automate complex agreements, create novel financial instruments, and build autonomous organizations that could operate without the need for traditional intermediaries. One could envision a world where financial services were not gatekept by institutions but were available as open protocols, where risk could be managed with programmatic precision, and where capital could flow freely across borders to where it was most needed.

Yet, over fifteen years into the blockchain experiment, this grand vision remains largely unfulfilled. While the ecosystem has seen explosive growth in specific verticals, its penetration into the daily lives and core economic activities of the global population remains minimal. Compared to the billions of users engaging with centralized web applications daily, the user base of decentralized applications (dApps) is orders of magnitude smaller. The question, therefore, is not whether the promise was compelling, but what fundamental obstacles have prevented its realization. While challenges in user experience and transaction throughput have played a role, another key structural limitation lies at the heart of the issue: the blockchain's isolation from the real world.

1.2 The On-Chain / Off-Chain Chasm: The Oracle Problem

Blockchains owe their strength—security, immutability, and censorship resistance—to their deterministic and self-contained nature. In theory, a smart contract could query an external API for data. However, this approach introduces a critical vulnerability: the data source remains unverifiable. How can the network ensure that the API has not been compromised or is not deliberately providing false information? A single inaccurate data point, if accepted as truth, could irreparably undermine the consensus that forms the foundation of the entire system.

This fundamental challenge, known as the Oracle Problem, creates a profound divide between the on-chain realm of smart contracts and the off-chain world of real-world data where human activity unfolds. Without a secure and reliable mechanism to bridge this gap, the potential of smart contracts is severely limited. They remain confined to handling on-chain-native assets and logic, unable to engage with the vast array of economic, social, and political information that shapes our reality.

This singular issue is arguably one of the primary barriers preventing blockchain technology from realizing its full potential. To enable the development of advanced financial derivatives, parametric insurance products, prediction markets, and decentralized governance systems as originally envisioned, smart contracts must have access to a reliable stream of verifiable real-world data. The critical question then arises: who supplies this truth, and how can it be trusted? Relying on a single, centralized data provider reintroduces the very point of failure that blockchain technology seeks to eliminate, rendering the notion of a "decentralized" application misleading. Therefore, a truly decentralized solution to the Oracle Problem is not merely a supplementary feature; it is an essential prerequisite for fulfilling the transformative promise of blockchain technology.

1.3 Price-Feed Model Oracles

The first wave of successful oracle solutions, led by protocols like Chainlink[2] and Pyth[3] on Solana, focused on solving the most immediate and commercially pressing need of the decentralized finance (DeFi) ecosystem: reliable price data for crypto and traditional financial assets. These systems operate primarily as "price-feed" oracles. Their model involves a decentralized network of node operators who source data from numerous off-chain sources, aggregate it, and "push" a validated price onto the blockchain at regular intervals.

This architecture, particularly as exemplified by Chainlink, is a sophisticated, multi-stage process involving Reputation Contracts to vet nodes, Order-Matching Contracts to farm out requests, and Aggregating Contracts to validate and reconcile answers. This robust design has been immensely successful, securing billions of dollars in value and forming a foundational pillar of DeFi. However, while optimized for providing continuous data streams, this model is fundamentally ill-suited for the broader purpose of a universal truth machine. Its limitations include:

- **Push-based, not Pull-based Architecture:** The model broadcasts predetermined data feeds rather than enabling permissionless, on-demand queries. There is no generalized mechanism for users to ask novel questions like "Did cargo ship XYZ arrive in the Port of Singapore by its so and so contracted date?"

- **Economic Barriers to New Data Sources:** While technically open, launching new feeds requires coordinating with node operators, establishing contracts, and setting up economic incentives - making it prohibitively expensive for niche or one-time use cases.
- **Structural Mismatch for Event Verification:** The multi-contract, aggregation-heavy process is designed for continuous price streams, not discrete event verification. It's economically unviable to establish dedicated feeds for every potential question, leaving a vast "long tail" of verifiable data unserved.

While essential for DeFi, price-feed oracles solve only a slice of the Oracle Problem. They do not provide a generalized, permissionless mechanism for anyone to bring any verifiable truth onto the blockchain.

1.4 The Conceptual Breakthrough: Optimistic Framework

A conceptual leap in solving the Oracle Problem came with the introduction of the "optimistic oracle", implemented by the Universal Market Access (UMA)[4] protocol. Instead of relying on a network of nodes to proactively push data, the optimistic model flips the script: it assumes any data asserted on-chain is true unless it is disputed.

This framework operates on a simple but powerful "assert-dispute" cycle.

1. **Permissionless Assertion:** Anyone can propose an answer to any query and back their assertion with a financial bond.
2. **Liveness Period:** A challenge window opens, during which any other participant can dispute the assertion by posting their own bond.
3. **Dispute and Escalation:** If a dispute occurs, the question is escalated to a decentralized data verification mechanism (DVM). UMA token holders vote on the correct outcome, using a Schelling Point game to incentivize honest, independent voting.
4. **Resolution:** The party that is voted to be correct is rewarded with a portion of the loser's bond. If no dispute occurs within the liveness period, the original assertion is accepted as truth, and the asserter is rewarded.

This design was a breakthrough because it created a truly permissionless system for truth verification. For the first time, a developer building an insurance protocol for weather events in a specific county or a user creating a prediction market on a niche political outcome could access a decentralized truth-finding mechanism without needing prior approval from a centralized entity.

The success of this model is best exemplified by its role in powering Polymarket, one of the largest and most successful prediction markets in the ecosystem. Many users of Polymarket may not realize that the complex and often contentious process of resolving market outcomes is arbitrated by UMA. UMA performs the difficult, behind-the-scenes work of achieving decentralized consensus on real-world events, proving that the optimistic model is not just theoretical but practically viable for securing significant economic value. UMA deserves immense credit for pioneering this architecture and proving its product-market fit.

1.5 The Implementation Bottleneck and the Factlink Thesis

Despite the conceptual elegance and proven viability of the UMA model, the DVM's implementation on the Ethereum mainnet has exposed fundamental architectural and economic limitations that prevent it from achieving true global scale and long-term sustainability. While

UMA proved the *what* and the *why* of a generalized oracle, its choice of *where* has created a bottleneck. The core problems stemming from this architectural choice are threefold (as will be detailed in Section 3):

1. **Prohibitive Operational Costs:** Dispute resolution, the system's core security function, requires every participating DVM voter to submit multiple transactions to the Ethereum blockchain. The gas fees associated with this process are substantial and volatile, meaning the aggregate cost of resolving a single dispute can range from hundreds to tens of thousands of dollars.
2. **Unsustainable Economic Model:** Because these operational costs are so high, the protocol is forced to subsidize its own operation through direct gas rebates and inflationary rewards without any underlying earnings. This means the DVM, the very heart of the protocol, operates at a significant negative gross margin.
3. **A Theoretical Security Guarantee:** UMA's whitepaper posits a core security guarantee: that the Profit from Corruption (PfC) will always be less than the Cost of Corruption (CoC). However, the mechanisms proposed to enforce this are not programmatically implemented on-chain and are impractical in a competitive market. The protocol's security relies more on its market capitalization than on an enforceable, self-correcting economic model.

