Incentiv

INCENTIV WHITEPAPER V2.0

UPDATED ON 14 July, 2025

Incentiv

Table of Contents

Incentiv

Legal Disclaimers and Risk Warnings

Informational Nature
This White Paper is provided solely for general informational purposes. It is not intended to form the basis of, and should not be relied upon in connection with, any contractual or investment decision.

No Offer or Solicitation
Nothing herein constitutes, or shall be construed as, an offer to sell, a solicitation of an offer to buy, or an invitation to subscribe for, purchase, or otherwise acquire any securities, commodities, digital assets, or other financial instruments in any jurisdiction. This document is not, and should not be interpreted as, a prospectus or offering memorandum.

No Financial, Legal, or Tax Advice
The contents of this White Paper do not constitute advice of any kind—including but not limited to financial, legal, investment, accounting, or tax advice. Prospective participants should conduct independent due diligence and consult their professional advisers before engaging with the Incentiv network or acquiring any $CENT tokens.

Regulatory and Jurisdictional Considerations
Digital-asset regulations vary widely among jurisdictions and are subject to evolving interpretation. Access to the Incentiv network or to $CENT tokens may be restricted or prohibited in certain regions. It is the sole responsibility of each individual to ensure full compliance with all applicable laws, regulations, and restrictions in the relevant jurisdiction(s).

Risks and Volatility
Participation in the Incentiv ecosystem and any transaction involving $CENT tokens entail significant risks—including, without limitation, market volatility, technology failure, cybersecurity threats, loss of private keys, and possible changes in legal or regulatory environments. No guarantee is made regarding token value, network adoption, or return on any form of participation.

Forward-Looking Statements
This White Paper contains forward-looking statements that reflect current expectations and projections about future events. Such statements involve known and unknown risks and uncertainties and are not guarantees of future performance. Actual results may differ materially from those anticipated. The Incentiv team undertakes no obligation to update forward-looking statements.

Iterative Development
The Incentiv protocol—including the Incentiv+ Engine, $CENT token economics, technical roadmap, and governance model—remains under active development. Specifications, features, and timelines described herein are subject to change as research progresses, technology evolves, and community feedback is incorporated. All material changes will be communicated promptly and transparently through official channels.

Limitation of Liability
The authors, contributors, and affiliated entities disclaim all liability for any direct, indirect, incidental, consequential, or other losses arising from or related to the use of, or reliance on, the information contained in this document. All actions taken on the basis of this White Paper are undertaken at the sole risk of the reader.

By continuing to read or use this White Paper, you acknowledge that you have read, understood, and accepted the foregoing disclaimers and agree to be bound by them.

Incentiv

1. Executive Summary

Incentiv was conceived to bridge critical gaps in today's blockchain landscape. Traditional blockchains often **reward power and capital over actual contribution**, leading to unsustainable economies and concentrated benefits. Users face fragmented, technical experiences, and many networks rely on short-term speculation or inflationary rewards that erode long-term value. Incentiv's vision is to **create a sustainable, high-utility blockchain** that flips this paradigm. It aims to align incentives across all participants and deliver a seamless user experience, so that every interaction with the network contributes to a virtuous cycle of growth. By making the system intuitive and **human-centric** through innovations like account abstraction, and by economically rewarding meaningful participation, Incentiv addresses both the usability and incentive problems head-on.

Advanced Account Abstraction (AAA) – Usability as a First-Class Feature

Incentiv incorporates **Advanced Account Abstraction** (**AAA**) at the protocol layer. AAA introduces programmable smart-accounts that:

- abstract away **gas and fees management** (multi-token fee payment, sponsored transactions, batch execution);
- enable **richer security models** (social recovery, session keys, automated spending limits); and
- support **one-click onboarding** flows familiar to Web-2 users.

By shifting complexity from the user interface into account logic, AAA removes much of the friction that has kept mainstream users on the sidelines. It also provides developers with a powerful design space for consumer-grade applications.

Integrated Economic Model & Unified Reward Pool

At the heart of Incentiv's design is an integrated economic engine (the **Incentiv+ engine**) that links every on-chain action to shared value creation. Instead of isolated fee markets or miner-only rewards, Incentiv introduces a **unified reward pool** that collects a portion of **transaction value and fees** from every block into a communal fund. This pool is then redistributed transparently to contributors based on their **role and real contributions**, as reflected in their scoring.

Miners, developers, bundlers, liquidity providers, and everyday users all earn from the same pool according to a multi-dimensional scoring system tailored to each role's impact. This **role-based contribution model** ensures that value flows to those who actually drive network utility – whether by securing the chain, improving the ecosystem, providing liquidity, or simply engaging. By rewarding **proportional value creation**, the network incentivizes behaviors that strengthen the network, creating a self-reinforcing flywheel where each block's activity feeds the pool, which then **pays out to active participants and encourages further usage and innovation**.

Early-Stage Acceleration and Network Bootstrapping

To jump-start this positive cycle, Incentiv employs an early-stage reward acceleration strategy. A significant reserve of **$CENT** tokens (the **Community Rewards** allocation) is pre-loaded into the reward pool as an initial subsidy. This ensures **meaningful rewards from day one**, even before transaction volume is high. In practical terms, early miners, developers, and users are generously compensated for their contributions, which **bootstraps network activity and engagement** when it's most critical. This unified early incentive gives every stakeholder a strong financial reason to participate at the outset, seeding a robust community and demonstrating the network's value proposition. Additionally, a Short-Term Growth Fund is available to selectively bolster key activities or strategic initiatives in the launch phase. By **subsidizing usage in the beginning** in a meaningful and efficient way, Incentiv overcomes the cold-start problem that many new networks face, attracting hash power, developer talent, liquidity, and user activity to establish network effects rapidly.

Transition to Fee-Based Sustainability

Crucially, Incentiv's economic model is designed to gracefully evolve from this subsidy-driven start to a self-sustaining, fee-funded system. The early token incentives follow a **tapering schedule** – for example, the community reward allocations are released on a hyperbolic decay curve over the first few years. As network usage grows, **organic fee revenue** steadily replaces the token subsidies in the reward pool. During mid-growth stages, transaction fees might begin covering a substantial share of rewards, reducing reliance on reserves while still maintaining attractive payouts. Network participants are gradually acclimated to a fee-supported model without experiencing sudden drops in rewards. Eventually, all rewards will be **100% fee-driven** – meaning every $CENT distributed to contributors will be sourced from real usage of the network rather than inflation. This planned transition happens smoothly and predictably, with no cliffs, allowing contributors to anticipate how the mix of token subsidy vs. fees changes over time, planning their involvement for both short and long-term gains.

Long-Term Alignment and Sustainable Value

The end-state of Incentiv's model is a blockchain economy that sustains itself on its own utility. The **$CENT token** has a fixed supply, so once the initial allocations for growth have been distributed, no new tokens are minted. All participants' stakes in $CENT are preserved from inflation, and their value grows only through the genuine expansion of network usage and demand. This creates a powerful long-term alignment across all roles. Miners, developers, users, and other contributors share a unified incentive to make the platform useful and attractive, since **the prosperity of each group depends on the overall health of the ecosystem**. By uniting these interests, Incentiv avoids the zero-sum pitfalls of earlier crypto economies. Instead, it delivers a positive-sum framework where **contributing to the network's growth is the optimal strategy for every participant**. The combination of a **next-generation user experience** (through advanced account abstraction and features like paymasters) with this robust economic loop results in a high-utility platform built for longevity. In summary, Incentiv's integrated infrastructure and economic design provide a clear path from launch to maturity.

Incentiv

2. Introduction & Vision

Blockchain technology has made great strides in decentralization and security, yet it still suffers from critical gaps that limit mainstream adoption. Most blockchains today reward power and capital rather than genuine contribution, offer fragmented and technical user experiences, and fail to align incentives across the full range of network participants. Users face complex wallets, fragmented fee structures, and economic models that prioritize short-term speculation over sustainable growth.

Incentiv is an **EVM-compatible Layer 1 blockchain** protocol designed to solve these challenges by aligning economic rewards with meaningful contribution, and making blockchain technology intuitive, accessible, and human-centric. It combines the security of Proof-of-Work consensus with the flexibility of native **Advanced Account Abstraction (AAA) and Incentiv+** - an economic engine that **rewards behavior and performance rather than just capital or computation.**

At its core, Incentiv rethinks how value flows through a blockchain, built around three foundational pillars:

- **Proportional Value Distribution:** A transaction-value fee flows into a shared reward pool and is algorithmically distributed, using a multi-dimensional scoring framework tailored to each role among miners, developers, bundlers, liquidity providers, and users. Weighted by genuine on-chain contribution, these rewards create a self-reinforcing flywheel - each block injects fresh capital that sparks innovation, deepens liquidity, and draws in further activity, accelerating the cycle of growth.
- **Advanced Account Abstraction (AAA):** Incentiv makes programmable smart-contract wallets the default, while legacy Externally Owned Accounts (EOAs) remain compatible. Paymasters enable any wallet pay fees in any token, and an intent-centric interface lets bundlers turn a user's goal into an optimised transaction. Wallets can batch multiple actions into one UserOperation, enforce policies such as social recovery, spending caps, session keys, or scheduled payments, and still enjoy permissionless access. Together these features deliver a seamless user experience, on par with modern web applications, without sacrificing decentralization.
- **Contribution-Scored Proof-of-Work:** Incentiv preserves the core security and permissionless nature of Proof-of-Work, where miners secure the network through computational effort. However, block rewards are not fixed—they grow with network activity and are distributed from a shared reward pool. This pool's size reflects real economic usage, such as transaction volume and network fees. This creates a situation where electricity is spent on blocks that carry real user value, not empty hashes, and honest, efficient operators earn more than wasteful ones.

These three pillars run on a modular core - an **EntryPoint** contract that processes every user operation, an **Epoch Engine** that resets scores and distributes rewards on predictable time-cycles, and an implementation of a **DAO-driven governance** for future parameter tuning and upgrades.

By aligning incentives across all key contributors and streamlining the user experience, Incentiv transforms everyday activity into a common engine of growth, making open finance and decentralised applications accessible to a far wider audience.

This white paper maps out the architecture, economic mechanics, and design rationale behind Incentiv. It shows how aligning rewards with contribution yields a human-centered blockchain where security meets efficiency and usability meets decentralization.

3. Economic Philosophy

3.1. Contribution, Not Capital

Incentiv's economic philosophy is built on a simple premise — contribution, not capital, forms the base layer of value in the network.

The protocol treats its economy as a dynamic feedback system where value generated on-chain flows into a communal reward pool and is then redistributed according to each participant's on-chain measurable contributions.

In practice, every transaction feeds the shared pool through fees, and those funds cycle back out to active contributors rather than accumulating as passive profit for capital alone. By making **"proof of contribution"** the basis for rewards, Incentiv ensures that value must be created to be earned – discouraging rent-seeking and speculative hoarding. This is a stark departure from the stake-to-earn or lottery-style mining paradigms, which rewarded mere ownership or luck; instead, Incentiv's model channels all incentives toward productive participation. In doing so, it aligns each participant's success with the network's success, creating a regenerative, positive-sum cycle in place of zero-sum competition.

3.2. Challenging Legacy Paradigms

Underlying this philosophy is a critique of the legacy economic paradigms that blockchains have often inherited. Traditional systems and many first-generation crypto economies were anchored in artificial scarcity and a need for infinite growth. As Galbraith (1958)[1] and Daly (1996)[2] warned, artificially imposed scarcity tends to concentrate wealth at the top, while pursuing perpetual growth at all costs leads to resource depletion and systemic instability.

More recently, Raworth (2017)[3] has critiqued these models as "degenerative by default," arguing that they exclude and exhaust rather than include and regenerate. Such paradigms also encourage rent-seeking, where actors profit by controlling assets or tolls rather than by creating new value. Unfortunately, many blockchain protocols unwittingly replicate these flaws: for example, Bitcoin's strict supply cap enforces scarcity and incentivizes hoarding, and other networks that chase raw growth often see economic power centralized in the hands of a small group, that its interest are not necessarily aligned with the long term interest and goals of the network and other participants. **Incentiv breaks from these precedents**, instead fostering an economy where growth emerges from shared value creation and where wealth is a by-product of contributing to the commons rather than exploiting it.

3.3. Aligning Incentives for Commons Stewardship

In designing a contribution-centric economy, Incentiv also addresses the classic **"The Tragedy of the Commons"** described by Hardin (1968)[4], wherein individuals acting in self-interest can collectively deplete a shared resource. Traditional solutions to this dilemma have been heavy-handed (imposing regulations from above) or exclusionary (privatizing the commons) – approaches antithetical to open, permissionless systems. **Incentiv takes a third path** by leveraging blockchain's ability to align individual incentives with collective stewardship. By tying each participant's reward to the real and proportional value they add to the network, the protocol makes preserving and enhancing the common resource (the network's utility and economic health) the most rational and rewarding strategy for each actor. In effect, what was once a fatal dilemma is transformed into a virtuous cycle of mutual gain; contributions to the commons grow the overall reward pool, which then pays out proportionally and spurs further participation.

3.4. Positive-Sum Dynamics and Game-Theoretic Foundations

This feedback loop creates a genuinely positive-sum environment where individual competition to contribute value ends up reinforcing the prosperity of all. Self-interest and the common good cease to be at odds – they become one and the same. In game-theoretic terms, the Incentiv economy is structured so that **cooperation is the equilibrium strategy**: the incentive design makes contributing honestly the best response for each participant given that others do the same, essentially encoding a Nash equilibrium that maximizes collective benefit (Nash, 1950)[5]. Over time, behaviors

that create more value for the network are consistently rewarded and thus spread, much like an evolutionary dynamic in which successful strategies propagate (Taylor & Jonker, 1978)[6]. At the same time, insights from behavioral economics (Kahneman & Tversky, 1979)[7] inform the system's parameters to ensure that these incentives resonate with real human decision-making and perceptions of fairness. The result is a carefully calibrated mechanism where individual rationality and psychological motivations are harnessed to drive participants toward positive-sum behavior.

3.5. Concrete Funding Mechanisms

Crucially, this economic philosophy is realized through concrete protocol mechanisms. Every block contributes to the communal reward pool: standard **gas fees** and a **value fee** on each transaction are collected and funneled into the pool, effectively turning every on-chain action into an investment in the network's health. To jump-start the ecosystem, a short-term allocation of tokens is embedded to boost the reward pool in the early phases. This subsidy tapers off gradually as organic network activity – and the fees derived from it – ramp up. By design, as usage grows the fee contributions come to dominate the pool, allowing the network to smoothly transition to self-sustainability with fees from real economic activity becoming the primary reward source. In the long run, all rewards distributed are backed by actual on-chain value. **In short, the economy is funded by the network's own productive throughput – a closed-loop design in which value circulated on-chain is continuously recycled to fuel further participation.**

3.6. Equitable Allocation and Epoch Resets

Incentiv's reward distribution mechanism ensures those pooled funds go to the right places – **to those who genuinely drive the network forward**. The protocol continuously measures on-chain activity across all stakeholder roles and computes a contribution score for each participant. Miners, developers, users, bundlers, and liquidity providers are each evaluated by role-specific metrics that capture their unique contributions. These metrics are transparently recorded and weighted in a multi-dimensional scoring algorithm that determines how the reward pool is split. Contribution scores are **recorded and then reset at the end of every epoch**, preventing winner-take-all dynamics and ensuring that **performance is rewarded only so long as it continues**.

3.7. Transparent Parameters and Adaptive Governance

All economic parameters governing these processes are set **transparently from the start**. Key variables (such as base transaction fee rates, reward-curve gradients and epoch length) are pre-defined at launch and visible to all, providing a clear and predictable rulebook for the economy. While the system begins with pre-determined parameters, **Incentiv's governance framework** can adjust them over time, allowing the community to fine-tune fees, reward weightings, and cadence to preserve the protocol's health and equitability, achieving adaptability without sacrificing accountability.

3.8. Contribution as the Beating Heart

Ultimately, **measurable contribution is elevated to the network's primary currency of value and recognition**. By explicitly rewarding real on-chain value creation above all else, Incentiv builds an economy that is inclusive, cooperative, and self-reinforcing. Contributors are not just fueling growth; **they are the growth**, rewarded in direct proportion to their impact. This shift from extractive to regenerative design underpins every technical component that follows. Measurable contribution is the beating heart of Incentiv's economy, proving that a blockchain can achieve security and scalability **without sacrificing fairness or sustainability**—building an open economy that works for everyone and grows stronger with every meaningful contribution.