This paper puts forth the thesis that the optimistic oracle model is correct, but it requires an environment where its core functions can operate profitably and at scale. The limitations of UMA are not flaws in the optimistic concept itself, but rather symptoms of a mismatch between the protocol's design, its on-chain implementation, and its underlying blockchain infrastructure.

Factlink solves these implementation bottlenecks. By building a new optimistic oracle and DVM from the ground up on the Solana blockchain, Factlink retains the game-theoretical elegance of UMA's model while leveraging an infrastructure that is orders of magnitude cheaper and more performant. This enables, for the first time, an optimistic oracle that is scalable, economically self-sustaining, and possesses a credible path to verifiable security. This paper will detail the architecture of Factlink, provide a data-driven analysis of its economic superiority, and argue that it represents a critical step toward fulfilling the promise of Blockchain.

2 The Optimistic Oracle and Data Verification Mechanism (DVM) Model

To comprehend the innovations presented by Factlink, it is essential to understand the foundational architecture upon which it builds. The optimistic oracle, as pioneered by UMA, represents a paradigm shift from the node-based, price-feed model. This section will deconstruct the core principles of this canonical model, including its game-theoretical underpinnings, which are critical for appreciating both its strengths and the limitations.

2.1 The Optimistic Principle: Assumed Truthfulness

Traditional oracle systems operate on a principle of proactive, pessimistic verification. They assume that data is unknown or untrusted until a quorum of nodes actively fetches, validates, and reports it on-chain. This process, while robust for its intended purpose, is resource-intensive and continuous. It requires constant work to maintain the state of truth, regardless of whether that truth is actively being questioned.

The optimistic oracle inverts this logic. It operates on the principle of assumed truthfulness with the right to challenge. The system's default state is to "optimistically" assume that any data proposed to it is correct. This simple inversion has profound implications for efficiency. It eliminates the need for constant, proactive validation for every piece of data. Instead, the expensive and resource-intensive work of verification is reserved only for instances where there is an explicit disagreement.

This model is predicated on an economic and game-theoretical understanding of human behaviour. It posits that in a system with properly aligned incentives; most participants will act honestly most of the time. The network's resources should therefore be focused on resolving the exceptional cases of malice or error, rather than treating every data point as a potential threat. This results in a system that is quiescent and inexpensive during normal operation, only activating its powerful dispute resolution machinery when necessary.

Empirically, as of 22' Jun '25, UMA has recorded 68648 assertions[5], and 1369 disputes[6], resulting in a dispute to query ratio of 1.99%. This observation shows that, with right incentives, the overwhelming majority of assertions are indeed honest.

2.2 The Canonical Architecture: The Assert-Dispute-Settle Lifecycle

The optimistic principle is implemented through a clear, multi-stage process that governs the lifecycle of any data request from its inception to its final resolution. This lifecycle can be broken down into four primary phases:

1. **Query and Reward:** The process begins when a user or smart contract raises a query. This is a request for a specific piece of verifiable information, such as "What was the closing price of AAPL on NASDAQ on December 31, 2025?" or "Did the candidate from Party X win the 2028 U.S. Presidential Election?". The entity raising the query also posts a reward, which serves as an incentive for participants to provide a correct answer.[1]

2. **Assertion and Bonding:** In response to a query, any participant can step forward to become an asserter. The asserter proposes an answer and, crucially, posts a bond. This bond is a financial stake that acts as collateral, signalling the asserter's confidence in their answer. By putting their own capital at risk, asserters are disincentivized from providing false or frivolous information. Once an assertion is made, a liveness period begins. This is a pre-defined challenge window (e.g., 24-48 hours) during which the assertion is considered pending.[1]

3. **Dispute and Escalation:** During the liveness period, any other participant who believes the assertion is incorrect can become a disputer. To do so, they must also post a bond, equal in value to the asserter's bond. The act of disputing immediately halts the optimistic process. The query is no longer considered pending; it is now officially disputed. The question of truth cannot be resolved optimistically and must be escalated to a more robust, definitive arbitration mechanism.

4. **Settlement:** The final outcome depends on whether a dispute occurred:
   - **Undisputed Settlement:** If the liveness period expires without any disputes, the system's optimistic assumption holds. The initial assertion is accepted as the final, correct answer. The asserter's bond is returned to them, and they claim the reward offered by the query owner.
   - **Disputed Settlement:** If a dispute was raised, the query is settled according to the final verdict of the escalation mechanism. The party deemed correct (whether the

[1] This describes the canonical model similar to UMA's Optimistic Oracle V2. While the architecture has since evolved in implementations like UMA's V3, where the query raiser can also be the initial proposer of truth, with others able to dispute it, the core principle of "assert-dispute-escalate" remains constant across versions.

asserter or the disputer) receives their original bond back, claims the reward, and is awarded a portion of the losing party's bond. The losing party forfeits their bond as a penalty for submitting an incorrect value, with the remainder often claimed by the protocol as a dispute resolution fee.

This elegant lifecycle creates a powerful economic dynamic. Honest asserters are rewarded for providing valuable information. Honest disputers are rewarded for policing the system and correcting inaccuracies. Malicious or incorrect participants are financially penalized, making dishonest behaviour an economically irrational strategy.

2.3 The Arbiter of Last Resort: The Data Verification Mechanism (DVM)

When a dispute is escalated, it is sent to the Data Verification Mechanism (DVM). The DVM is the system's ultimate arbiter of truth, a decentralized supreme court for data. In the canonical model established by UMA, the DVM is composed of the protocol's native token holders. These token holders are enlisted as a decentralized jury to vote on the correct outcome of the disputed query.

The DVM's task is to reach a definitive consensus on what the "truth" is. This is not a simple majority vote. To prevent collusion and ensure that voters are incentivized to determine the objective truth independently, the DVM employs a sophisticated voting scheme based on a game-theoretical concept known as a Schelling Point.