4. Protocol Architecture Overview

Incentiv's architecture is **modular** and **EVM-compatible**, built as a Layer 1 blockchain that combines **Advanced Account Abstraction** (AAA) with a **Proof-of-Work** (PoW) consensus mechanism.

The design follows a clear separation of concerns by dividing core blockchain operations, economic scoring, and fee routing into discrete, upgradeable components. This unified framework integrates technical and economic subsystems through feedback loops, aligning incentives across all participants instead of treating consensus, economics, and applications as isolated layers.

4.1. Architectural Design Principles

The protocol architecture is guided by several key design principles:

- **Modular Composition:** Each component has clear boundaries and interfaces, allowing independent upgrades without compromising overall system integrity.
- **Feedback Integration:** Components share economic signals, enabling the system to adapt to participant actions without any centralized coordination.
- **Contribution Measurement:** On-chain activity is tracked to feed the incentive distribution system, turning technical operations into economic signals that influence how rewards are allocated.
- **Intent-First Interaction:** Enabling users to specify desired outcomes (intents) rather than low-level transaction parameters, abstracting away complexity without sacrificing user control.

These principles ensure that technical functionality and economic incentives work in concert to create a responsive, aligned network. Each principle directly reflects the philosophical foundations from Section 2 by translating ideas like contribution-centric value creation, proportional rewards, and human-centered design into concrete architectural features.

4.2. Core Components

Incentiv's architecture consists of several integrated components, each responsible for specific protocol functions. Below is a high-level overview of each component (with detailed discussions in later sections of the paper):

4.2.1. Advanced Account Abstraction Layer (Section 5)

The Advanced Account Abstraction (AAA) layer is the universal interface for all user interactions with the network. It replaces traditional externally owned accounts (EOAs) with smart-contract wallets and introduces an intent-based transaction model.

Key elements of this layer include:

- **EntryPoint Contract:** The protocol-native gateway that receives and processes user operations.
- **UserOperations:** Intent-based transaction objects that encapsulate a user's desired actions (potentially bundling multiple actions into one atomic transaction).
- **Smart-Contract Wallets:** Programmable accounts (smart wallets) that serve as the default account type for users, while still supporting legacy EOAs for compatibility.
- **Bundlers:** Specialized externally operated network nodes that collect, validate, and bundle UserOperations into transactions, often aggregating multiple operations for atomic execution.

- **Paymasters:** Optional service providers that sponsor or facilitate gas fees, enabling flexible gas payment models (for example, allowing fees to be paid in various tokens).

This AAA layer fundamentally transforms how users interact with the blockchain, enabling enhanced security, flexible multi-token fee payments, and rich programmable transaction logic by default.

4.2.2. Consensus Layer (Section 6)

The consensus layer secures the network through Proof-of-Work mining, ensuring permissionless participation and decentralized block production.

This layer components include:

- **Ethash Mining:** A memory-hard PoW algorithm (Ethash) used for block production, ensuring decentralization through broad participation in mining.
- **Enhanced Fork Choice:** The chain selection follows the longest-chain rule based on cumulative work, as in classic PoW systems, ensuring objective and verifiable consensus.
- **Block Validation:** Standard full validation of blocks and transactions, enforcing correctness of state transitions and execution results.

While the consensus mechanism itself follows the proven PoW model, Incentiv introduces **enhanced economic incentives outside of fork choice**, rewarding miners not only for block production. This economic layer encourages long-term alignment with network health , without altering the core PoW fork choice logic.

4.2.3. Economic Engine: Incentiv+ (Section 8)

The **Incentiv+** engine serves as the protocol's economic brain, measuring contributions and distributing rewards based on those contributions.

Its main components are:

- **Scoring System:** Role-specific metrics that evaluate each participant's on-chain contributions (with different metrics for miners, developers, users, bundlers, liquidity providers, etc.).
- **Reward Pool:** A mechanism that collects transaction value fees (a value-based fee on each transaction) into a shared pool and redistributes these funds as rewards.
- **Gradient Model:** A continuous reward multiplier system that ranks participants by performance and assigns reward multipliers on a sliding scale (higher performance relative to peers earns a higher multiplier compared to using fixed tiers).
- **Epoch Controller:** A timing system that defines epochs (e.g. weekly cycles) and synchronizes scoring and reward distribution at the end of each epoch.

This component translates raw on-chain activity into meaningful contribution metrics, ensuring that rewards are aligned with actual participation and driving the protocol's incentive alignment.

Execution Environment

The execution environment is the underlying layer that processes transactions and maintains blockchain state. It provides a familiar runtime with the following features:

- **EVM Compatibility:** Full support for Ethereum-compatible smart contracts and opcodes, making it easy for developers to build and deploy using existing tools.
- **State Management:** Maintenance of account balances, contract storage, and network parameters in the global state, just as in other EVM-based chains.
- **Gas System:** A gas metering mechanism to price computational resources and enforce resource limits per transaction (to ensure fair usage of network capacity).
- **Native DEX:** An integrated decentralized exchange that can facilitate on-chain token swaps, used for converting fees or providing liquidity within the protocol.

This execution environment offers a seamless developer and user experience similar to Ethereum's, while also supporting Incentiv's unique features such as the AAA model and value-fee mechanism.

System Workflow and Interaction

All of the above components work together in a continuous process that handles transactions, measures contributions, and distributes rewards. The system's operation can be understood in two interconnected cycles: the per-transaction flow and the epoch-based reward cycle.

Transaction Flow

Each user-initiated transaction (expressed as a UserOperation) flows through the system as follows:

1. **User Operation Creation:** A user creates a **UserOperation** describing their intended action. This could include one or multiple bundled actions that must execute atomically (all-or-nothing).
2. **Submission to Bundlers:** The UserOperation is submitted to network **bundlers** via a specialized mempool (separate from the normal transaction pool) dedicated to user operations.
3. **Bundling and Execution Call:** A bundler validates the UserOperation (checking signatures, nonce, etc.) and packages it into a transaction that calls the **EntryPoint** contract.
4. **Mining into a Block:** A miner running the PoW consensus then includes the bundler's transaction (which contains the user's operation) into a new block.
5. **Validation via Wallet Logic:** When the block is executed, the **EntryPoint** contract invokes the target smart-contract wallet specified in the UserOperation. The wallet's custom logic verifies the operation's validity (e.g. correct paymaster, sufficient funds, valid signatures/policies).
6. **Execution of Action:** Upon successful validation by the wallet contract, the intended user action is executed (e.g. transferring funds, invoking a contract call, etc.).
7. **Fee Handling:** Gas costs and the value-based fees are calculated during execution. The appropriate fees are deducted (potentially via a paymaster) and routed to the **reward pool** as part of the protocol's fee recycling mechanism.
8. **Event Emission:** The transaction's outcome is recorded, and events are emitted for transparency (for example, logs of actions taken, fees paid, etc.).

Epoch Cycle

In parallel to the day-to-day transaction processing, Incentiv operates on an **epoch-based cycle** for scoring and reward distribution. Over each epoch (a fixed number of blocks or a time period, e.g. one week), the following occurs:

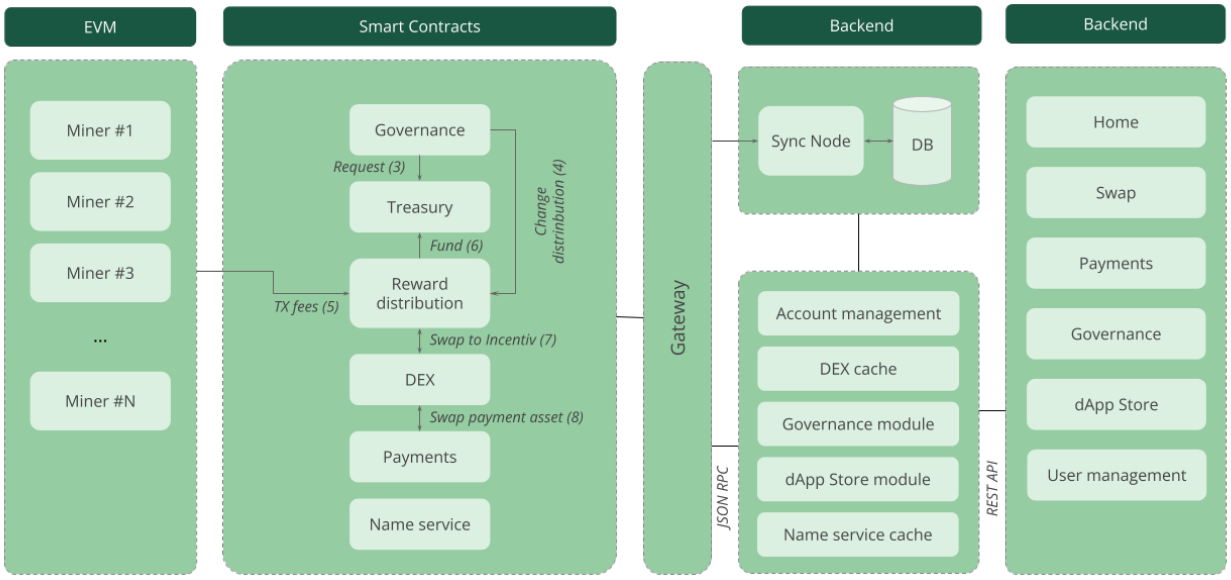
1. **Epoch Timing:** The **Epoch Controller** tracks the passage of time in terms of blocks and defines the boundaries of each epoch.
2. **Ongoing Contribution Measurement:** Throughout the epoch, the activities of all participants (miners, bundlers, developers, users, etc.) are monitored and measured according to the scoring system. Each participant accumulates points or metrics based on their contributions in their respective roles.
3. **End-of-Epoch Scoring:** At epoch close, the scoring system normalizes and aggregates these contribution metrics. Participants are ranked within each role, determining their **gradient position** (relative performance rank). This data is then packaged into an **attestation** and submitted via the EIP-1973 compliant mechanism to prepare for reward distribution.
4. **Reward Distribution:** The protocol's **reward pool** releases funds to a reward-claiming contract, proportionally allocated according to the gradient-weighted scores. Participants can then claim their earned rewards for that epoch based on their contribution ranking and the total rewards available.
5. **Parameter Updates:** If needed, certain parameters are adjusted between epochs. For example, new time-weighted average price (TWAP) calculations might be applied for fee purposes, or other configurations regarding weight or parameters for the next epoch. These updates ensure the system stays balanced and up-to-date.
6. **Reset and New Epoch:** The contribution scores are then reset for the new epoch, and the next epoch begins, repeating the process.

This regular economic rhythm ensures that incentives are predictable and cyclic, while still allowing participants to adapt their strategies over time. By tying rewards to contribution in each epoch, the network encourages consistent, value-adding participation and can adjust to changing conditions in a structured way.

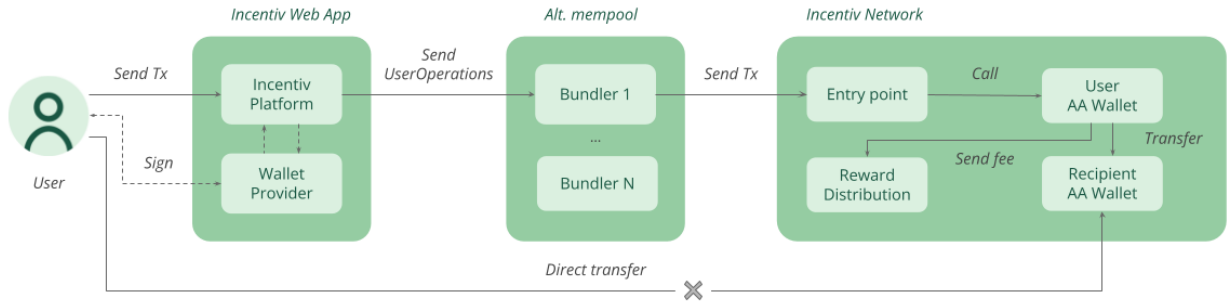
Incentiv's architecture is designed as a modular, EVM-compatible Layer 1 with Advanced Account Abstraction (AAA) and a Proof-of-Work (PoW) consensus framework. The architecture follows a clean separation-of-concerns principle, dividing core blockchain operations, economic scoring, and fee routing into discrete, upgradeable components.

4.3. Architectural Diagrams

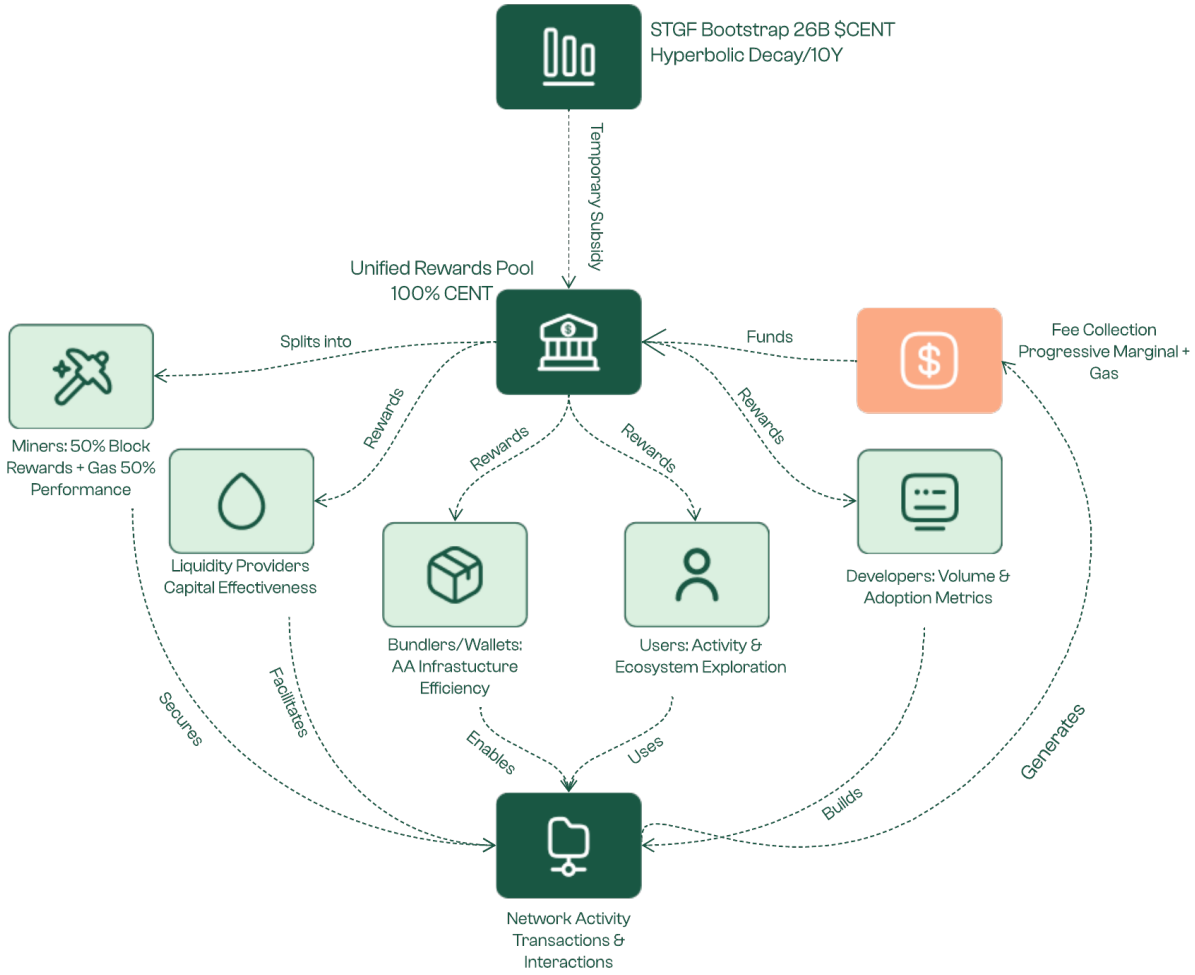
4.4.1 Component Interaction Diagram



4.4.2 Transaction Flow Diagram



4.4.3 Economic Signal Flow



Technical Specifications

4.5.1 Key Parameters

- **Block Time**: ~13 seconds (variable based on difficulty adjustment).
- **Gas Limit**: 30-35 million gas per block.
- **Ethash DAG Size**: Growing at ~8 GB per 3 years.
- **Epoch Length**: ~40,320 blocks (7 days, configurable).
- **Value Fee**: Progressive fee collection model, see details in section 8.1.2 (configurable).

4.5.2 Technology Stack

- **Base Layer**: Custom Ethereum-derived client with enhanced consensus.
- **Smart Contracts**: Solidity with EVM compatibility.
- **RPC Interface**: Ethereum-compatible JSON-RPC with AAA extensions.
- **Client Library**: Web3.js and Ethers.js compatible SDK with AAA support.

4.5.3 Network Requirements

- **Storage**: ~500 GB for full node (projected year 1).
- **Bandwidth**: 25 Mbps recommended for miners.
- **Hardware**: 16 GB RAM, 4+ cores for node operation (excluding mining).

4.4. Developer Experience

In addition to robust core protocols and user-facing features, an excellent **Developer Experience** is crucial for driving adoption. By providing powerful tools and clear extension points, the architecture ensures that developers can easily build on, extend, and troubleshoot the system.

4.4.1 SDK and API Support

To encourage developer adoption, the protocol offers comprehensive tooling:

- **High-Level SDKs:** The platform provides high-level Software Development Kits that abstract away protocol complexity by exposing simple APIs for common operations. This allows developers to focus on application logic rather than dealing with low-level infrastructure details.
- **Wallet Integration Kits:** There are pre-built components to handle tasks such as user wallet connection, transaction (operation) construction, and status tracking. By using these integration kits, applications can achieve full wallet functionality with minimal custom integration effort.
- **Simulation Tools:** Local simulation environments (e.g. sandbox networks or local testnets) enable testing and debugging without deploying to a live network. Developers can verify contract behavior and transaction flows in isolation, catching issues early before any mainnet interaction.
- **Debugging Infrastructure:** The architecture includes robust debugging support with comprehensive logging, transaction tracing, and error reporting. These tools make it easier to identify and resolve issues during development and provide valuable insights for troubleshooting in production.

4.4.2 Extensibility Points

The protocol is designed with well-defined extension points that encourage innovation and customization:

- **Custom Wallet Modules:** Developers can create reusable wallet modules to add new features or logic to existing smart wallets. This modular approach lets innovation accumulate within the ecosystem (via shared enhancements) rather than spawning fragmented, incompatible wallet implementations.

- **Paymaster Services:** The architecture allows for custom paymaster implementations, unlocking novel transaction fee sponsorship models. For example, developers might introduce paymasters for loyalty reward programs, cross-chain fee subsidies, or conditional fee waivers—broadening how transaction costs can be handled on behalf of users.
- **Bundler Strategies:** Bundler implementations (which package and submit user operations) can explore new optimization strategies. For instance, different bundlers might use advanced batching algorithms or priority rules to improve efficiency. Such competitive improvements among bundlers ultimately benefit users through faster, cheaper, and more reliable transaction processing.
- **Application Integration:** Deep integration hooks enable applications across domains, from gaming and DeFi to social platforms, to seamlessly incorporate the protocol's capabilities. This means developers can embed the protocol's features natively into their apps, providing a smooth user experience without needing complex workarounds.

4.5 Protocol Modularity and Extensibility

Incentiv's protocol can adapt and grow over time without compromising its core integrity. Two categories of mechanisms facilitate this: **upgrade paths** for evolving existing components, and **extension points** for adding new capabilities. Together, these ensure the protocol can be improved via governance and innovation while maintaining deterministic execution and transparent operation.

4.5.1 Upgrade Paths

- **Parameter Adjustments:** Network parameters (economic weights, thresholds, rates, etc.) can be fine-tuned through on-chain governance votes. This allows the community to adjust economic incentives and protocol constants as needed without hard-forking the entire system. By tweaking parameters via governance, Incentiv can respond to changing conditions (e.g. market volatility or usage patterns) in a controlled manner.
- **Smart Contract Updates:** The network's smart contracts are modular and follow upgradeable design patterns. This means core logic can be improved or patched through secure upgrade mechanisms (such as proxy contracts or scheduled replacements) without disrupting the network. Modular contract design allows deploying new versions of contracts to introduce feature enhancements or bug fixes, ensuring the protocol's logic can evolve over time.
- **Protocol Enhancements:** Deeper changes to consensus or core logic are possible via coordinated network upgrades (similar to hard forks but with community coordination). When a significant improvement is needed (e.g. a new consensus algorithm or major feature), the architecture supports deploying it in a controlled upgrade process. This path ensures that even fundamental protocol layers can be enhanced when stakeholders agree, keeping Incentiv's core up-to-date with technological advances.

4.5.2 Extension Points

- **Custom Wallet Logic:** The protocol supports plugging in custom wallet smart contracts, allowing advanced security or recovery mechanisms. For example, users or organizations could use wallets with multi-signature approvals, social recovery (trusted guardians for key recovery), or time-locked withdrawal logic. These custom wallet extensions enhance security and usability without modifying the core protocol.

- **Specialized Bundlers:** Incentiv's design allows for domain-specific transaction bundlers or operation processors. Bundlers are entities that can collect and process user operations (similar to how Ethereum's ERC-4337 uses "Bundlers" for account abstraction transactions). Specialized bundlers could, for instance, optimize for certain application types or implement custom mempool logic for particular use-cases, extending how transactions are handled while remaining compatible with the base protocol.
- **Score Plugins:** The architecture can accommodate additional **plugins** that introduce new metrics for evaluating roles or contributions in the ecosystem. For example, beyond basic metrics like stake or uptime, a plugin might score miners or validators on factors like responsiveness, reliability, or community service. These score plugins feed into Incentiv's transparent scoring system so that contributors are measured on a richer set of criteria, all without altering core consensus rules.
- **Layer 2 Solutions:** The protocol is designed to be **Layer-2 friendly**, meaning scalability extensions can integrate seamlessly. Incentiv can support rollups or sidechains that anchor to it, and those Layer 2 solutions can benefit from the base protocol's features (like its account abstraction model) while providing higher throughput. This extensibility ensures that as demand grows, additional throughput and functionality can be added on top of the base layer without compromising security.

These extension points enable ecosystem-driven innovation on top of Incentiv. Developers and community members can introduce new capabilities through wallets, plugins, or Layer-2 implementations, all while the core protocol remains secure and stable. The modular design means the **integrity of the core logic is preserved** – changes are opt-in and additive, preventing ad-hoc modifications that could introduce vulnerabilities.

4.6 Architecture Summary

Incentiv's architecture tightly interweaves technical components with economic incentives to create a balanced, self-sustaining system. Every participant is **algorithmically measured and fairly compensated** for their contributions. Key architectural characteristics include deterministic execution of smart contracts, a transparent scoring mechanism for contributions, and an inclusive design that allows a wide range of participants to take part in consensus and operations.

Each element of the system thus serves a dual purpose: it fulfills a functional requirement and aligns incentives. For example, the scoring system not only tracks performance for transparency, but also directly drives reward distribution so that participants are rewarded in proportion to their positive contributions. This alignment ensures that participants are incentivized to uphold the network's health and security.

Overall, Incentiv's architecture ensures that the integrity of the design logic is maintained across the entire ecosystem, even as individual parts evolve. The use of modular upgrade paths and extension points means each part of the architecture can improve or adapt **without undermining the whole**, allowing the protocol to **evolve in a forward-compatible way**.

5. Advanced Account Abstraction (AAA)

One of the most significant aspects of Incentiv's design is its embrace of Advanced Account Abstraction (AAA) as a core feature. Account abstraction is the result of a years-long evolution in blockchain account models. To appreciate Incentiv's approach, it helps to understand how account abstraction developed in the Ethereum ecosystem and why making it native to the protocol is so impactful.

Evolution of Account Abstraction in Ethereum

Since Ethereum's inception, the account model has been split into two types: Externally Owned Accounts (EOAs), controlled by private keys, and Contract Accounts, which run code. This dual system introduced artificial constraints. Notably, only EOAs can initiate transactions on Ethereum – smart contracts cannot start a transaction on their own. An EOA must always be the originator, signing the transaction with its private key. Moreover, Ethereum historically limited all accounts to a single built-in signature scheme (ECDSA, the Elliptic Curve Digital Signature Algorithm). In practice, this means wallet software has had no flexibility to use other cryptographic schemes. There is also no native mechanism for account recovery or social recovery in the base protocol – if a user loses their private key, the only recourse is restoring from a backup. These limitations in control, cryptography, and recoverability have long been seen as obstacles to a more user-friendly and secure blockchain experience.

Early Ethereum developers recognized these issues and began proposing "account abstraction" solutions to overcome them. As far back as 2016, Vitalik Buterin introduced EIP-86, aimed at abstracting transaction origin and signature verification. The idea was to allow smart contracts to verify signatures and initiate transactions, effectively treating contract accounts more like first-class citizens. Over the years, several iterations followed. In 2020, EIP-2938 was proposed to add a new transaction type (sometimes called an "AA transaction") that would let smart contracts act as top-level accounts. Another approach, EIP-3074 (circa 2020–2021), suggested new opcodes (AUTH and AUTHCALL) to let an EOA delegate control to a smart contract which could then perform actions on the user's behalf. Each of these proposals attempted to bring greater flexibility to account management and address the one-size-fits-all limitations of EOAs. However, they all faced an adoption hurdle: most required changes to Ethereum's core protocol. Changes at the base layer are slow and require broad coordination, and indeed none of those early EIPs were implemented on Ethereum's mainnet.

The concept of account abstraction gained significant traction with ERC-4337[8], introduced as an Ethereum improvement in late 2021 and eventually deployed on Ethereum mainnet in early 2023. Unlike prior proposals, ERC-4337 achieved account abstraction without any consensus-layer changes. It works as a higher-layer mechanism: users send their operations to a special **UserOperation** mempool, and **Bundler** nodes package these user operations into normal Ethereum transactions that call a global **EntryPoint** smart contract. In essence, ERC-4337 created a decentralized framework on top of Ethereum for smart contract wallets (sometimes called "smart accounts") to behave more like EOAs. This enabled features like multiple signatures, session keys, third-party gas payments (via paymaster contracts), and social recovery – all through smart contract logic rather than new protocol rules. The deployment of ERC-4337 was a major milestone, proving that account abstraction features could be offered on Ethereum as an opt-in service at the application layer. However, because it operates on top of the base protocol, it still treats account abstraction as an add-on rather than a native feature. For instance, even with ERC-4337, an EOA (the Bundler's key) is still ultimately required to get user operations included in a block. In other words, Ethereum's mainnet account abstraction is **non-native** – it does not fundamentally change the rule that only EOAs can initiate a transaction. It works around this by introducing an intermediary layer where contract wallets *simulate* initiating transactions, but a traditional externally owned account (the bundler) actually triggers execution on-chain.

Other blockchains and layer-2 networks have since explored making account abstraction more native. Several Ethereum Layer-2 solutions (e.g. StarkNet, zkSync, Aztec) enshrined account abstraction at the protocol level, where all accounts are smart contracts and transactions can be initiated directly by these contracts. This illustrates a broader point: until recently, most blockchains treated account abstraction as either an optional feature, a second-layer add-on, or a nice-to-have enhancement rather than a core design principle.

Native Account Abstraction in Incentiv's Architecture

In contrast to previous approaches, Incentiv integrates account abstraction as a fundamental architectural layer from the outset. Smart contract wallets (contract-based accounts) are *first-class citizens* in the Incentiv network, meaning they can initiate transactions and enforce arbitrary verification logic natively. At the same time, legacy support for EOAs is maintained for compatibility – users who prefer traditional key-controlled accounts can still use them. The paradigm shift is that the default and encouraged mode in Incentiv is to use smart-contract-based accounts with built-in flexibility. This unlocks enhanced security and usability features across the entire ecosystem by design, not as an afterthought.

By enshrining account abstraction in the base protocol, Incentiv can offer capabilities that currently, other networks such as Ethereum achieve through extra layers. For example, multi-signature approvals, social recovery mechanisms, custom cryptographic schemes, and delegated fee payments can all be handled at the protocol level in Incentiv. A transaction on Incentiv can be directly originated by a user's smart wallet contract (e.g. requiring M-of-N signatures or other custom logic), without any externally owned key having to physically initiate it. This native approach contrasts with Ethereum's ERC-4337, where such logic lives in higher-layer infrastructure and requires an EOA-driven bundler to ultimately trigger the execution.

The benefits of Incentiv's integrated account abstraction are significant. **Security** is improved as users are not constrained to a single private key – they can distribute risk across multiple keys, devices, or biometric factors, and use social or institutional guardians for recovery. **Usability** is enhanced through features like built-in account recovery (users can regain access via trusted parties or other devices if a key is lost), flexible key management (for example, rotating keys or using secure hardware enclaves for signing), and gas abstraction (allowing transaction fees to be paid in different tokens or by sponsors, removing a common pain point for new users). In short, every account on Incentiv can be as smart and programmable as a deployed contract, bringing the full power of smart contracts to user wallets natively.

By prioritizing smart contract wallets while still accommodating EOAs, Incentiv ensures that no user is left behind during this transition. Those who wish to continue using simple key-based accounts may do so, but the system as a whole gains the powerful capabilities of account abstraction at its core.

This approach aligns with emerging industry trends. For example, several Ethereum rollups and L2 networks now tout native account abstraction to improve user experience. Incentiv takes this philosophy to the layer-1 protocol itself. The result is an ecosystem where every participant's account is flexible and secure by default, and advanced features that previously required complicated workarounds are available out-of-the-box. This represents a new standard for blockchain account management, reflecting lessons learned from nearly a decade of incremental improvements in the Ethereum community and positioning Incentiv at the forefront of user-centric blockchain design.

5.1 The Advanced Account Abstraction Model

5.1.1 Fundamental Principles

Account abstraction represents blockchain's most significant usability evolution since the advent of smart contracts. While Ethereum's ERC-4337 demonstrated the concept's viability as an overlay, Incentiv implements account abstraction as native infrastructure – transforming it from an optional enhancement into a fundamental part of the architecture.

This architectural decision stems from a simple recognition, according to which if blockchain systems continue to demand that users understand private keys, manage gas fees, and manually construct transactions, these systems will remain tools for technical specialists rather than truly global infrastructure. Advanced Account Abstraction flips this

paradigm, creating a system that adapts to human needs rather than forcing humans to adapt to the system's technical requirements.

Incentiv realizes this vision through three key principles:

- **Intent-Oriented Interaction:** Users initiate high-level *actions* such as "send 100 USDC to Alice," "swap my tokens," or "update my account permissions," without worrying about the low-level transaction mechanics. The wallet and network interpret and execute these goals, abstracting away the minutiae like function calls, gas limits, and nonce management. This makes blockchain interactions more intuitive and task-focused, enhancing accessibility for everyday users.
- **Flexible Access and Recovery Models:** Since every user's primary account is a smart contract, wallets can support customizable authentication and recovery methods. Users are free to choose security setups that match their needs and risk tolerance. This could range from simple PINs or biometric authentication, to multi-signature approvals (e.g. requiring consent from multiple devices or people for large transfers), to programmable policies (like time-locking certain actions or limiting daily spending). Losing one key no longer means irrecoverable loss of funds – social recovery and other backup methods are built-in.
- **Economic Abstraction:** Transaction fees become an implementation detail rather than a direct user concern. The AAA design allows gas to be paid in any token or even by third parties, and lets users bundle multiple operations together for efficiency. A user might never need to hold the network's native coin just to pay fees. The protocol handles the complexity of fee conversion and payment behind the scenes (through components like Paymasters, described later), so that users can focus on the value or service they want – not on how to pay miners or validators.