2.4 The Schelling Point and the Commit-Reveal Scheme

The concept of a Schelling Point, introduced by Nobel laureate Thomas Schelling, describes a solution that people tend to choose by default in the absence of communication. It is the focal point or "obvious" answer that individuals, acting independently, will converge upon. For example, if two people need to meet in New York City on a specific day but cannot communicate the time or place, a likely Schelling Point would be "noon at the Grand Central Station information booth." It is the most common-sense, focal answer.

In the context of an oracle, the "truth" is the Schelling Point. The DVM is designed to incentivize every voter to independently research the query and vote for what they believe is the objectively correct answer, assuming that other honest voters will do the same. The system rewards voters who are part of the majority consensus and penalizes those in the minority. This creates a powerful incentive to vote for the truth, as it is the most likely focal point for all other honest voters.

To ensure the integrity of this process and prevent voters from simply waiting to see how others are voting before casting their own ballot, the DVM employs a commit-reveal scheme:

1. **Commit Phase:** During the initial voting period, voters do not submit their actual vote. Instead, they submit a hash of their vote combined with a secret random number (a "salt"). This hashed commitment is recorded on-chain. It acts as a digital lockbox, proving that a vote has been cast without revealing its content. Since all commitments are just cryptic hashes, no voter can know how others have voted.

2. **Reveal Phase:** After the commit phase has closed, the reveal phase begins. Voters now submit their actual vote along with the salt they used to create the hash. The DVM's smart contract can then verify that the revealed vote, when hashed with the provided salt, matches the commitment made during the previous phase. This guarantees that voters could not have changed their vote after seeing how the commitments were stacking up.

This commit-reveal process is a cornerstone of decentralized voting mechanisms. It breaks the flow of information between voters, forcing each participant to make their decision independently. By combining this cryptographic scheme with the economic incentives of the Schelling Point game, the DVM creates a robust environment for achieving decentralized consensus on subjective or complex real-world data. It is this combination of optimistic efficiency and game-theoretically secured dispute resolution that made the UMA model a landmark achievement in the ongoing quest to solve the Oracle Problem.

3 Limitations of current Optimistic Oracle Implementation

While the optimistic oracle model represents a conceptual breakthrough, its practical implementation within a first-generation blockchain environment like Ethereum reveals significant structural weaknesses. These are not flaws in the optimistic theory itself, but rather consequences of an architectural mismatch between a high-frequency, cost-sensitive mechanism and a low-throughput, high-cost ledger. The reliance on Ethereum's mainnet for dispute resolution creates a cascade of challenges that fundamentally limit the model's scalability, economic sustainability, and, ultimately, the credibility of its security guarantees. This section provides a detailed, data-driven analysis of these limitations, using UMA's implementation as the primary case study.

3.1 The Scalability and Cost Barrier

The security and decentralization of the DVM are predicated on the active participation of a large and distributed set of voters. However, the commit-reveal process, while cryptographically sound, imposes a significant transactional burden. Each voter must submit at least two separate transactions to the blockchain for every single disputed query: one to commit their vote and another to reveal it. When the settlement layer is a high-cost network like Ethereum, the economic consequences are severe.

To quantify this, we can model the cost of a single dispute. Based on empirical analysis of UMA's DVM contract interactions on Ethereum from May 1st '25 to May 31st '25[7], a commit transaction on average consumed 618253.38 gas whereas a reveal transaction on average consumed 392514.12 gas. So, the gas per voter per dispute roughly totals 1,000,000 gas. UMA rebates voters who have stake more than 500 UMA tokens. In May '25, there were 276 unique voters who were rebated.

The total cost of a single dispute can be calculated based on gas prices and the price of Ethereum, as shown in the table below. Costs are estimated using an ETH price of $2,250.

Network State	**Median Gas Price**	**Total Dispute Cost (w/ 276 Voters)**	**Total Dispute Cost (w/ 1,000 Voters)**
Uncongested	0.5 Gwei	∼$310	∼$1,125
Moderate Congestion	5 Gwei	∼$3,105	∼$11,250
High Congestion	25 Gwei	∼$15,525	∼$56,250

**Costs calculated with an ETH price of $2,250.*

These figures are alarming when contrasted with the typical revenue generated from a single dispute, which is derived from the losing party's bond and often amounts to only ∼$250. The stark reality is that the DVM's core security function operates at a negative gross margin, with costs exceeding revenue by one to two orders of magnitude.

This economic inversion forces the protocol into an unsustainable position. It cannot pass these exorbitant costs onto voters, so it must subsidize its own operation. This is done through direct gas rebates and unsustainable inflationary token rewards. Analysis of UMA's public data reveals the scale of these subsidies: in 2024 alone, over \$570,000[8] was paid out in voter gas rebates, in addition to the \$576,000[8] disbursed between 2020 and 2023.

The risk this model presents becomes clear during periods of network congestion or high usage. During the U.S. election cycle in November and December 2024, when Polymarket activity surged, UMA paid out \$275,000[8] in rebates in just two months that too with moderate network congestion (median gas prices of 12 Gwei in Nov'24 and 8 Gwei in Dec'24). This demonstrates that as the oracle and its dependent applications scale, these subsidy costs will spiral, placing ever-increasing pressure on the token's value and the protocol's long-term viability. This architectural choice creates a hard ceiling on the system's potential scale.

3.2 The PfC < CoC Fallacy

The cornerstone of UMA's security philosophy is the inequality: Profit from Corruption (PfC) < Cost of Corruption (CoC). This asserts that the system is secure as long as the potential profit from manipulating an outcome is less than the cost to corrupt the DVM. However, this elegant equation breaks down under real-world conditions, transforming from a security guarantee into a demonstrable fallacy.

As of June 21, 2025, the Total Value Secured (TVS) by UMA's Optimistic Oracle stands at \$670,850,833[9], which serves as a proxy for the system-wide PfC. Additionally, the TVS tied to the UMA token, including oSnap (their governance protocol), reaches \$1,252,660,005[9].

In contrast to TVS, the Cost of Corruption (CoC), defined as the capital required to acquire 51% of the voting power, is approximately \$70 million[10]. This figure is derived from the protocol's fully diluted value of \$137 million[10]. Alarmingly, the TVS of the Optimistic Oracle alone is nearly 10 times the CoC, while the total value secured across all UMA-related systems is over 20 times the CoC, representing a direct and severe inversion of the required security guarantee ($PfC < CoC$).

This critical vulnerability persists because the mechanisms designed to enforce the inequality are either impractical or unimplemented, as explained further. While the $PfC < CoC$ inequality appears theoretically sound as a security guarantee for UMA, its practical implementation is deeply flawed. The protocol's ability to maintain this critical balance relies on mechanisms that are not only unimplemented but also economically and competitively infeasible.