5.1.2 Structural Components

To turn these principles into reality, Incentiv's Advanced Account Abstraction (AAA) architecture uses several interconnected components. Together, they transform raw blockchain functionality into more accessible services:

- **Smart Contract Wallets:** These serve as the user's primary interface with the network, effectively replacing the traditional EOA while maintaining backward compatibility for those who need it. A smart contract wallet is not merely a keypair and an address; it is a programmable agent that can hold assets, enforce user-defined rules, and automate complex operations. Each user's wallet is a smart contract that can specify its own logic for verifying transactions (e.g. what signatures or conditions are required), allowing features like multi-signature security or spending policies natively at the account level.
- **EntryPoint Contract:** The EntryPoint is a special contract that acts as the universal gateway for executing abstracted transactions (UserOperations) on the chain. It is a single, well-known contract that all smart wallets and bundlers interact with. The EntryPoint is responsible for validating UserOperations and executing them in a deterministic, secure manner. It ensures that security checks (like signatures and nonce management) are passed and that fees are paid, before forwarding calls to the target contracts. By having a single entry point, the protocol can efficiently manage and sequence the execution of many user operations while maintaining consistent rules for all.
- **UserOperation Objects:** A UserOperation is a new type of transaction object that encapsulates a user's intent in a self-contained package. Instead of a traditional transaction that just says "call X contract with Y data and pay Z gas," a UserOperation includes higher-level details: the target action or contract call, any parameters, the account that should execute it (smart wallet), authentication proof(s) (e.g. signatures), and details for fee

payment (which token, which paymaster will cover it, etc.). This standardized format enables wallets, bundlers, and the EntryPoint to coordinate seamlessly while supporting arbitrary complexity in what the user wants to do.

- **Bundlers:** Bundlers form a specialized **network layer** that collects UserOperations from users and packages them into batches for inclusion in blocks. Importantly, bundlers are *not* part of the blockchain consensus itself, but rather off-chain actors (similar to transaction relays) that participate in the network. They operate external to the chain's core protocol, running on their own infrastructure, and their job is to help user operations get on-chain efficiently. Bundlers listen to the user operation mempool, compete to create the most profitable or efficient bundles of operations, and submit those bundles as ordinary transactions (calling the EntryPoint) to the blockchain. This market-driven layer ensures users benefit from competition among service providers – for example, bundlers might offer faster inclusion or lower fees. In Incentiv, bundlers are first-class participants in the ecosystem, but they remain external service providers (they are not miners/validators). This separation of roles allows for a rich marketplace of transaction packaging and related services without changing the core consensus logic of the chain.
- **Paymasters:** Paymasters are unique smart contracts that enable flexible payment of transaction fees. They allow the gas for a UserOperation to be paid by someone other than the user's own account. In practice, a paymaster can sponsor the gas costs (in the native token) for a user's operation under certain conditions, effectively allowing users to pay fees in alternative tokens or even no tokens at all. Paymasters can be offered by dApps, wallet providers, or third parties who want to improve user onboarding (by covering users' fees) or accept fees in a token of the user's choice. This component removes one of the long-standing barriers in blockchain use: the requirement that every user hold the chain's native coin just to do anything. Paymasters are secured by the protocol (they have to put down a stake and can be penalized if they misbehave), ensuring that fee sponsorship does not compromise the network's security or spam resistance.

5.2 EntryPoint and Transaction Flow

5.2.1 Intent-Based Transaction Model

Traditional blockchain transactions force users to specify *exact* execution details – you have to choose the function call, encode parameters, set a gas limit, manage nonces, and pay in the correct fee token. This level of manual control offers determinism, but it also creates a steep complexity barrier that excludes many non-technical users.

In contrast, Incentiv's AAA model allows users to initiate transactions in natural, intent-oriented terms. Instead of thinking "I need to call function X on contract Y with Z gas," users think in terms of desired outcomes, and the system figures out the rest. For example, consider a scenario in which a freelancer wants to automate their finances: "Every Friday, receive my payment from Client A in USDC, automatically convert 20% of it to ETH, and move the remaining 80% into my savings account."

On a traditional chain, accomplishing this would require the user (or client) to perform multiple transactions and manual steps on an ongoing schedule: one transaction each week to send USDC, another to swap tokens via a DEX, and another to transfer funds to a savings wallet – not to mention manual calculation of 20% of the total USDC and gas for each step. With Incentiv's account abstraction, the freelancer can set this up as a single high-level instruction to their smart contract wallet. The wallet's internal logic orchestrates everything: scheduling the weekly trigger, pulling the USDC from the client (or accepting it when sent), executing the token swap through a DEX at the best rate, and transferring the remainder to the savings contract. All of the low-level operations (splitting amounts, interacting with a DEX, handling gas fees and transaction ordering) are abstracted away from the user. They experience it as one continuous "automation" rather than many disparate actions.

This kind of abstraction isn't limited to finance. Whether it's setting up recurring DAO contributions, executing a staged token vesting plan, or managing access controls for a deployed contract, users can interact with the blockchain through familiar *intentions*. The smart contract wallets interpret those intentions and handle the underlying blockchain logic. This means users can unlock powerful capabilities without needing deep technical expertise for each action. The blockchain becomes a platform that can understand and execute user-level goals directly, greatly simplifying the experience.

5.2.2 Abstracted Transaction Flow

Behind the scenes, a lot happens to turn a user's high-level intent into actual state changes on the blockchain. The entire transaction lifecycle in Incentiv's AAA can be broken down into a sequence of stages (see the diagram above). This flow involves multiple actors and checks to ensure everything is secure and efficient. The process unfolds as follows:

1. **Intent Submission:** The user interacts with a familiar interface (a mobile wallet app, web dApp, or other service) and expresses a desired outcome. For instance, they tap "Send assets" or approve an automated rule. At this point, the user is dealing with simple terms (amounts, recipients, actions) rather than low-level transaction data.
2. **Wallet Processing:** The user's wallet client (which could be the wallet app or browser extension connected to their smart contract wallet) takes the user's intent and formulates a **UserOperation** object. This includes encoding the necessary contract calls to achieve the intent, applying the user's security policies (e.g. adding required signatures or approvals), and preparing details on how fees will be paid (which token, which paymaster, etc.). Essentially, the smart wallet translates the high-level request into a self-contained operation that it can later execute on-chain.
3. **Local Validation:** Before broadcasting anything, the wallet performs a local simulation or validation of the UserOperation. It ensures that the operation meets all user-defined rules and is likely to succeed. This might include checking that the user has sufficient funds (or an appropriate paymaster is attached), that required signatures are present, and that any custom logic (like spending limits or whitelisted recipients) is satisfied. By catching errors or rule violations early, the wallet prevents the user from wasting fees on transactions that would ultimately fail.
4. **Bundler Selection:** Once the UserOperation is ready and locally validated, the wallet submits it to the network by sending it to a Bundler. There may be multiple bundler nodes available (run by different providers), so wallets can choose where to send the operation. This creates a marketplace: bundlers might advertise different fee rates or quality of service. The wallet could pick a bundler based on price, speed, reputation, or a pre-existing arrangement (for example, the wallet developer might partner with a particular bundler). At this stage, the user's operation enters the **alternative mempool** maintained by the chosen bundler, rather than the normal transaction pool used by validators/miners.
5. **Bundle Assembly:** The bundler gathers the user's operation along with others in its mempool and starts forming a bundle. The bundler's goal is to package multiple UserOperations into one efficient batch (which will be wrapped into a single transaction to the EntryPoint contract). It will optimize this bundle for gas efficiency (sharing overhead costs among operations) and may reorder operations for optimal execution, as long as any specified dependencies or atomicity requirements are respected. Advanced bundlers might also offer **value-added services** at this stage, such as protecting the bundle from MEV (Maximal Extractable Value) exploitation or ensuring certain privacy guarantees for the included operations.
6. **EntryPoint Execution:** The bundler submits the bundled transaction to an Incentiv block proposer (validator). When this transaction is included in a block, the **EntryPoint contract** is invoked. The EntryPoint then iterates

through each UserOperation in the bundle and executes them one by one. For each operation, it first validates the operation's signatures and paymaster conditions (preventing invalid or unauthenticated ops from proceeding). Then it executes the operation's specified actions (calling the user's smart wallet and whatever contract interactions were requested). Critically, the EntryPoint also handles fee payment: it deducts the appropriate gas cost, either from the user's account or from the designated paymaster's balance, and transfers it to the block producer. All of this happens within one atomic transaction. If any operation in the bundle fails its validation or execution, the EntryPoint will abort that operation and move on (or potentially revert the entire bundle if atomic execution was required), ensuring that one user's error doesn't break others.

7. **Result Propagation:** After execution, the state changes from each successful UserOperation are now part of the blockchain's state. The effects (token transfers, contract calls, etc.) have been applied. The outcome of each operation is then reported back to the user. The wallet or dApp interface will inform the user that the action is completed (or if it failed, provide an error). Users might receive notifications via their wallet app (for instance, "Your scheduled payment was sent and converted to ETH successfully"). From the user's perspective, they simply see that their requested action has been fulfilled, without needing to know the intricate dance of bundlers and EntryPoint that made it happen.

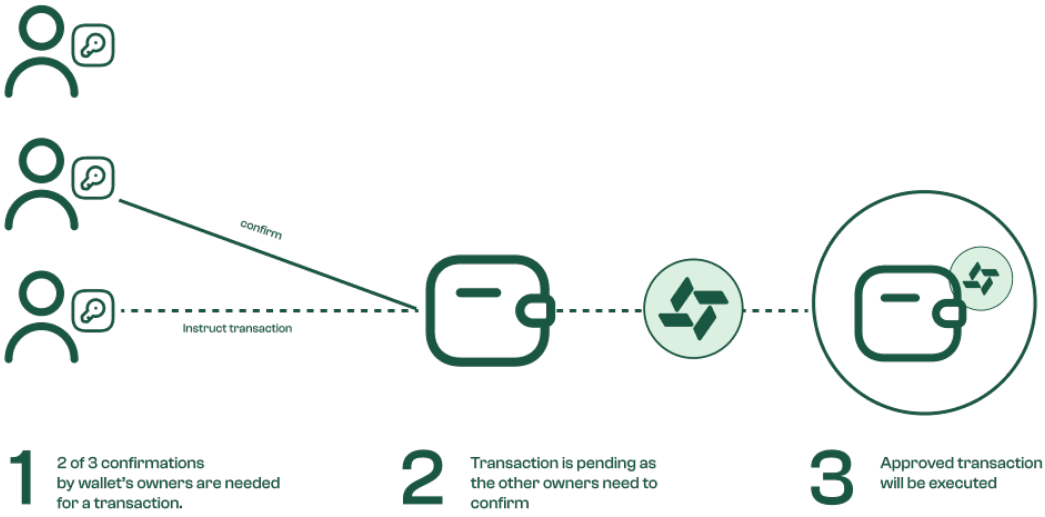
This entire flow is designed to be seamless and secure. Each layer – from the wallet, to the bundler, to the EntryPoint – adds value (usability, efficiency, security checks) without burdening the user with complexity. Importantly, while there are new actors like bundlers and paymasters in the mix, the protocol ensures they behave correctly (through economic incentives and verification logic) so that users do not need to trust any single intermediary blindly.

5.3 Smart Contract Wallets

5.3.1 Core Features

Smart contract wallets turn blockchain accounts from static keypairs into dynamic, programmable agents. By doing so, they enable features that are impossible with traditional accounts tied to a single private key. Some core features and benefits of Incentiv's smart wallets include:

- **Multi-Signature Security:** Users can define complex authorization rules for their wallet. For example, a wallet can require multiple signatures from different keys (or even different people) to approve a high-value transfer, greatly reducing the risk of theft from a single compromised key. The wallet could also enforce time-locks on certain operations (e.g. require 24 hours before a large withdrawal is finalized, giving time to react to a breach) or other conditions. This level of security policy is customizable by the user and enforced by the wallet contract itself.



- **Social/Wallet Recovery:** Losing access to a traditional crypto wallet (EOA) can be catastrophic. In a smart contract wallet, users can build their own recovery mechanisms. For instance, a wallet might designate a set of "guardian" addresses (friends, family, or even institutional services) that can, with some threshold of agreement, help restore access if the owner loses their keys. This eliminates the single point of failure of a seed phrase, making the overall system more resilient and user-friendly. You no longer have to choose between security and convenience – you can have both.
- **Spending Controls:** Because the wallet's logic is programmable, users can impose spending limits or require additional checks for unusual activity. For example, a user could set a daily token transfer limit, whitelist certain "trusted" addresses that don't require approvals for payments, or automatically distribute incoming funds according to a budget. This turns the wallet into a personal finance management tool, not just a passive container of assets.
- **Transaction Policy Enforcement:** Incentiv smart wallets integrate TransferGate—a programmable rule engine that allows users to approve or reject transactions based on custom-defined criteria. Transactions can be conditionally blocked, flagged, or delayed if they involve high-risk addresses, exceed predefined thresholds, or originate from unauthorized sources. This enables real-time enforcement of user-specified policies, turning each wallet into an intelligent security layer capable of actively protecting itself before any state changes occur.
- **Session Keys:** Smart wallets can create temporary, limited-permission keys for specific applications or time periods. For instance, a user playing a blockchain game might grant the game a session key that allows it to make small in-game transactions from the user's wallet for the next hour, but that key cannot move funds elsewhere or is limited to a small amount. Similarly, a DeFi application could be given a time-bound permission to rebalance a user's portfolio. This allows seamless interaction with dApps without constantly prompting the user to approve transactions or exposing the main keys – improving both security and user experience.
- **Upgradeable Logic:** Over time, security best practices and user needs evolve. Incentiv's smart wallet contracts can be designed to support upgrades (when opted-in by the user). Using established patterns like proxy contracts, a user can migrate their wallet's logic to a new, improved version without changing the wallet's address or moving funds. This means wallets can continuously improve – adding new features, patching vulnerabilities, or adapting to new protocol changes – all while the user's public identity (their address) remains

the same. Of course, upgradeability is handled carefully and usually requires owner consent (or even multi-sig approval) to prevent any unauthorized changes.

5.3.2 Implementation Standards

While innovation in wallet design is encouraged, the Incentiv protocol also provides standards and frameworks to ensure interoperability and security across all smart contract wallets on the network. Key implementation standards include:

- **Modular Architecture:** Smart wallets on Incentiv are built with modularity in mind. There is a clear separation between the core wallet logic (the essential functions every wallet must support, like verifying a signature or executing a transaction) and optional modules or extensions that add extra features. This means users can customize their wallet by adding or removing modules (for things like social recovery, specific DeFi integrations, etc.) without needing to change the core wallet or deploy a whole new contract. It fosters an ecosystem of plug-in enhancements for wallets.
- **Standardized Interfaces:** To avoid fragmentation, common wallet functionalities use standard interfaces and conventions. For example, there might be a standard function signature for "executeUserOperation" or for adding a guardian or session key. Applications and bundlers can rely on these standard interfaces to interact with any compliant wallet. This ensures that even though individual wallets can have different features, they all speak a common "language" as far as outside apps are concerned, making integration with dApps and services smooth.
- **Security Best Practices:** The protocol comes with reference implementations and audited examples of smart wallets that embody best practices. Common patterns (like multi-sig, social recovery, etc.) are provided in libraries that developers can use rather than writing from scratch, reducing the likelihood of bugs. Additionally, the network can impose certain baseline requirements (for instance, a wallet must implement a particular validation function for UserOperations) to ensure no wallet design inadvertently undermines security. Incentiv's approach is to balance flexibility with safety by guiding developers towards proven patterns.
- **Minimal Proxy Deployment:** To make deployment of smart wallets efficient, new wallet instances can be created using the minimal proxy pattern (also known as "clone contracts"). In essence, rather than deploying a full copy of the wallet contract code for each user, the network uses a single implementation contract for the wallet logic, and each user's wallet is a thin proxy that delegates calls to this implementation. This dramatically reduces the cost (in gas and storage) of deploying a new wallet and ensures that all users are running a well-audited codebase. Each proxy holds the user's unique data (like their keys or settings) but shares the common code. If the wallet logic is upgraded (to a new implementation contract), users can choose to point their proxy to the new version, allowing a smooth upgrade path with minimal cost.

5.4 Bundlers and the Alternative Mempool

5.4.1 Bundler Implementation

Incentiv introduces **bundlers** as a specialized class of nodes that exist alongside regular validators. These bundlers play a crucial role in the account abstraction framework by operating an alternative mempool for UserOperations and optimizing how those operations are included in blocks. After evaluating various bundler implementations (from the Ethereum ecosystem and beyond), Incentiv has identified several key performance and capability metrics to guide the bundler ecosystem: implementation language, stability, throughput performance, multi-chain support, feature set, and maintenance maturity. Below is a comparison of some known bundler implementations (many originating from Ethereum's ERC-4337 ecosystem):

Bundler Implementation	Language	Stability	Performance	Multi-Chain Support	Feature Set	Maintenance
Infinitism **(reference)**	TypeScript	0.4	0.4	0.4	0.6	0.6
Stackup	Go	0.8	0.8	0.8	0.6	0.8
Rundler **(Alchemy)**	Rust	0.9	0.9	0.8	0.8	0.9
Biconomy	TypeScript	0.6	0.6	0.8	0.8	0.6
Skandha **(Etherspot)**	TypeScript	0.6	0.6	0.9	0.6	0.6
Alto **(Pimlico)**	TypeScript	0.6	0.6	0.8	0.9	0.8

Note: Scores range from 0.0 (insufficient) to 1.0 (exceptional) based on internal benchmarking across multiple operational parameters.

From these, **Rundler** (a Rust-based bundler by Alchemy) emerges as a reference implementation due to several technical advantages it demonstrated in testing:

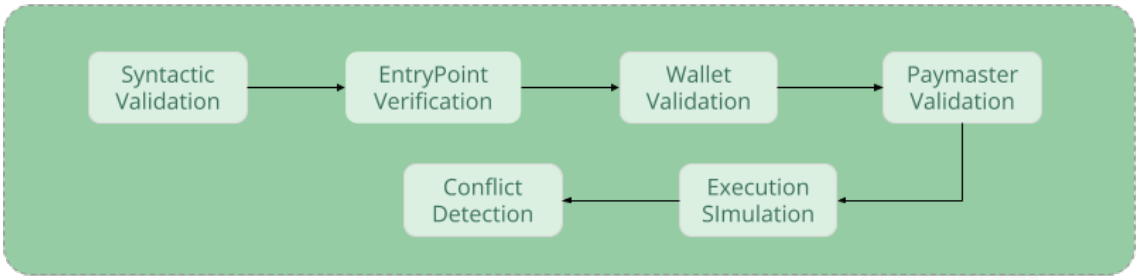
- **Performance:** It achieved the highest throughput (over 10,000 UserOperations per minute in benchmarks) with minimal latency. This means it can bundle and process a large number of user intents very quickly, which is crucial as the network scales.
- **Stability:** Written in a memory-safe language (Rust) with careful error handling, Rundler showed excellent stability under load. Crashes or hangs were virtually nonexistent in tests, and it cleanly handled edge cases, which is vital for a service expected to run continuously.

- **Extensibility:** Rundler has a component-oriented architecture, making it easier to customize for network-specific needs. For example, if Incentiv introduced a unique ordering rule or a different pricing mechanism for operations, Rundler's modular design would allow implementing that without rewriting the whole system.
- **Future-Readiness:** This implementation is built with upcoming improvements in mind – including better peer-to-peer distribution of UserOperations (so bundlers can share operations among themselves) and advanced mempool management. It's poised to adopt future protocol upgrades with minimal fuss.

The technical work of a bundler goes beyond simply relaying transactions from point A to point B. Key aspects of a bundler's functionality include:

- **UserOperation Mempool Architecture:** Unlike a traditional Ethereum mempool (which just pools individual transactions sorted by gas price), a bundler's mempool maintains more sophisticated data structures. It may organize UserOps by multiple factors such as offered fee, operation type or complexity, and even the reputation of the issuing smart wallet or paymaster. Bundlers track dependencies between operations (for example, if one operation's effect might enable another) to decide an optimal execution order. They might also implement fairness mechanisms (ensuring that a large flood of ops from one user doesn't starve others) and dynamic pricing to manage congestion. This tailored mempool is key to bundlers efficiently handling the rich transaction flow that AAA enables.
- **Simulation and Validation Pipeline:** Before a bundler includes a UserOperation in a bundle, it runs a multi-stage verification process locally. This includes checking that the UserOp is formatted correctly and not malformed (syntactic validation), verifying that it targets a valid EntryPoint contract and that the user's smart wallet contract actually exists on-chain (to prevent wastage on operations bound to fail), and running the smart wallet's verification logic (e.g. signature checks) in a sandbox. If a paymaster is sponsoring the gas, the bundler checks that the paymaster is properly staked and has enough balance to cover the cost. It even simulates executing the UserOperation (dry-run) to estimate gas usage and ensure it will succeed given the current state. If any of these steps fail, the bundler will reject the UserOperation before it ever hits the blockchain, protecting users and the network from invalid operations and potential attacks.

Bundler Simulation Pipeline



- **Bundle Construction:** After validation, the bundler assembles a set of compatible UserOperations into a single **bundle transaction**. The bundler must decide which ops to include and in what order. This can involve complex algorithms to maximize efficiency – for instance, grouping operations that call the same contracts (to save overhead), or ordering operations to avoid conflicts (two ops trying to spend the same funds might need to be separated or handled carefully). The bundler also ensures that if certain operations must be executed atomically together (all-or-nothing), they end up in the same bundle. Ultimately, the bundler produces one transaction that calls `EntryPoint.handleOps(UserOps[])` with the array of operations. A well-

constructed bundle will minimize redundant gas costs and take advantage of any synergies between the operations included.

Throughout these processes, bundlers act as **optimizers and service providers**. They offload work from the consensus layer (by pre-validating and packaging transactions) and create a competitive market for users' business. It's important to note that anyone can run a bundler on Incentiv – there's no special permission needed. This open model encourages multiple implementations and continuous improvement, as bundler operators vie to offer the fastest, cheapest, or most reliable service.

5.4.2 Bundler Behavior and Economics

The introduction of bundlers opens up new economic dynamics in the network that were not possible in a traditional monolithic transaction model. Because intent generation (by wallets) is separated from transaction inclusion (by bundlers), we can have innovative fee arrangements and services. Some possibilities include:

- **Revenue-Sharing Arrangements:** Wallet providers and bundlers can form partnerships. For example, a popular wallet app might direct all its users' operations to a particular bundler by default. In return, the bundler could share a portion of the revenue (fees or any MEV captured) back with the wallet provider or even the users. This creates an incentive for wallets to integrate with high-quality bundlers and for bundlers to compete for wallet partnerships.
- **Service Tiers:** Bundlers might offer different levels of service at different fee rates. For instance, a *Standard* tier could simply include transactions as cheaply as possible on a best-effort basis. A *Priority* tier might, for a slightly higher fee, guarantee inclusion in the next block or give the operation precedence over others. A *Premium* tier could include extras like MEV protection (ensuring the user doesn't get front-run or sandwiched in DeFi trades), privacy (via private mempool submissions), or detailed execution reports. Users (or wallets on their behalf) can choose the tier that meets their needs, similar to shipping options like standard vs. express delivery.
- **Integration Marketplaces:** We may see the emergence of specialized bundlers that integrate closely with certain applications. For example, a DeFi platform could run its own bundler tailored to its users, with custom logic to optimize those specific transactions or offer them at a discount thanks to protocol subsidies. Bundlers could expose APIs for wallets or dApps to plug in, maybe even white-labeling the service so that a wallet can seamlessly use a "built-in" bundler service behind the scenes. This way, bundling can become part of an application's offering (e.g. a game ensuring all in-game transactions are fast and fee-covered via its own bundler).
- **Cross-Domain Operations:** Some bundlers might operate on multiple chains or layers, allowing them to bundle operations that involve cross-chain interactions. For example, if a user operation on Incentiv triggers an action on an Ethereum L2 (via a bridge or message-passing system), a sophisticated bundler could manage the sequence of actions on both chains, optimizing for the user's end-to-end experience. Similarly, bundlers might coordinate transactions that spread across L1 and L2 (or multiple L2s), handling the complexity of each domain but presenting a unified result to the user.

These economic innovations reshape the traditional "one-size-fits-all" transaction inclusion model into a dynamic, market-driven service layer. Instead of every transaction paying a similar fee for the same undifferentiated service from miners/validators, users now have choices and can benefit from competition. A casual user might opt for the lowest-cost bundler, whereas a DeFi trader might pay a premium for MEV-protected bundling. The outcome is generally improved user experience and potentially lower costs, as bundlers find more efficient ways to package transactions.

Importantly, this is achieved while preserving the open, permissionless nature of the network. Any developer or company can deploy a bundler and try to attract users by offering a better service. Likewise, any wallet or user can choose which bundler to use at any time (and can fall back to a different one if service degrades, ensuring no lock-in). The protocol does not centrally favor any bundler; it simply provides the framework (EntryPoint, mempool rules, etc.) for them to operate. This structural separation of concerns (user intent vs. block inclusion) not only adds flexibility, but also enables new sustainable business models in the ecosystem that align incentives toward better usability.

For wallet developers, partnering with reliable bundlers can become a revenue source and a way to differentiate their product (e.g. "our wallet has a fast-track transaction service built-in"). For bundlers, integrating with popular wallets brings a steady flow of operations and the opportunity to scale. This symbiotic relationship encourages innovation and performance. In short, the bundler ecosystem turns transaction inclusion into an active marketplace – one that can continually evolve and improve the user's journey, all while the base blockchain remains secure and decentralized.

5.5 Paymasters and Gas Abstraction

5.5.1 Gas Payment Options

Paymasters dramatically improve usability by removing the long-standing requirement that users must hold a specific network token to pay for gas. In Incentiv, paymasters enable **gas fee abstraction**, meaning users and applications gain much more flexibility in how transaction fees are handled. Several key capabilities include:

- **Token Abstraction:** Users can pay transaction fees with *any* supported token, not just the native coin ($CENT). For example, if a user only holds USDC stablecoins, a paymaster could cover the gas cost in $CENT on the user's behalf and later be reimbursed in USDC. Behind the scenes, the paymaster will convert that USDC (via a DEX or other liquidity source) into the needed $CENT. This happens seamlessly at the protocol level, with conversion rates determined by real-time market prices. From the user's perspective, they spent a little USDC to get their transaction through – they never needed to acquire $CENT at all.
- **Sponsored Transactions:** Applications and services can sponsor gas fees for their users. This opens the door to familiar Web2-like models such as free trials or freemium services. For instance, a new decentralized game might cover the first 100 transactions for each new player (via a paymaster) to let them get started without any upfront cost. Or a DeFi platform might run a promotion where certain operations (like swapping through their exchange) are gas-free for a period. This encourages onboarding and usage by removing the initial friction of paying gas.
- **Subscription Models:** Paymasters can enable subscription-based payment of fees. Imagine a scenario where a user pays a fixed monthly fee (in any token) and in return a service paymaster covers all their transaction costs on the network for that month. Different tiers could offer different levels of usage (similar to phone data plans or streaming service tiers). This makes transaction costs predictable for users and allows service providers to bundle blockchain usage into subscription products.
- **Conditional Sponsorship:** Because paymasters are smart contracts, they can implement arbitrary logic to decide when to pay for a user's operation. This means fee sponsorship can be tied to certain conditions or behaviors. For example, a DeFi protocol's paymaster might sponsor the gas for users who are doing a beneficial action (like providing liquidity or executing a trade above a certain size), effectively offering a rebate. Another example is a paymaster that only pays for new users' first few transactions (to help with onboarding) or one that requires the user to hold a certain NFT to qualify for free transactions. The possibilities are wide open – essentially, paymasters let developers design how and when they want to subsidize or accept alternative payment for gas fees.

5.5.2 Paymaster Mechanics

From a technical standpoint, paymasters must be designed carefully to be useful yet safe. The Incentiv protocol defines how paymasters interact with the EntryPoint and what rules they must follow:

- **Validation Hooks:** A paymaster is a smart contract that implements a specific function (e.g. `validatePaymasterUserOp`) which the EntryPoint will call during transaction verification. This function checks the UserOperation and decides whether the paymaster agrees to pay for it. For instance, the paymaster might verify that the user included a valid coupon code or has enough of a certain token to reimburse later. If the checks pass, the paymaster effectively "sponsors" the operation, and the EntryPoint knows it can charge the paymaster for gas. This hook allows arbitrary logic (any conditions the paymaster wants) but must execute quickly and predictably to not stall the network.
- **Stake and Deposit Requirements:** To participate, a paymaster must lock up a certain amount of funds (stake) in the EntryPoint contract and maintain a deposit of $CENT for paying fees. This is similar to how Ethereum's ERC-4337 paymasters work. The stake acts as a security bond – if the paymaster misbehaves (e.g. its validation hook violates protocol rules or it runs out of funds after promising to pay), it can be penalized or slashed. This economic incentive ensures paymasters only approve operations they genuinely intend to cover and have the funds to do so.
- **Accurate Gas Estimation:** When sponsoring transactions, paymasters need to accurately estimate how much gas an operation will consume to ensure they have sufficient balance to cover it. The Incentiv protocol provides tools for paymasters to get gas estimates (for example, by simulating the operation during validation). Additionally, paymasters can include a margin of safety in their deposit. Any unused gas fee from the operation will be refunded back to the paymaster at the end of execution. Designing the paymaster's logic to handle gas costs carefully is important – if it underestimates, it might run out of funds mid-operation (which would cause the transaction to fail and could slash its stake).
- **Post-Operation Settlement:** After the EntryPoint executes a sponsored operation, it will deduct the actual gas fee from the paymaster's deposit and credit the block producer (paying the miner/validator). At this point, the paymaster's job isn't necessarily done – it can execute additional logic *after* the operation as part of its contract. For example, the paymaster might record that user X used service Y at a certain time (for analytics or billing purposes), or it might update an internal balance that tracks how much the user owes (if the user agreed to reimburse later). In some cases, the paymaster might immediately trigger a mechanism to get paid back – for instance, pulling tokens from the user's wallet or swapping some funds on a DEX to recoup the $CENT it spent. This post-operation phase allows the paymaster to settle accounts and maintain its economic viability.

In practice, many kinds of paymaster contracts can be deployed to serve different needs. For example, a **ContractFactory Paymaster** could sponsor gas for deploying new contracts (to encourage developers by covering their deployment costs), a **TransferGate Paymaster** might only sponsor transactions that move through a specific router or comply with certain policies, and a **DEX Paymaster** could accept various tokens from users and handle converting them to $CENT via on-chain liquidity. These examples show how paymasters can be tailored – one might focus on user onboarding, another on promoting activity in a dApp, and another on enabling true multi-token gas payments.

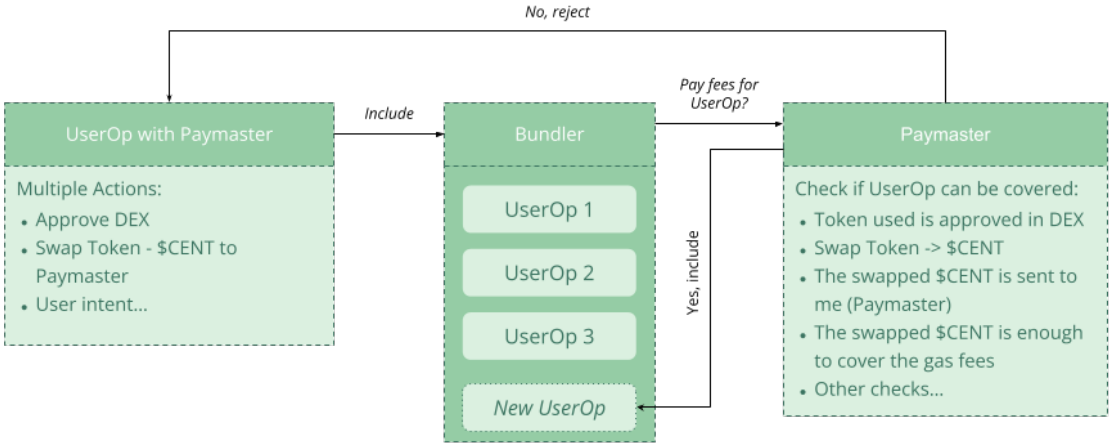
Overall, paymasters are a powerful tool in Incentiv's AAA model to eliminate the "gas token" hurdle and enable rich fee mechanisms. They introduce some complexity in design, but the protocol's rules (staking, validation checks, etc.) mitigate risks and ensure that paymasters operate safely within the network.