- **Untracked Profit from Corruption (PfC):** For the inequality to be actively managed, the protocol must have a reliable method to measure the PfC, which represents the total financial value at risk tied to a specific oracle query. Although UMA's smart contracts include a placeholder for this PfC value, there is currently no on-chain mechanism to automatically track or update it[4]. Calculating the PfC for a complex ecosystem like Polymarket—with thousands of open positions across hundreds of markets—is a challenging data aggregation task that is performed off-chain.
- **The Impracticality of Reactive Fees:** Even if a mechanism to track PfC were in place, the proposed solution of dynamically levying higher fees on systems that increase the value at risk remains unimplemented. The idea is that, for example, if Polymarket's total value locked (TVL) grows to a point where the potential profit from corruption approaches the Cost of Corruption (CoC), UMA would charge Polymarket a percentage of its TVL as a fee. This fee would theoretically be used to buy back and burn UMA tokens, increasing the token's price and thereby raising the CoC. However, this reactive fee mechanism is economically and competitively unviable. No protocol integrator would accept a system where core operational costs could unpredictably spike by arbitrary and exorbitant amounts. It is telling that this mechanism has not been activated, likely due to the understanding that its implementation would severely undermine adoption.

Given these critical shortcomings, the $PfC < CoC$ guarantee does not function as a dynamic, self-correcting property of the system in practice. Instead, the protocol's security relies on the brute-force deterrent of its current market capitalization rather than the elegant equation it promotes. While this static defense may suffice to deter attacks on most low-value markets today, it is not the robust, programmatic security model that was promised. Furthermore, it weakens during market downturns and may prove insufficient to safeguard the multi-billion-dollar ecosystems UMA aims to secure in the future.

3.3 Protocol Unsustainability and Negative Cash Flow

The confluence of the high operational costs and the fallible security model leads to a damning conclusion about the protocol's economic design. From a first-principles financial perspective, the value of any productive asset is the sum of its discounted future cash flows. A protocol, like a business, should be designed to generate more revenue than it costs to operate.

The UMA DVM, however, has been operating with a negative gross margin on its core product: dispute resolution. As demonstrated, the cost of resolving a dispute far exceeds the direct fee revenue it generates. The protocol is therefore in a state of perpetual operating loss, sustained only by aforementioned subsidies.

This has serious implications for the long-term value of the protocol's native token. If the system's primary function is a cost center, the token cannot derive intrinsic value from the protocol's economic activity. Its value becomes dependent on secondary, speculative factors rather than productive cash flow. A system that relies on subsidies rather than generating intrinsic revenue is fundamentally fragile. UMA proved the optimistic oracle concept, but in doing so, it also proved that the Ethereum mainnet is the wrong foundation on which to build a scalable and enduring verifiable world, unless the cost of a transaction on Ethereum becomes $10 - 100$x cheaper.

4 Factlink: A Solana-Native, Low-Cost, Self-Sufficient Solution

The limitations of first-generation optimistic oracles are not an indictment of the optimistic principle itself, but rather a clear demonstration of the critical importance of the underlying execution environment. Factlink is born from this understanding. It is an implementation of the optimistic oracle and DVM framework, re-architected to leverage the unique capabilities of the Solana blockchain. By aligning its design with a high-throughput, low-cost ledger, Factlink resolves the fundamental bottlenecks of its predecessor and introduces a model that is not only scalable but economically self-sufficient from its inception.

4.1 Design Philosophy

The core design philosophy of Factlink began with two non-negotiable principles:

1. **Hyper-Scalability:** The system must be able to handle a massive increase in both optimistic assertions and DVM disputes without a corresponding linear increase in cost or a degradation in performance. It must be built for a future where millions of queries are resolved annually, not a few thousand.
2. **Economic Self-Sufficiency:** The protocol's core functions must be cash-flow positive. It must generate more revenue from its operations than it costs to run them. The protocol should not be reliant on external subsidies for its core operation, scalability and long-term survival. It must be a self-sustaining economic engine.

These principles led us to two foundational architectural choices. The first was the selection of Solana as the execution and settlement layer. The second was the design of a novel, state-optimized smart contract architecture that minimizes on-chain storage and thus costs and a robust economic model that establishes intrinsic protocol revenue from day one.

4.2 The Solana Advantage

Solana's architecture offers a radically different cost and performance profile compared to Ethereum's mainnet. This is not an incremental improvement but a categorical shift that fundamentally changes the economics of running a DVM. The advantages can be analysed in two key areas: transactional efficiency and storage optimization.

4.2.1 Transactional Efficiency

As previously analysed, the cost of a DVM dispute on Ethereum is dominated by gas fees. Solana's fee market operates on a completely different scale. The median transaction fee on Solana is typically a fraction of a cent (∼$0.001)[11]. This is possible due to Solana's fee structure, which consists of a low, fixed base fee (5,000 lamports) and an optional priority fee. Based on internal testing, Factlink's commit, reveal, and claim transactions consumed on average ∼75000 Compute Units. This low computational requirement means transactions can be reliably processed with minimal or zero priority fees, even during periods of moderate network activity. While optimizations like durable nonces could be used to schedule transactions and mitigate fees during extreme congestion, they are unlikely to be necessary for normal operation.

Let us re-run the DVM cost model from Section 3.1, this time for Factlink on Solana. Assuming each of the 276 voters submits three transactions (commit, reveal, and claim) at Solana's median cost of $0.001[11] per transaction:

- Cost per Voter: 3 transactions * $0.001/transaction = $0.003
- Total Cost for 276 Voters: 276 * $0.003/voter = $0.828

Comparing this to the Ethereum-based model reveals a staggering difference. A dispute that costs anywhere from $310 to $15,525 on Ethereum would cost less than $1 on Solana. This represents a 3 to 5 order-of-magnitude reduction in cost.

This dramatic reduction has profound implications. It means that voter participation no longer requires external subsidies. The cost is so negligible that it can be easily absorbed by the protocol's intrinsic revenue streams. It unlocks the ability to scale the number of voters to enhance decentralization or to scale the number of queries served by the system, all while maintaining positive unit economics. It effectively eliminates the single greatest barrier to the scalability of the optimistic oracle model.

4.2.2 Storage Optimization and Rent Model

Beyond transaction fees, the cost of maintaining state on a blockchain can be significant on Solana. UMA stores extensive data on-chain, consuming valuable and expensive block space. However, we should note that, in UMA's case, this cost is included along with the gas we calculated earlier, on Ethereum.