5.6 Unified Token Feature

5.6.1 Architecture Overview

Building on the power of paymasters, Incentiv introduces a *Unified Token* mechanism for fee payment. The goal is simple: **users should be able to use the network seamlessly with whatever tokens they have**, without stopping to think about gas and the native coin of the network. The unified token feature works through a combination of paymaster delegation and on-chain token conversion, orchestrated by the protocol:

- **Paymaster-Based Delegation:** When a user submits a transaction (UserOperation), they can designate a paymaster to handle the fees. If the user wants to pay fees in a token other than $CENT, a paymaster for that token steps in. Essentially, the paymaster fronts the required $CENT to pay the network operators, allowing the transaction to be executed, and in return the user will give up a small amount of their chosen token as the fee. The user's experience is that they paid, say, in USDC or DAI or any token – they never needed to touch $CENT. This delegation removes the circular dependency of "you need $CENT to do anything, including obtain $CENT", while still allowing $CENT to fulfill its purpose as the native coin of the network.
- **Automated Conversion via Router:** Once the paymaster has agreed to sponsor the transaction, the protocol handles conversion of the user's payment token into $CENT behind the scenes. Incentiv's infrastructure includes a *router* that is connected to the network's native decentralized exchange (DEX) and possibly other liquidity sources. After the transaction is executed, the router will automatically swap the fee amount (in the user's token) for an equivalent amount of $CENT, to reimburse the paymaster's reserves. For example, if the user was paying in TokenX, the router finds the best rate (using on-chain liquidity pools) to convert a bit of TokenX into the needed amount of $CENT. This conversion is done at market rates and can tap into the native DEX or external DEXs/bridges as needed, to ensure the paymaster is not left exposed to the market. The key point is that this happens seamlessly and atomically with the transaction's execution.
- **Native Price Discovery:** The conversion rates for the unified token feature are determined by on-chain price discovery rather than any fixed oracle. That means the cost to pay in an alternate token is always based on current market exchange rates for that token versus $CENT. This approach improves security and fairness – it's hard to game or manipulate because it uses the same pricing everyone sees on the DEX. The user effectively pays the "true" cost of gas in their token (plus perhaps a small premium for the service), and the network doesn't subsidize or arbitrarily set rates. It's all driven by real liquidity.
- **Reliability Fallbacks:** In the rare case that the user's chosen token doesn't have sufficient liquidity or there's an issue with conversion (say the DEX price slippage would be too high, or the token's value suddenly plummeted), the system has fallback strategies. The transaction might be temporarily delayed while the router looks for alternate liquidity (maybe splitting the trade across multiple pools or using a different path), or the user could be prompted (via their wallet UI) to top-up or choose a different token for the fee (not $CENT and not necessarily the token used for the transaction itself). The design prioritizes completing the user's transaction safely – so it won't, for example, execute and then find out the fee couldn't be converted or leave the paymaster under-compensated. However, such scenarios should be extremely rare, especially for blue-chip or other liquid tokens.



5.6.2 User Experience

The Unified Token model fundamentally changes the day-to-day experience of using the blockchain. For users (and developers creating user-facing apps), it means interacting with Incentiv without worrying about the *type* of token needed for gas. Here are some additional implications of this model:

- **Batch-Aware Fee Resolution:** When multiple operations are bundled (say a user action triggers calls to multiple contracts or a series of tasks), the unified token model ensures that all parts of the bundle are handled under the same fee abstraction. Even if different parts of the bundle could theoretically be paid by different entities, the system coordinates so that the user doesn't end up with a half-executed batch due to fee issues. For example, if two actions in a bundle use different paymasters (one perhaps sponsored by a dApp, and another using the user's token), the EntryPoint and router will ensure that the total fees are settled properly without requiring separate gas payments from the user for each part. It all feels like one transaction with one fee (from the user's viewpoint).
- **Clear Cost Representation:** Wallets can present transaction fees to users in their preferred terms. Instead of quoting gas prices in gwei or showing network fees in $CENT (which a newcomer might find confusing – "what's this amount of CENT and why do I need it?"), the wallet can say "Fee: 0.05 USDC" or "Fee: 10 XYZ tokens" directly. Since the conversion is known at the time of simulation, the exact cost in the user's token of choice is clear. This clarity not only improves transparency but also helps users mentally evaluate costs (most people can judge a dollar amount better than 0.0001 of a coin).
- **Token-Agnostic Onboarding:** For new users coming to Incentiv, the unified token feature is a game-changer. They can be onboarded by receiving any token – maybe an airdrop of a stablecoin or some reward points token – and they can immediately start using the network with that. They don't face the usual hurdle of "you have tokens but you cannot transact as you do not hold the native token required for gas payment." This makes user acquisition and growth much smoother. A dApp could literally send a user an NFT or some tokens, and the user can interact right away, because the dApp's paymaster will accept a bit of those tokens for gas when the user transacts.
- **Scheduled and Automated Operations:** Many users may set up automated tasks (like the weekly payment example, or periodic investments via a DeFi app). With unified token gas, these scheduled operations won't suddenly fail months later because the user forgot to maintain a $CENT balance. As long as the user has *something* of value in their wallet at execution time, the paymaster system can convert a piece of it to gas. This

makes long-term automation reliable – you could set something today and even if a year from now you only hold different assets, the network can still carry out your instruction.

- **Inbound Token Consolidation:** As an added convenience feature, smart wallets could use the unified token mechanism to streamline their holdings. For instance, a user might configure their wallet such that whenever they receive various tokens (say as payments or rewards), those are automatically converted into a single preferred token or a set ratio of tokens. Suppose Alice wants to keep most of her funds in DAI stablecoin. If someone pays her in ETH or $CENT or any other token, her wallet's logic (using the router and perhaps a paymaster) could immediately swap those into DAI. This way, Alice doesn't have to manage conversions manually, and she always has her balance in her asset of choice. Moreover, this kind of feature can work hand-in-hand with fee abstraction – part of the conversion can also allocate a tiny portion to pay for the conversion gas itself, creating a self-contained flow.

In summary, the unified token feature makes the *act* of paying for transactions almost invisible. Users engage with the network in the tokens they care about, and Incentiv's protocol, via paymasters and the router, handles the behind-the-scenes exchange to settle fees. This is a major step toward making blockchain usable by mainstream audiences, who are accustomed to apps handling fees or embedding them in the service cost (think of how you don't directly pay an "AWS fee" when using a web app – the app takes care of its infrastructure costs). Here, the fee is managed in a flexible way that adapts to the user's context.

5.7 Security and Risk Mitigation

5.7.1 UserOperation Validation Safeguards

Introducing greater flexibility with account abstraction also introduces new surfaces for potential abuse. Incentiv's design incorporates several safeguards at the protocol level to ensure that UserOperations are safe and cannot be exploited to harm the network or other users:

- **Replay Protection:** Each UserOperation includes a nonce and a chain identifier to ensure it can't be replayed on another chain or reused on the same chain. This is analogous to Ethereum transactions having a nonce and chain ID. The EntryPoint will reject any UserOperation that has a nonce that has already been used by that smart wallet or one that doesn't match the current chain. This prevents attackers from capturing a signed UserOperation and trying to execute it multiple times or on sister networks.
- **Simulation Verification:** Before the network accepts a UserOperation into a block, it must pass a simulation (dry-run) check. Bundlers perform this simulation as part of their workflow (and validators may do so as well before finalizing a block). If the operation would fail (due to, say, running out of gas or a failed required condition in a contract), it won't be included. This prevents a class of griefing attacks where someone could spam the mempool with operations that always fail and waste everyone's time. It also saves users from paying for failed transactions – if it's going to fail, it won't be published.
- **Storage Access Restrictions in Verification:** During the verification phase (when the EntryPoint or paymaster checks signatures and balances), the UserOperation is not allowed to arbitrarily read or modify contract storage outside certain bounds. In other words, the verification logic (like wallet signature check or paymaster logic) should be *pure* or at least not depend on random parts of the blockchain state. This rule ensures that verification remains deterministic and cannot be abused to, for example, create side effects or do things like oracle reads that could be manipulated. It also means a malicious user can't craft an operation that

tries to exploit the verification phase to do something sneaky. Everything important should happen in the execution phase, after all checks pass.

- **Deterministic Gas Usage in Validation:** Similarly, the validation of a UserOperation (wallet and paymaster checks) is designed to consume a predictable amount of gas and not loop unboundedly. The protocol might enforce a gas limit for validation steps. This prevents denial-of-service scenarios where someone submits an operation with an extremely complicated validation that could tie up a bundler or validator. By keeping the validation step bounded and straightforward (basically just signature checks and a few calculations), the network can safely handle large volumes of UserOperations without per-op validation costs spiraling.

5.7.2 Paymaster Risk Management

Paymasters, by their nature, deal with money and trust. A malicious or buggy paymaster could try to exploit the system, so Incentiv includes strong checks and balances for paymasters:

- **Stake Slashing:** Every paymaster must put down a security stake. If a paymaster violates the rules – for example, if its validation function improperly approves an operation that then fails to pay for gas, or if it tries to withdraw funds it shouldn't – the protocol can slash (take away) some or all of that stake. This is a powerful economic deterrent. It means any paymaster has a lot to lose by misbehaving, so rational paymasters will be careful to only sponsor legitimate operations that they are truly willing and able to pay for.
- **Rate Limiting:** Paymasters can set internal limits on how much they are willing to sponsor in a given time frame, and the protocol could also enforce some limits. This prevents scenarios like a single paymaster suddenly being drained by a flood of requests. For instance, a paymaster might only accept X number of UserOps per block or cap the total gas it will sponsor per hour. If an attacker tries to abuse a generous paymaster, these limits act as a circuit breaker, giving the paymaster owner time to react (or the contract logic time to shut off sponsorship if a threshold is exceeded).
- **Whitelisting and Scope:** Many paymasters will choose to only sponsor certain users or certain types of operations. The protocol fully allows this – paymasters are not required to serve everyone. For example, a DeFi project's paymaster might only sponsor transactions that interact with its own contracts, and ignore others. Or it might be invite-only: only users who have an NFT from the project get their gas paid. This selective approach means paymasters can manage their risk by narrowing the scope of what they pay for. If something falls outside their criteria, they simply don't sponsor it (and the user would have to use a different paymaster or pay in $CENT). From a security perspective, this is good because it limits the "blast radius" if something goes wrong – issues in one dApp's logic won't drain another paymaster that has nothing to do with it.
- **Economic Bounds:** The Incentiv protocol can enforce upper bounds on sponsorship to prevent catastrophic losses. For instance, it might refuse to let a paymaster sponsor a single operation that would cost more than, say, 1% of the paymaster's total stake or deposit. This way, even if there's a bug or an oversight, the paymaster can't be tricked into footing an outrageously large gas bill unexpectedly. It sets an implicit budget per operation. This kind of safeguard ensures that even edge cases or extreme scenarios won't bankrupt a paymaster in one go.

5.7.3 Bundler Security

Bundlers operate in an open marketplace and have to deal with arbitrary user submissions, so they also need mechanisms to protect against abuse and to maintain quality of service:

- **Reputation Tracking:** Bundlers can maintain a reputation score or blacklist for wallets and paymasters. For example, if a particular smart wallet has a history of submitting operations that always fail (perhaps someone is testing the system or trying to DoS), a bundler might deprioritize or even temporarily ban operations from that wallet. Similarly, if a paymaster's operations often end up not paying (maybe the paymaster is underfunded or buggy), bundlers will avoid those to not risk carrying out work that won't get compensated. This is done in a decentralized way – each bundler can decide its own criteria, and over time this creates a form of "soft trust" network-wide, where good actors get quick service and bad actors find it hard to get their transactions included.
- **Economic Filtering:** To defend against spam, bundlers will typically require a minimum fee or have a fee market of their own for UserOps. If someone submits a large number of operations with very low fees, bundlers will naturally pick the higher-paying ones first. Extremely low-fee ops might get dropped or ignored if the bundler is busy. Additionally, bundlers might require that paymasters put up a certain minimum stake or security bond before they consider those operations – ensuring that the bundler has some assurance of being paid. Essentially, while the network as a whole is open, bundlers as services can set economic thresholds to ensure they are dealing with serious transactions and not being overwhelmed by junk.
- **Private Mempools:** Some bundlers may offer a private submission channel for UserOperations (for a premium service or sensitive transactions). In a private mempool, a user can send their operation encrypted or directly to a trusted bundler, so it doesn't appear in the public pool for others to see or potentially front-run. This is especially useful for transactions that involve trades or other MEV-sensitive actions – the bundler will include them without broadcasting the intent beforehand. This not only helps users retain privacy and value (no one can sandwich your trade if they don't see it until it's executed), but also can improve efficiency by reducing competition on certain operations. It's up to each bundler to decide if they provide such a service, but the protocol supports it since the bundler just needs to pass the operation to EntryPoint like any other.
- **Parallelized Verification:** To achieve high throughput, bundlers can validate multiple UserOperations simultaneously. Because each UserOperation is largely independent (except for some that might conflict by design, which bundlers detect), a bundler can utilize parallel processing – modern servers with multi-core CPUs can simulate and check many ops at once. Incentiv's reference implementations encourage this, so a bundler isn't bottlenecked by sequentially checking each operation one by one. The result is that even if thousands of users submit operations at the same time, a well-implemented bundler can handle the load without lag, by scaling across CPU cores or even distributed machines. This keeps the network responsive and ensures the benefits of AA don't come at the cost of performance.

Overall, these security measures ensure that the flexibility of AAA does not open doors to instability. Every new component (wallets, bundlers, paymasters) operates within rules and economic constraints that align their incentives with the health of the network. Just as Ethereum's miners/validators follow the rules because of protocol incentives, Incentiv's bundlers and paymasters also follow rules because it's in their economic interest. And users, on their part, get a safer, smoother experience with many built-in protections against common pitfalls.

5.9 Beyond ERC-4337: Protocol-Native Integration

5.9.1 Key Differences from ERC-4337

While Incentiv's account abstraction design is inspired by the groundwork of ERC-4337 on Ethereum, implementing AA natively at the protocol layer (instead of as an add-on) unlocks a number of improvements and efficiencies:

- **Consensus-Level Integration:** In Incentiv, account abstraction hooks directly into the consensus and block production process. This means the network can offer features like *priority execution* for certain UserOperations (e.g. system-level operations or emergency recovery ops might be given special treatment in blocks), and it can guarantee inclusion rules that an off-chain system cannot. By comparison, ERC-4337 operates as a higher-layer service; the base blockchain doesn't know about UserOperations or EntryPoint logic – it just sees regular transactions from bundlers. Incentiv's validators, on the other hand, are aware of UserOperations in blocks and could potentially optimize or handle them differently if needed, because it's part of the protocol.
- **Reduced Overhead:** The native approach removes the extra indirection and overhead that ERC-4337 has. On Ethereum, every UserOperation ends up packaged in a transaction calling the EntryPoint, which adds calldata overhead, additional gas for the EntryPoint's logic, and some redundancy (like the need to pass the same EntryPoint address each time, etc.). Incentiv can streamline this since it doesn't need to shoehorn operations into a transaction – it can treat them as first-class objects. Data structures and execution paths are optimized specifically for UserOperations. This can lower gas costs and improve throughput, because we cut out the "middleman" of the external EntryPoint contract's layer.
- **Unified Fee Market:** In Ethereum's ERC-4337, the fee market for UserOperations is somewhat separate from the native fee market, and bundlers have to balance between the two. In Incentiv, because everything is unified, the economics are simpler. There's one coherent system for fees and incentives that covers both normal transactions and abstracted transactions. For example, a validator in Incentiv directly sees the fees offered by UserOperations and can include them in blocks accordingly, rather than relying on an outside bundler's decisions. This alignment means that improvements like EIP-1559 fee burning or future changes in the fee mechanism can uniformly apply to user operations as well.
- **Stronger Security Guarantees:** By baking account abstraction into the protocol, Incentiv can enforce security constraints at a lower level. For instance, it can ensure that certain critical checks (like paymaster stake verification or wallet bytecode verification) are part of the block validation itself, not just left to an honest assumption on bundlers. If Ethereum's EntryPoint contract had a bug or needed an upgrade, that would require coordination off-chain; in Incentiv, such logic is part of the protocol and benefits from the same scrutiny and upgrade path as other core protocol features. In short, there are fewer moving parts that rely on "code running as a service" and more that are simply "code that *is* the blockchain."

5.9.2 Future Directions

Having account abstraction as a native feature of the blockchain opens the door to further innovations down the line. Some forward-looking possibilities that the Incentiv architecture could explore include:

- **Cross-Chain Abstraction:** In the future, Incentiv could enable UserOperations that span multiple chains or domains. For example, a single user intent might initiate an action on Incentiv L1 and also trigger something on an L2 or another connected blockchain. With a standardized abstraction layer, the user wouldn't need to handle the bridging or multiple transactions – the network (possibly via cross-chain bundlers or relays) could abstract that complexity. This would make experiences like moving assets or invoking cross-chain dApps much more seamless.
- **AI-Powered Wallet Optimization:** As the ecosystem grows, the choices for how to execute a given intent (which bundler to use, which DEX path to take for conversion, when to execute a scheduled task for best price, etc.) become complex. Smart contract wallets could integrate AI or machine learning components that learn

from user behavior and market conditions to optimize these choices. For instance, an AI module could decide to delay a non-urgent transaction by a few minutes because it predicts gas prices will drop, or choose one DEX over another because historically it gave better rates at this time of day. The account abstraction layer makes such decisions autonomous – the wallet can act in the user's best interest without constant input.

- **Social and Collaborative Features:** Native AA also means the blockchain can support capabilities such as social recovery and multi-user accounts at a fundamental level. In the future, we might see **collaborative wallets** (like DAO-controlled accounts or family wallets) that are natively recognized by the chain's rules (e.g. a transaction might require signatures from 3 of 5 family members). Reputation systems could be built in, where wallets gain a reputation score based on past behavior (useful for paymasters or bundlers to decide on service quality). Also, because accounts are contracts, they could natively support things like **social feeds** or status indicators (imagine being able to set an "out of office" or "compromised" flag on your wallet that others can see and that changes its behavior). These kinds of features blur the line between purely financial accounts and general user identity.
- **Built-in Compliance Tools:** While blockchains are global and permissionless, certain users (especially businesses) have compliance requirements. A programmable wallet could be configured to automatically enforce compliance rules – for example, logging all transactions above a threshold for audit, or only allowing transfers to addresses that have been KYC-verified if the user opts into such restrictions. By having this at the wallet level, the user remains in control (the blockchain isn't censoring anything network-wide, it's the user's wallet choosing to self-regulate according to the user's needs or local laws). In the future, Incentiv could provide templates for "compliant wallets" that companies could use to satisfy regulatory needs without relying on centralized exchanges or custodians.

In conclusion, the Advanced Account Abstraction (AAA) layer transforms the blockchain from a technical playground into a user-centric platform. By hiding complexity while exposing powerful capabilities, Incentiv's design makes decentralized apps and services accessible to mainstream users – all without compromising the core values of decentralization and security. This isn't just an incremental improvement to the status quo; it's a fundamental reimagining of how people can interact with a blockchain network. Instead of adapting to the technology's quirks, the technology adapts to the users' needs. This paradigm shift paves the way for the next generation of blockchain applications and, ultimately, broader adoption of decentralized technology.

6. Consensus

6.1. From Raw Physics to Adaptive Security Budgets

Modern Layer-1 networks already secure value flows on the scale of mid-sized nation-states. Any mechanism charged with protecting such capital must satisfy three **non-negotiable constraints**:

Constraint	Practical test	Failure mode if unmet
Open access	A new validator can join with nothing more than commodity hardware and an internet link.	De-facto permissioning or cartel formation.
Physics-priced tamper cost	Re-writing N blocks requires spending $\geq O(N)$ in irreversible real-world resources.	Low-cost history rewrites; economic finality collapses.
Elastic security budget	The network self-adjusts its spend on defence in step with the economic value it now safeguards.	Chronic over-spend (waste) or under-spend (risk).

6.2. Why Proof-of-Work Survives the Cut

After auditing all candidate primitives, **hash-power rented from the open energy market remains the only resource whose attack curve is *linear in physics*:** every incremental unit of influence burns proportional electricity, with no discounted path via governance or capital leverage. That irreducible cost floor is why security engineers continue to treat classic Proof-of-Work (PoW) as the reference adversary model.

6.3. What Classic PoW Gets Wrong Economically

While cryptographically sound, vanilla PoW ossifies three economic dead-zones:

1. **Static subsidies** → miners earn the same reward whether a block carries $0 or $1B in value, encouraging empty blocks and hash-arms races.
2. **Emission rigidity** → security budget cannot expand in bull-markets nor contract in bear-markets, delivering either wasted spend or weakened liveness.
3. **Hash-only elections** → operators with no stake in token value can rationally sabotage if short-term gains outweigh one-block revenue.

6.4.Incentiv's Demand-Coupled Remedy

Incentiv preserves the physics-priced backbone and closes the economic feedback loop:

Legacy artefact	Design lever	Outcome

Fixed subsidy	**Value-Fee feedback:** an ad-valorem levy channels yesterday's usage into tomorrow's block budget.	Security expenditure auto-scales with real demand.
Hash-only reward	**Holdings multiplier *M(H)*:** extra payout proportional to miner's on-chain CENT balance.	Aligns miner incentives with token health; raises cost of sabotage.
One-way emission	**Tapered floor subsidy:** deterministic, declining curve.	Guarantees liveness in low-traffic epochs without perpetual inflation.
ASIC monoculture risk	**Memory-hard Ethash + upgrade runway.**	Sustains heterogeneous miner set; green-switch path retained.

The schematic above meets the three constraints stated at the start.

6.5. Consensus Model Overview

Classic PoW delivers censorship-resistant finality through brute economic force, yet its vanilla economics are mismatched to live networks. Incentiv keeps the cryptographic certainty of PoW **but upgrades its economic circuit so that security spend rises and falls with the network it protects.**

Guiding premise: *blocks should earn what they are worth — no more and no less.*

When utilisation surges, fees automatically thicken the reward pool and hash-rate follows; when traffic subsides, a tapered subsidy guarantees liveness without over-paying for idle security.

Key components

- **Security primitive — Hash-power.** A valid block must present an Ethash solution below the current target. There is *no* alternate path to finality.
- **Economic alignment — Fee-indexed reward basin.** 100 % of per-transaction gas is paid instantly to the block author. A distinct **Value Fee** — an ad-valorem levy on every transfer — accrues into a forward-funded reward contract. Epoch *n + 1* is therefore capitalised by the economic reality of epoch *n*.
- **Counter-cyclical subsidy.** When aggregate usage is muted, a deterministic subsidy curve underwrites a non-zero floor reward, ensuring miner retention without long-run inflation.

For rational miners the dominant strategy is clear: **maximise hash-rate and foster organic network utilisation — any censorship or degradation directly erodes their future cash-flow.**

6.6. Ethash Mining Mechanism (Concise)

Ethash is retained unmodified because its security objective is *memory-bandwidth saturation:* ASICs gain little advantage over general-purpose hardware once memory bandwidth is the bottleneck. The algorithm proceeds in three stages per block:

Incentiv

Ethash Algorithm (Per Block)

1. **DAG Generation**

   For each **epoch** defined as:

   $$e = \left\lfloor \frac{\text{height}}{30{,}000} \right\rfloor$$

   Miners rebuild a **Directed Acyclic Graph** D_e from a 16 MiB cache C_e.

   The DAG size grows **linearly** with each epoch:

   $$\text{Size}_e \approx 6.05 \text{ GiB at genesis} + (8.192 \text{ MiB} \times e)$$

2. **Hashimoto Mixing**

   For each nonce n, the miner performs:

   - 128 pseudo-random 64-byte reads from D_e
   - Folds these using **FNV-64** into a 256-bit **mix digest**

3. **Proof Validation**

   The result must satisfy:

   $$\text{digest} < \text{target}$$

   The **target** is adjusted via the **difficulty governor** to maintain the expected block time:

   $$\tau = 15 \text{ seconds}$$

Difficulty Governor (first-order stable controller)

$$D_{n+1} = D_n \left(1 + k\left(\frac{\Delta t}{\tau} - 1\right)\right), \quad k = \frac{1}{2048}$$

Where:

- D_n: current difficulty
- Δt: actual time between block n and n+1
- τ: target block time (15 seconds)
- k: feedback gain constant

How it Works:

- $If \Delta t > \tau, then \left(\frac{\Delta t}{\tau} - 1\right) > 0$:
  → Blocks are too slow → difficulty increases.
- $If \Delta t < \tau, then \left(\frac{\Delta t}{\tau} - 1\right) < 0$:

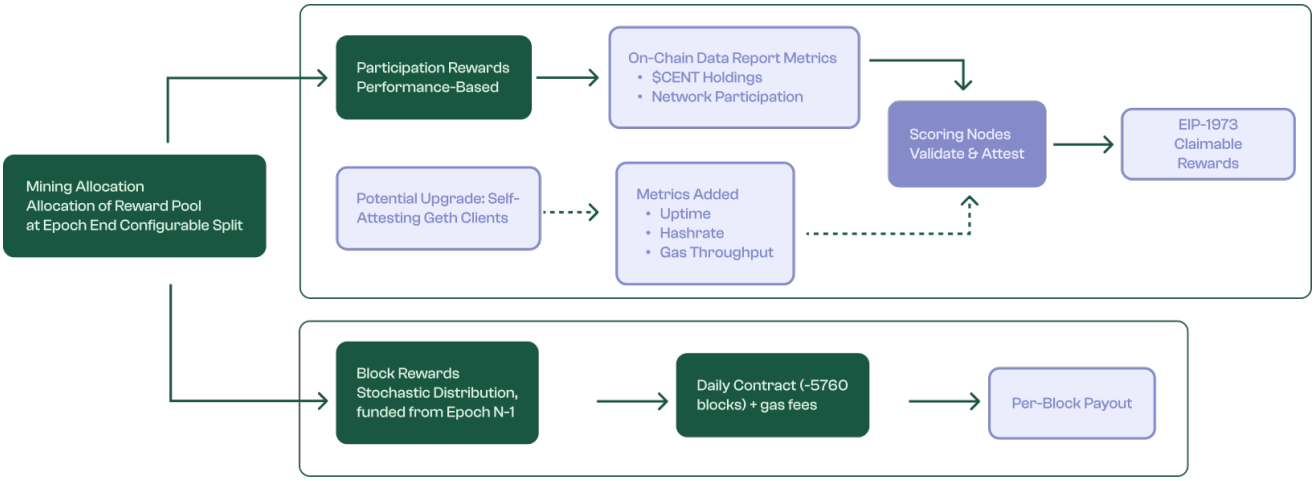
→ Blocks are too fast → difficulty decreases.

Why It Works:

- This is a **first-order control system** with a **pole at** 1 - k.
- With 0 < k < 2 the system converges to the target interval (monotonic when k < 1).
- The chosen gain $k = \frac{1}{2048}$ is calibrated for:
  - Stable convergence over ~90 blocks
  - Minimal oscillation or overshoot

6.7. Mining Rewards — Deterministic Economics

Incentiv pays miners through a two-layer contract lattice: **Base-Reward** and **$CENT Holdings**.



6.7.1 Epochic Base-Reward

1. **Budget formulation.** At epoch boundary t_n two inflows are transferred atomically into *RewardPool*.
   - **Subsidy(t_n)** — scheduled, decaying block subsidy.
   - **ValueFee(t_n−1)** — aggregate ad-valorem fees from the preceding epoch.
2. **Per-block determinism.**

$$R_{\text{base}} := \frac{\text{Budget}n}{N\text{target}}, \qquad N_{\text{target}} \approx 5{,}760 \text{ blocks/epoch}$$

Every block in epoch *n* therefore yields exactly *Rbase* CENT, independent of luck.

3. **Remainder roll-forward.** Any unspent tokens ("missed blocks") roll into the next epoch's budget — supply never leaks.

Equation check. *Rbase* is simply a *quota*: total budget divided by target block count. Because both inputs are on-chain and deterministic, anyone can recompute the number and verify payouts.

6.7.2 Aligned Holdings Multiplier

Block authors additionally claim *M(H) × Rbase* where *H* is the author's on-chain CENT balance at height *h*.

$$M(H) = 1 + \alpha \cdot log_{10}(H/H_0)$$

subject to: $0 \leq \alpha \leq 1, and H \geq H_0$

- **Interpretation.** Above a threshold H_0, each decade increase in holdings adds $\alpha \cdot Rbase$ to the reward. Below H_0, *M(H) = 1* so small miners are not penalised.
- **Funding discipline.** The multiplier is carved from the *same* per-block budget, so total emission remains fixed.

6.7.3 Variance Management

- **Solo vs Pool.** Pools reduce variance but dilute the multiplier because only the pool's aggregate balance counts. Operators therefore choose between higher expected value (solo) and smoother payouts (pool).
- **Mean-reversion.** Because *Rbase* is fee-indexed, revenue naturally mean-reverts rather than swinging jackpot-style.

6.8. Network-Security Analysis

Security rests on two pillars: (i) cryptographic work (Ethash) and (ii) an economically self-reinforcing funding loop that guarantees a hard-to-bribe hash-rate even in the chain's infancy.

6.8.1 Adversarial Cost Models

Symbol	Meaning
Htot	Live network hash-rate = *Hmin + Hutil*
chash	Amortised cost [USD · s^{-1} · H^{-1}] of renting/buying Ethash power
Rbase	Deterministic reward per block in the current epoch
$M(H_a)$	Multiplier of the attacker's address
τ	Block interval (15 s)

51 % attack cost. To rewrite *L* blocks the adversary must out-hash the honest chain for the same duration *plus* forgo honest earnings:

$$C51\,(L) = 0.51 Htot\,chash\,L\tau + \sum_{i=0}^{L-1} Rbase\,M(Ha)\delta^{i}$$

where $\delta \geq 0.999999$ is a per-block discount factor capturing the time-value of money.

- **Left term (cap-ex).** Real-world power or rental payments proportional to attack length and required hash-rate.
- **Right term (op-ex).** Honest revenue the attacker forfeits by mining on a private fork.
- **Sanity check.** Both terms scale linearly with $L \Longrightarrow$ tamper cost is $O(N)$ in physics, honouring the second constraint.

Selfish mining profit ratio. Modifying the classic Eyal-Sirer model to include the multiplier:

$$\Pi selfish \Pi honest = \alpha(1-\alpha)^2 - \alpha + \alpha 2 \;\; 1M(Ha), \frac{\Pi_{\text{selfish}}}{\Pi_{\text{honest}}} = \frac{\alpha^2(1-\alpha)}{\alpha^2+\alpha-1} \; \frac{1}{M(H_a)}$$

where α is the attacker's hash share. Because $M(H) \geq 1.02$ above threshold holdings, the ratio stays < 1 for $\alpha \leq 0.25$, eliminating profitable selfish mining in the realistic range.

6.8.2 Evolving Deterrence Gradient

Budget = Subsidy + ValueFee. As real transaction volume v rises, both **hash-rate** and **per-block reward** rise, so the slope $\partial C_{51}/\partial v \gg 0$. In plain words, **users automatically "buy" stronger finality by using the chain.**

6.9 Take-aways

- **Physics-priced deterrence, economically tuned.** Attack cost grows linearly with attempted damage and is funded endogenously.
- **Aligned incentives.** Miners with token exposure earn more, encouraging stake-holding and honest behaviour.
- **Elastic yet deterministic spend.** The chain never over- or under-pays for security: yesterday's utilisation wires tomorrow's budget.

Appendix B — Subsidy Curve (informative)

A simple exponential decay with floor:

$$Subsidy(t) = S0\; e - \lambda t + Smin, \lambda \approx 0.115 \;\; yr - 1. \text{Subsidy}(t) = S_0\, e^{-\lambda t} + S_{\min}, \quad \lambda \approx 0.115\ \text{yr}^{-1}.$$

- S_0 sets genesis reward; S_{min} (≈ 2 % of S_0) guarantees liveness even if usage drops to zero.
- The curve is *quantised monthly* on-chain so miners can verify the exact token flow.