Factlink's architecture is designed to be lean and state-efficient.

- **Verifiable History via Events:** Factlink's smart contracts are designed to minimize on-chain storage. Instead of persisting the entire history of votes for a query within its primary account, the contract emits detailed events for every crucial action (e.g., `VoteCommitted`, VoteRevealed). These events are an immutable part of the blockchain's history and can be queried by off-chain indexers. This design provides full transparency and allows any third party to audit or reconstruct the complete history of a dispute, without requiring the protocol to pay for the permanent on-chain storage of that data. The on-chain accounts only need to store the current state and final aggregated results, dramatically reducing their size and cost.

- **Solana's Rent Mechanism:** Solana's "rent" model requires accounts to hold a minimum SOL balance proportional to their data size. A key feature is that this rent is fully refunded when an account is closed. Factlink leverages this extensively. Each voter only needs to deposit approximately 0.07216 SOL[12] (∼$10 at a $150 SOL price) to create a voter account. With just this voter account, they will be able to vote in over 225 disputes in every single DVM round. Notably, even this $10 deposit is not absolutely necessary; compared to the dispute rate on UMA, this amount could be reduced to less than $1. However, we have kept it at $10 to ensure that even if our scale increases to 100x its current level, voters need not increase their account size. Crucially, this deposit is not a fee; it is fully refunded should a voter decide to stop voting, close their account, and exit the system. This creates a highly capital-efficient model for participation.

By combining Solana's low transaction fees with a state-efficient architecture and aggressive use of the rent-refund mechanism, Factlink achieves a level of operational efficiency that is unparallelled in comparison to existing solutions.

4.3 A Self-Sustaining Economic Model

This radical cost reduction is not an end in itself; it is the foundation upon which a truly sustainable economic model can be built. Because Factlink's operational costs are negligible, it can generate significant positive cash flow from its core functions. This "protocol revenue" creates a virtuous cycle, funding the system's security and growth without relying on unsustainable subsidies.

4.3.1 Intrinsic Profitability of the Factum DVM

The Factum DVM, Factlink's dispute resolution mechanism, is designed to be a profit center. When a dispute occurs, both the asserter and the disputer post a bond. The losing party forfeits their bond. In Factlink, a portion of this forfeited bond (e.g., 50%) is distributed to the winner, and the remaining portion is captured by the protocol as a dispute fee.

Consider a typical dispute with a $500 bond:

- Total bond at stake: $1,000 ($500 from asserter, $500 from disputer)
- Dispute fee (50% of one bond): $250
- DVM operational cost: ∼$1(Assuming low congestion)
- Protocol Gross Profit per Dispute (including voter fee reimbursement): ∼$249

This stands in stark contrast to the UMA model, which has, all the while, incurred net loss on all disputes till date[8]. For every dispute it resolves, the Factlink protocol accumulates capital in its treasury. This revenue stream is directly proportional to the usage and contention within the ecosystem, turning the DVM's core function into a source of strength and value accrual.

4.3.2 Reward Fees

Relying solely on dispute fees means the protocol only generates revenue during contentious events. To create a more consistent and predictable cash flow, Factlink introduces a novel mechanism called the Reward Fee.

When a user raises a query, they specify a reward to incentivize an assertion. The Factlink protocol automatically takes a small, configurable percentage of this reward as a fee. For example, if a user offers a 10 USDC reward and the reward fee is set to 50%, the protocol treasury receives 5 USDC.

While small on an individual basis, this fee is applied to every single query raised through the system, regardless of whether a dispute occurs. As the oracle gains adoption and thousands or millions of queries are processed, this creates a steady, predictable, and scalable revenue stream for the protocol.

4.3.3 The Treasury Flywheel

The combination of dispute fees and reward fees directs a consistent flow of capital into a protocol-owned treasury. This treasury forms a potent tool to maintain the $PfC < CoC$ inequality in the long run should we achieve scale. However, the flywheel is a not an immediate or an infallible solution.

The CoC is a function of the market value of the voting tokens required to overtake the network. A treasury funded by real protocol revenue can be deployed by the Factlink DAO for activities that directly increase the CoC, such as using treasury funds to purchase Factlink tokens on the open market.

This creates a powerful Treasury Flywheel:

1. Increased protocol usage generates more fees with the treasury growing with real capital.
2. The funds held in the DAO treasury increase the intrinsic value of the voting coin, increasing the Cost of Corruption.
3. Enhanced security attracts high-value integrations, restarting the cycle.

While this self-reinforcing loop makes Factlink's security model dynamic, its effectiveness is contingent on scale and time. The treasury must accumulate tens or hundreds of millions of dollars over multiple years to become a formidable defense against well-capitalized attackers targeting multi-billion-dollar contracts. In the protocol's early days, the revenue generated will not be sufficient to meaningfully raise the CoC in response to a sudden spike in PfC from a new, high-value integration.

This pragmatic understanding of the flywheel's limitations is a critical insight. It highlights that for ultra-high-value markets, purely economic deterrents may require fortification. This acknowledgement directly motivates the need for the "Progressive Fortification" strategies outlined in Section 7, such as an optional pseudonymous voter identity verification mechanisms.

5 Factlink System Architecture

The Factlink protocol is a synergistic composition of two main components: the Factlink Optimistic Oracle (OO)[13], which serves as the public-facing interface for queries, and the Factum Data Verification Mechanism (DVM)[14], the decentralized judicial system for resolving disputes. These components are designed to interact seamlessly, providing a complete, end-to-end solution for verifiable data on Solana.

5.1 High-Level Overview

The process begins when any on-chain entity—be it a decentralized application (dApp) or an individual user—requires verifiable real-world data. This requester initiates the process by submitting a query to the Factlink Optimistic Oracle.

The system operates on an "optimistic" principle. An answer is proposed along with a bond and, if it remains unchallenged for a specified liveness period, it is accepted as correct with the asserter receiving the reward and his original bond. This "happy path" provides a fast and cost-effective resolution, with the OO delivering the verified answer directly to the requester.

However, if any participant believes the asserted answer is incorrect, they can stake a bond to dispute it. This act of dispute automatically pauses the optimistic process and escalates the query to the Factum Data Verification Mechanism (DVM) for a definitive resolution. The DVM, a decentralized court of token holders, then votes to determine the correct outcome. The DVM's binding verdict is relayed back to the Optimistic Oracle, which then settles the query, accordingly, rewards the honest parties, and delivers the final, validated data to the original requester.

5.2 The Factlink Optimistic Oracle (OO)

The Factlink OO is the entry point to the system. It is a program on Solana that manages the entire lifecycle of a data query up to the point of a dispute and then for settlement as well. Its architecture is defined by its components, lifecycle, and incentive structures.

5.2.1 Key Components

- **Admin Properties:** A set of configurable parameters controlled by the protocol's governance (initially, the founding team, transitioning to a DAO). These include the addresses of key contracts (like the DVM), the basis points for fees, and other system-wide settings.
- **Whitelists:** To maintain quality and security, the OO utilizes several on-chain whitelists:
  - **Query Type Whitelist:** Standardizes the format of questions (e.g., `YES_NO_QUERY`, `PRICE_QUERY`, etc) to ensure assertions and resolutions are handled consistently.
  - **Mint Whitelist:** Defines the SPL tokens (e.g., USDC, SOL) that are permissible for use as bonds and rewards, preventing the use of low-liquidity or spam tokens.
  - **Escalation Manager Whitelist:** Lists the approved dispute resolution contracts. While the Factum DVM is the default, this provides future flexibility to integrate other specialized resolution mechanisms.
- **Query Accounts:** Each query creates a dedicated Program-Derived Account (PDA) on Solana. This account stores all relevant information for the query: its hash, the bond and reward details, the liveness period, the identity of the asserter and disputer (if any), and its current state (e.g., Pending, Disputed, Settled).

5.2.2 The Query Lifecycle

The process within the OO follows the canonical optimistic model, optimized for Solana:

- **Raise:** A user pays the rent for a new Query Account and submits the query details, including the reward, bond requirement, and a CID pointing to detailed specifications on IPFS. The novel Reward Fee is collected by the protocol at this stage.

- **Assert:** An asserter provides an answer and deposits the required bond into an escrow account controlled by the OO program. The query's expiration time is set.
- **Dispute:** Before the expiration time, a disputer can challenge the assertion by depositing their own bond. The query's state is updated to Disputed, and an instruction is prepared to escalate it to the DVM.
- **Escalate:** The OO formally submits the disputed query to the Factum DVM for resolution. The OO then awaits a callback from the DVM with the final verdict.
- **Settle:** Upon receiving the verdict from the DVM (or upon the expiration of the liveness period if undisputed), the OO settles the query. It distributes the bonds and reward according to the outcome and records the final, resolved value in the Query Account.
- **Close:** Once settled, the original query owner can close the on-chain accounts associated with the query, reclaiming the SOL that was deposited for rent.

5.3 The Factum Data Verification Mechanism (DVM)

The Factum DVM is the ultimate source of truth for the Factlink ecosystem. It is a highly optimized, stake-weighted voting system built on Solana.

5.3.1 Key Components

- **Admin Properties:** Like the OO, the DVM has a set of governable parameters, including the designated voting token, phase lengths for voting rounds, and the crucial slashing penalties for incorrect or non-participating voters.
- **Voter Accounts:** To participate, users must create a Voter Account. This account tracks their total staked token amount, their active delegations, their voting history for the current round, and their claimable rewards.
- **Delegation:** Factum DVM supports a secure delegation model. A token holder can stake their assets from a secure cold wallet while delegating the ability to perform voting actions (commit, reveal, claim) to a more convenient hot wallet. The delegate has no authority to move or unstack the principal assets, providing both security and operational flexibility.
- **Round Structure:** The DVM operates in continuous, sequential rounds. Each round is a discrete period (e.g., 48 hours) during which a batch of disputed queries is processed.

5.3.2 The Dispute Resolution Process: The Four Phases

Each DVM round is divided into four distinct, time-gated phases to ensure an orderly and fair voting process:

- **Commit Phase:** At the start of a new round, the batch of disputed queries is finalized. During this phase, registered voters (or their delegates) submit the hashed commitments of their votes for each query in the round. Staking and unstaking requests are also processed during this phase.
- **Reveal Phase:** Once the Commit Phase ends, the Reveal Phase begins. Voters now submit their plaintext votes and the corresponding salts. The DVM smart contract verifies that the revealed vote matches the on-chain commitment. All valid, revealed votes are tallied, and the stake-weighted totals for each potential outcome are calculated.
- **Claim Phase:** After the Reveal Phase, the outcome for each query is determined based on whether the voting thresholds (e.g., minimum participation and majority percentage)

were met. Voters who were on the winning side of a resolved query can now submit a transaction to claim their share of the voting rewards. Voters who voted incorrectly or failed to reveal their vote will be slashed.

- **Configure Phase:** This is a final, automated administrative phase. The DVM officially settles the resolved queries, communicating the verdicts back to the Factlink OO. Any queries that failed to reach a resolution may be rolled over to the next round (up to a maximum limit). The system is prepared for the next round, which begins immediately as the configure phase ends.

This highly structured, four-phase process, executed on Solana's low cost and high-speed infrastructure, allows the Factum DVM to reliably and efficiently process a large number of disputes.

6 Comparative Analysis

This section places Factlink and UMA, in a head-to-head analysis across three critical domains: dispute resolution cost, economic model sustainability, and overall system scalability. The data reveals that Factlink's design is not merely an incremental improvement but a fundamental re-platforming that yields orders-of-magnitude enhancements in efficiency and long-term viability.

6.1 Dispute Cost Analysis

The most consequential point of divergence between Factlink and UMA is the cost of executing their core security function: DVM dispute resolution. As established, this process requires each participating voter to submit transactions to the blockchain (a minimum of commit and reveal on UMA and a commit, reveal, and claim on Factlink). The cost of these transactions dictate the economic feasibility of the entire system.

The following table models the aggregate cost to resolve a single dispute with a reasonably decentralized voter set of 1,000 participants. For UMA, we use an empirically derived gas consumption of ∼1,000,000 gas per voter on Ethereum. For Factlink, we use a conservative cost estimate of three transactions per voter on Solana:

Metric	**UMA (Ethereum)**	**Factlink (Solana)**	**Difference Magnitude**
Low Congestion: ETH @ $2,250, Gas @ 0.5 Gwei[15] — SOL Tx @ $0.001[11]			
Cost Per Voter	$1.13	$0.003	∼375x
Total Dispute Cost	$1,130	$3	∼375x
Med Congestion: ETH @ $2,250, Gas @ 5 Gwei[15] — SOL Tx @ $0.01[11]			
Cost Per Voter	$11.25	$0.03	∼375x
Total Dispute Cost	$11,250	$30	∼375x
High Congestion: ETH @ $2,250, Gas @ 25 Gwei[15] — SOL Tx @ $0.02[11]			
Cost Per Voter	$56.25	$0.06	∼938x
Total Dispute Cost	$56,250	$60	∼938x

The implications of this data are profound and multifaceted:

- **Cost Volatility vs. Predictability:** The operational cost of UMA's DVM is subject to the extreme volatility of the Ethereum gas market. This makes the protocol's subsidy burden itself unpredictable and potentially explosive during periods of high network activity. Factlink's costs, by contrast, are stable and almost negligibly low, providing a predictable and reliable foundation for its operations.

- **The Inevitability of Subsidies:** No rational economic system can sustain a core operational function that costs thousands or tens of thousands of dollars while generating only a few hundred in direct revenue. The data in Table 6.1 proves that UMA's reliance on inflationary rewards and treasury-funded gas rebates is not a temporary growth strategy but a permanent, structural necessity.
- **The Barrier to Decentralization:** The security of a DVM is directly related to the number and distribution of its voters. UMA faces a difficult trade-off: increasing the number of voters to enhance security directly inflates its operational costs and subsidy burden. Factlink faces no such constraint. It can scale its voter set with a negligible impact on its bottom line, allowing it to pursue maximum decentralization without bankrupting the protocol.

In summary, the choice of the underlying ledger is the single most important architectural decision for an optimistic oracle. Factlink's selection of Solana fundamentally transforms the cost structure of dispute resolution from an unsustainable liability into a trivial operational expense.

6.2 Comparative Economic Model Analysis

Feature	**UMA**	**Factlink**
Primary Revenue Stream	Dispute Fees (from loser's bond).	Dual Streams: Dispute Fees (from loser's bond) + Reward Fees (from every query).
Secondary Revenue	None.	Protocol can earn yield on its treasury assets.
Protocol Profitability	Net-Negative. DVM costs have far exceeded dispute fee revenue. Reliant on external subsidies.	Cash-Flow Positive. Negligible DVM costs ensure dispute fees are profitable. Reward fees provide consistent revenue.
Token Value Proposition	Belief in future potential; Governance rights; Speculation.	Productive Asset. A claim on a share of real, sustainable protocol revenue. Value is enhanced by the Treasury Flywheel.
PfC < CoC Enforcement	Theoretical. Relies on unimplemented, impractical reactive fees and static market capitalization.	Theoretical in the short to medium term. "Treasury Flywheel" might help reach sustainability in the long run—should we reach scale.

6.3 Scalability & Throughput

Beyond cost, the ability of the system to handle a high volume of concurrent activity is paramount for long-term success. The most direct measure of scalability is the raw transaction throughput of the underlying blockchain. Solana consistently processes around 1,500[16] true transactions per second (TPS), whereas Ethereum's capacity is closer to 15 TPS[17]—a 100x difference in baseline performance.

At present, this disparity may not seem critical. Let's consider a highly optimistic scenario where Factlink processes 50,000[18] queries annually. Assuming a 2% dispute rate, this would result in 1,000 disputes. With a voter base of 276 (as seen in May), and two transactions per voter (commit and reveal), the total DVM load would be 552,000 transactions for the entire year.

- On Ethereum, processing this load at 15 TPS would require approximately 36,800 seconds (about 10.2 hours). This represents just 0.1% of the network's total block time available in a year.
- On Solana, the same load at 1,500 TPS would be cleared in just 368 seconds (about 6 minutes), consuming a negligible 0.001% of its annual capacity.

Clearly, the DVM's current operational load does not pose a significant bottleneck on either chain. However, our architecture is designed not for today, but for mass adoption. If the platform grows to 100x this scale, the DVM would demand over 10% of Ethereum's entire annual block space, creating severe network congestion and making dispute resolution prohibitively expensive. On Solana, this 100x growth would still only consume 0.1% of the network's capacity, allowing it to scale gracefully without becoming a bottleneck.

In conclusion, the comparative analysis is definitive. Across the critical vectors of cost, economic sustainability, and scalability, Factlink's Solana-native design provides not just an incremental improvement but a categorical leap forward. It takes the proven game theory of the optimistic model and places it on a foundation that can actually support its global ambition.

7 Future Work and Progressive Fortification

While Factlink's initial design establishes a new benchmark for scalable and economically sustainable oracles, our work is guided by a long-term vision. The pursuit of a system capable of securing a global financial infrastructure is a process of continuous adaptation. We recognize that as the value secured by Factlink grows, so too will the sophistication of potential threats. Our roadmap is therefore built on a philosophy of Progressive Fortification: a pragmatic, community-governed approach to enhancing security in response to the protocol's maturation and the evolving threat landscape.

7.1 Acknowledging the Limits of Security

Factlink's primary security guarantee is its cash-flow positive economic model and the resulting Treasury Flywheel, designed to ensure the Cost of Corruption (CoC) outpaces the Profit from Corruption (PfC) in the long run. However, we are pragmatic in acknowledging that this system has theoretical limits, especially when confronted with sudden ultra-high-value-integrations or extreme market conditions.

The $PfC < CoC$ inequality, while robust under normal conditions, faces potential stressors where it could invert. It is critical to identify these scenarios and plan for stronger defense mechanisms. These stressors include:

- **Severe Market Downturns:** In a prolonged bear market, the price of Factlink's native token could fall significantly, drastically reducing the CoC (the cost to acquire a 51% stake). However, it should be noted that value secured by the oracle—especially in contracts denominated in stablecoins or tied to real-world assets—may not decrease proportionally and could be deployed to mitigate such scenarios.
- **Sudden High-Value Integration:** A large-scale protocol could integrate with Factlink, causing the PfC to spike by billions of dollars overnight. The Treasury Flywheel, which

increases the CoC incrementally through sustained revenue, cannot react fast enough to a near-instantaneous change in the value at risk.

- **Irrational Adversaries:** The economic model assumes a rational actor seeking financial profit. It is less effective against a well-capitalized entity whose goal is not profit, but systemic disruption. For such an adversary, the financial "cost" of an attack may be an acceptable expense to achieve a larger strategic objective.

These stressors make it clear that for Factlink to mature into an oracle securing tens of billions of dollars in value, a final, non-economic backstop is a prudent and necessary component of its long-term design.

7.2 The Path to Fortification

To address the threat posed by ultra-high-value markets and sophisticated adversaries, we propose a future, optional fortification layer that introduces a powerful, real-world deterrent: voter ID validation.

7.2.1 An Optional, Opt-In Identity Layer

We envision a future protocol enhancement that integrates a decentralized identity (DID) solution, such as those being developed by Civic or other identity protocols on Solana. Crucially, this would not be a mandatory, system-wide requirement. Imposing identity verification on all users would violate the core permissionless ethos of the protocol.

Instead, this would be an opt-in feature that a query creator could enable for their specific, high-value query. This creates a tiered security model:

- **Standard Tier:** The default for all queries. Security is guaranteed by Factlink's robust and self-sustaining economic incentives. Voting remains pseudonymous.
- **Fortified Tier:** An optional setting for ultra-high-value queries (e.g., a multi-billion dollar insurance bond). The query creator can specify that only DVM participants who have voluntarily verified their identity are eligible to vote on that specific dispute. In return, these voters can expect a higher staking reward for their participation, capitalized by query raisers.

7.2.2 Redefining Cost of Corruption for Malicious Actors

The introduction of an optional identity layer is designed to deter one specific behaviour: intentional, malicious fraud with the intent to corrupt the oracle for financial gain. It is not intended to penalize honest differences of opinion or good-faith disagreements that can arise in complex disputes.

For an attacker participating in a fortified vote, this mechanism fundamentally redefines the Cost of Corruption. The CoC is no longer a finite, calculable market price for acquiring tokens. It expands to include the unquantifiable cost of potential prosecution. This ensures that even if a cabal of malicious actors could amass large pools of capital to break the DVM, they would be unwilling to do so when it means exposing their identities to potential legal jeopardy.

7.2.3 Limitations of Legal Deterrents

We also recognize that even this powerful deterrent is not absolute. A legal backstop is only effective against actors who are subject to a functional and cooperative legal jurisdiction. However, by implementing this layer, we dramatically raise the bar, effectively neutralizing the threat from the vast majority of potential high-capital attackers who are not prepared to risk their liberty and assets.

7.3 A Commitment to Progressive Decentralization and Governance

Factlink is committed to a path of progressive decentralization. The role of the founding team will systematically diminish over time, transferring stewardship of the protocol to the Factlink DAO, composed of its token holders.

The introduction of any fortification mechanism would represent a pivotal decision in the protocol's evolution. Such a proposal would not be implemented unilaterally. It would be subject to a rigorous and transparent process managed by the Factlink DAO, utilizing established frameworks like Solana Governance or Realms. The community would be entrusted to debate the trade-offs: the immense security benefits of enabling legal recourse for high-value markets versus the principles of privacy.

This community-governed approach ensures that Factlink remains adaptable and resilient, capable of evolving to meet future threats while staying true to the collective will and risk tolerance of its stakeholders. The DAO will be the ultimate arbiter of the protocol's path forward.

8 Building the Bridge to a Verifiable World

The optimistic oracle, while conceptually brilliant, has been economically constrained by its implementation on a high cost blockchain. This created a fundamental paradox where the core security function—dispute resolution—became an unsustainable liability, placing a hard ceiling on the model's scalability.

Factlink resolves this implementation crisis. By re-architecting the oracle and DVM from first principles on Solana, we transform the dispute mechanism from a financial drain into a positive cash-flow engine. The significance of this extends beyond superior architecture; it unlocks the oracle's true potential. By providing a scalable and sustainable bridge for verifiable data, Factlink serves as the foundational infrastructure for the next generation of decentralized applications.

This enables a future where the blockchain is no longer an isolated digital island but is deeply interwoven with the fabric of our global economy. It empowers builders to create the applications originally envisioned—from synthetic instruments that allow anyone to hedge against any risk, to prediction markets that drastically reduce noise and bias in the information we consume, etc.

Factlink is designed as a foundational public good: a secure, stable, and community-governed substrate upon which this new, more verifiable world can be built.

9 References

[1] UMA Protocol (2025) Universal Market Access. Available at: `https://uma.xyz/`.

[2] Juels, A., Breidenbach, L., Cachin, C., et al. (2021) Chainlink 2.0: Next Steps in the Evolution of Decentralized Oracle Networks. Available at: `https://research.chain.link/whitepaper-v2.pdf`.

[3] Pyth Data Association (2023) Pyth Network Whitepaper. Available at: `https://pyth.network/Pyth_Network_Whitepaper_v2.pdf`.

[4] Lambur, H. (2018) UMA - A Decentralized Financial Contracts Platform. Available at: `https://github.com/UMAprotocol/whitepaper/blob/master/UMA-whitepaper.pdf`.

[5] Risk Labs (2025) UMA Total OO Assertions. Dune Analytics. Available at: https://dune.com/queries/3609321/6081548.

[6] UMA Protocol (2025) UMA Protocol Dashboard. Dune Analytics. Available at: https://dune.com/uma_protocol/uma-protocol.

[7] UMA Protocol (2025) VoterGasRebateV2.ts script. GitHub. Available at: https://github.com/UMAprotocol/protocol/blob/master/packages/affiliates/gas-rebate/VoterGasRebateV2.ts. (Modified the referenced script to calculate the gas consumed for commit and reveal transactions for May'25. The modified script and results will be posted on Factlink's GitHub (https://github.com/factlinkoracle)).

[8] UMA Protocol (2025) UMA Gas Rebates. Google Sheets. Available at: https://docs.google.com/spreadsheets/d/18zT93_cFyTMLSFKksWOOivXbN88FQ6dMU6D3Us3Xpqw.

[9] Risk Labs (2025) UMA - Total Value Secured. Dune Analytics. Available at: https://dune.com/risk_labs/uma-total-value-secured.

[10] CoinMarketCap (2025) UMA Price, Chart, and Market Cap. Available at: https://coinmarketcap.com/currencies/uma/.

[11] Ilemi (2025) Solana Transaction Fee Tracker. Dune Analytics. Available at: https://dune.com/queries/3921548/6592789.

[12] gaussproof (2021) 'Answer to "What is the cost to store 1kb, 10kb, 100kb worth of data into the solana blockchain?"', Solana Stack Exchange, 29 December. Available at: https://solana.stackexchange.com/a/2549.

[13] Factlink (2025) Optimistic Oracle (OO) Working. Factlink Documentation. Available at: https://docs.factlink.xyz/oo-working.

[14] Factlink (2025) Data Verification Mechanism (DVM) Working. Factlink Documentation. Available at: https://docs.factlink.xyz/dvm-working.

[15] Glassnode (2025) Ethereum: Median Gas Price. Glassnode Studio. Available at: https://studio.glassnode.com/charts/fees.GasPriceMedian?a=ETH.

[16] Solscan (2025) Solana Explorer. Available at: https://solscan.io/.

[17] Etherscan (2025) Ethereum Block Explorer. Available at: https://etherscan.io/.

[18] Risk Labs (2025) UMA Total OO Assertions. Dune Analytics. Available at: https://dune.com/queries/3609321/6081548.