7. $CENT Token - Incentiv Native Token

7.1 Token Fundamentals

The **$CENT** token is designed as the coordination mechanism at the heart of the Incentiv economy. Rather than acting merely as a medium of exchange or speculative asset, $CENT aligns the incentives of all participants across the network in service of collective value creation. This philosophy builds directly on Incentiv's contribution-centric principles: the network rewards those who add measurable value, ensuring that growth emerges from shared contribution rather than passive capital alone. In practice, $CENT ties together technical infrastructure and economic behavior, providing an engine for mutual benefit from individual participation.

$CENT follows a fixed-supply model with a carefully planned distribution. The total supply is **100 billion** tokens, a quantity chosen to provide ample granularity for a global user base. The supply is neither inflationary through mining nor deflationary through burning; all tokens are minted at genesis and circulate through the ecosystem. Network operations are funded not by minting new tokens (which would dilute existing holders) or by destroying tokens (which creates artificial scarcity), but through **fee recycling**: transaction fees collected by the protocol are cycled back as rewards. This design yields predictable economics that participants can model and trust. Every aspect of $CENT's tokenomics is geared toward ensuring that as the network grows, all participants benefit proportionally, with no hidden wealth transfers or unsustainable monetary tricks.

7.2 Utility Functions

As the native unit of account in the Incentiv protocol, $CENT serves as much more than a fee token – it is the economic glue that enables complex coordination throughout the ecosystem. Key functions of $CENT include:

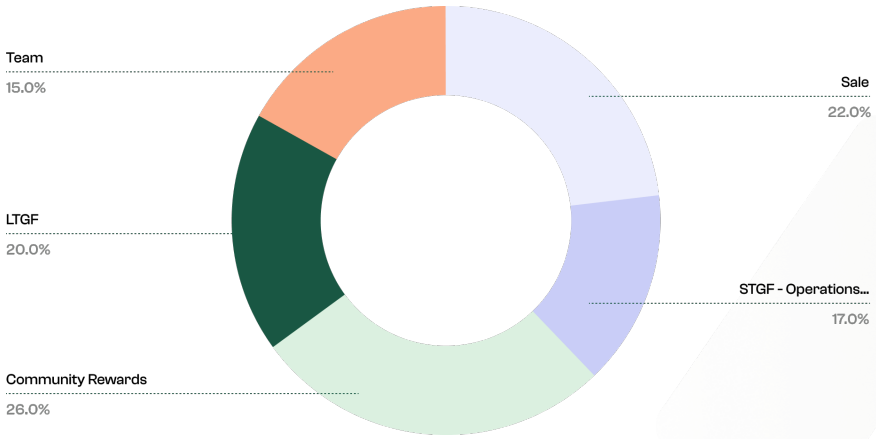
- **Fee Payment:** Users can pay transaction fees in any token via the network's paymaster system, but behind the scenes those fees are converted on-chain into $CENT before entering the communal reward pool. This mechanism creates a consistent baseline demand for $CENT directly tied to network usage.
- **Reward Distribution:** All rewards from the unified reward pool are disbursed in $CENT. By denominating rewards in the native token, the system ensures a common unit of value for contributions across miners, developers, users, bundlers and liquidity providers. $CENT thus becomes the vehicle through which the network redistributes the value created on-chain back to its contributors.
- **Contribution Scoring Influence:** Holding $CENT contributes to a participant's contribution score, reflecting long-term commitment to the ecosystem. Token holdings are one factor in the multi-dimensional scoring algorithm that determines reward shares. This provides an extra incentive for sustained engagement without allowing wealth alone to trump merit – a higher $CENT stake modestly boosts one's score but does not replace the need to actively contribute.
- **Liquidity Backbone:** $CENT is the principal reserve asset for liquidity within the network's native DEX. Most trading pairs in the native DEX involve $CENT, promoting deeper liquidity and efficient conversion for fees and other transactions. As the ecosystem grows, this central liquidity role means that demand for various services and tokens ultimately strengthens $CENT markets, reinforcing network effects and price stability.
- **Governance:** $CENT also embodies governance rights in the protocol. Token holders can participate in on-chain voting to influence key economic parameters and future upgrades. In essence, those who hold $CENT have a voice in the network's direction – aligning decision-making power with those most invested in the network's success and long-term health.

In tandem, the $CENT token design aligns the interests of all stakeholder groups, creating mutual opportunities. Every major participant in the ecosystem benefits from $CENT's role in a way that incentivizes them to support the network:

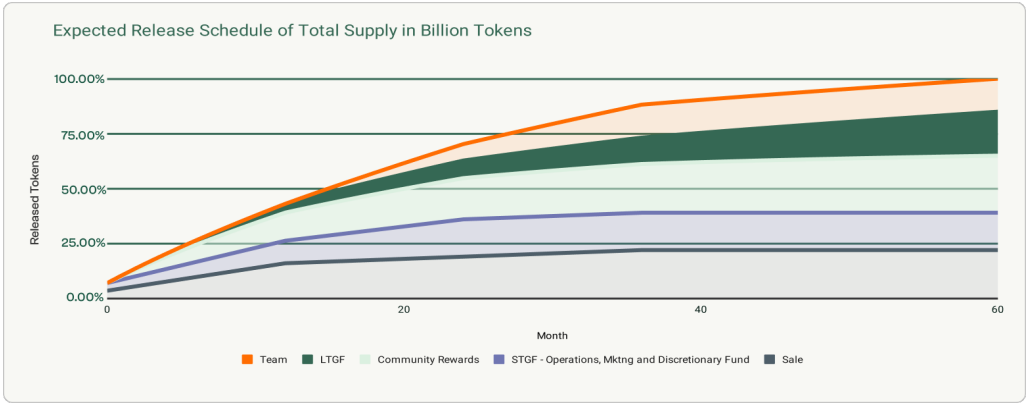
- **Miner Alignment:** Miners receive block rewards and fee distributions in $CENT, giving them a vested interest in the network's success. Any attack or behavior that undermines the network would directly devalue the tokens they have earned. This economic exposure naturally discourages malicious activity – securing the chain not just through hash power, but through miners' own stake in the ecosystem's prosperity.
- **Developer Incentives:** Developers earn $CENT rewards proportional to the usage of their dApps and smart contracts. A popular application will generate more fees and thus more rewards for its creator. Successful developers thereby accumulate $CENT over time, aligning their interests with the growth of the overall platform. Their personal upside increases as more users engage with the network, motivating developers to build valuable, high-quality services.
- **User Benefits:** Active users are rewarded in $CENT for their on-chain participation, which helps offset their transaction costs and creates a sense of ownership in the network. The more a user interacts – whether by transacting, providing services, or contributing data – the more they earn. Over the long term, engaged users can accumulate meaningful $CENT stakes, giving them greater influence (via improved contribution scores and governance) and aligning their future with the network's trajectory.
- **Liquidity Providers:** Liquidity providers (LPs) are rewarded with additional $CENT from the reward pool. By strengthening key liquidity corridors in the DEX, LPs perform a service that benefits all users, and they are compensated accordingly. This encourages the provision of deep liquidity exactly where the ecosystem needs it most. The result is a more robust and efficient market for $CENT and other assets, reducing slippage and improving the overall efficiency of economic activity on Incentiv.

7.3 Token Distribution and Vesting

The distribution of the $CENT supply is carefully structured to balance multiple objectives: reward early adopters and contributors, fund ongoing development, bootstrap network growth, and ensure long-term incentives remain strong. Each allocation of the 100 billion tokens is tailored to a specific strategic purpose and comes with appropriate vesting or release conditions to align with network growth. Table 7.3 below outlines the planned token allocations:



CATEGORY	Allocation	Lock-up and Vesting
Sale	22%	• 3.5% are released on TGE. • 12.5% are vested during the first 12 months • 6% are vested between month 13 to month 24.
Short Term Growth Fund	17%	24 months vesting
Community Rewards	26%	60 months vesting
Long Term Growth Fund	20%	60 months vesting
Team and Advisors	15%	12 Months lock-up followed by 24 months vesting



Note: These allocations represent the current plan and may be adjusted if necessary based on evolving market conditions or strategic considerations. Any changes would be made transparently and with the community's best interest in mind.

7.3.1 Short-Term Growth Fund (STGF)

The **Short-Term Growth Fund** represents **17%** of the total $CENT supply and vests monthly over the first twenty-four months. It is a discretionary pool managed by Incentiv to support activities required to launch, operate, and scale the Incentiv ecosystem during its formative period.

Additional token may be added to the allocation, such as unsold $CENT currently allocated to the Sale category or fees paid during the first year, during which rewards are being subsidized by Community Rewards.

As the network's conditions and strategic priorities can shift quickly, the STGF is intentionally flexible, so that tokens can be deployed where they create the greatest long-term value for the network, for example:

- **Ecosystem partnerships** – underwriting integrations with wallets, exchanges, and infrastructure providers that accelerate user adoption and unlock new use cases.

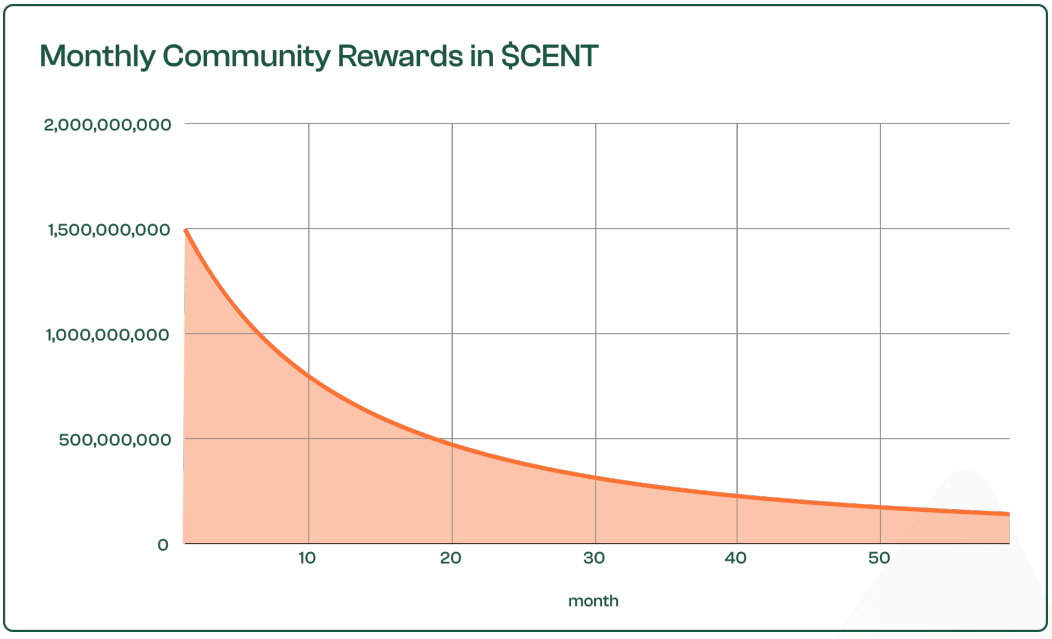
- **Developer initiatives** – funding grants, hackathons, audits, and tooling that encourage builders to ship high-impact applications on Incentiv.
- **Liquidity and market efficiency** – seeding core liquidity pools, supporting market-making programs, or subsidising early Paymaster activity to ensure low-friction onboarding and deep trading venues.
- **User-growth programmes** – financing marketing campaigns, referral incentives, educational content, and community events that expand the active user base.
- **Operational expenditure** – covering day-to-day costs such as infrastructure, operations, marketing, security and team expansion necessary to sustain the network's momentum.

The guiding principle is straightforward - **deployment of STGF resources where they generate the strongest and most immediate network effects**, whether that is attracting strategic partners, rewarding builders, deepening liquidity, or meeting operational needs.

Because the fund is time-bounded and non-replenishing, allocations are made with a clear focus on catalytic impact and measured return, ensuring that by the time the STGF is fully vested the ecosystem is sufficiently self-sustaining through organic fees and community rewards.

7.3.2 Community Rewards – Reward-Pool Pre-Load

The **Community Rewards** represents **26%** of the fixed $CENT supply. It is not a stand-alone giveaway programme but rather a **pre-load of the Incentiv reward pool**. Its sole purpose is to seed that pool with sufficient depth to pay **meaningful rewards from the very first epoch**, long before transaction-fee inflows alone could do so.



By frontloading the pool, the network can offer attractive payouts that draw miners, developers, users, liquidity providers, and bundlers to the platform while overall activity is still ramping up.

To balance early momentum with long-term sustainability, distribution follows a **hyperbolic decay curve** over roughly five years in a decreasing pace:

- Year 1 - ~49%.
- Year 2 - ~23% of the allocation.

- Year 3 - 13% of the allocation.

No Community Reward token is ever distributed outside the contribution-score system. Every $CENT from this pool is earned by measurable work—hashing blocks, processing user operations, building dApps, providing liquidity, or generating healthy user activity. Tying the pre-load to performance metrics prevents Sybil exploits and protects the allocation from opportunistic drain; if an address does not contribute real value, it earns nothing.

Role weightings can be tuned by on-chain governance as the ecosystem matures. In early epochs the Community Rewards portion dominates total payouts, giving every stakeholder class a strong financial reason to engage at once. As network usage grows and **fee inflows rise naturally**, the scheduled token injections shrink in relative and absolute terms, until rewards are funded primarily—eventually entirely—by economic activity rather than by reserves.

This pre-load strategy offers three strategic advantages:

- **Immediate Network Effects:** Rich early rewards attract hash power, liquidity, and application development on day one, making the chain useful to end-users far sooner than fee-only economics could allow.
- **Smooth Transition:** Continuous decay avoids reward cliffs and signals clear, predictable economics; participants can plan years ahead, knowing exactly how incentives will evolve.
- **Supply Integrity:** Because all Community Reward tokens circulate through the same pool that will later be fed solely by fees, the hand-off to a self-sustaining economy is automatic—no additional policy change or new mechanism is required.

In short, the Community Rewards allocation is the scaffolding that supports Incentiv's early growth. It gives the economy time to reach critical mass, after which organic fee recycling can uphold the reward cycle indefinitely, without further token issuance.

7.4 Long-Term Sustainability

$CENT's fixed-supply design supports a closed-loop reward pool that recycles network fees back to contributors without expanding the token base. While the mechanics of fee collection and distribution are detailed in the Economic Model section, the token's role in that loop is straightforward and foundational:

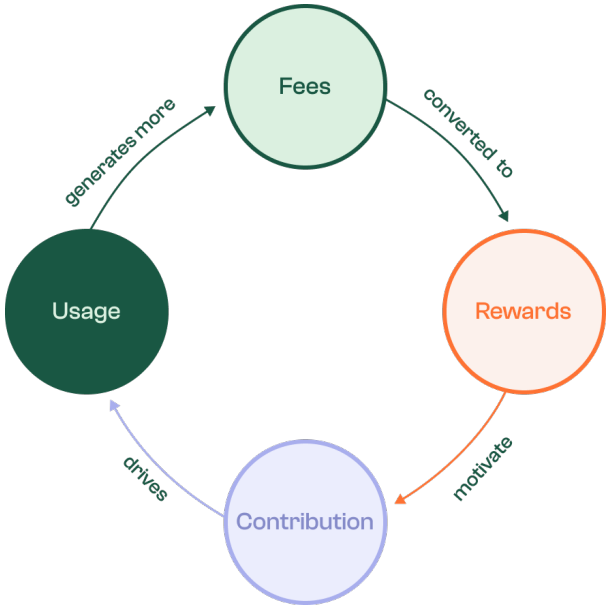
- Every fee that the protocol collects is ultimately denominated in $CENT and flows into a single reward reservoir.
- Each epoch, that reservoir pays out $CENT to miners, developers, users, liquidity providers, and bundlers according to their verified contribution scores.
- No new $CENT is minted after genesis, and none is burned by design; value simply circulates among active participants.

This architecture yields four durable advantages for token holders and network contributors:

1. **Value Preservation**: With supply capped at 100 billion, every reward represents an undiluted share of the existing pie. Contributors gain real economic weight rather than inflationary script.
2. **Predictable Economics**: Because the supply curve is flat and transparent, stakeholders—from miners sizing hardware outlays to users planning long-term positions—can model token dynamics years in advance. The only moving parts are fee volume and network usage, both observable in real time.
3. **Gradual Self-Funding Transition:** During the network's formative years, the Community Rewards pre-load supplements fee inflows so that rewards remain meaningful even at low transaction volume. As usage scales, fee revenue naturally replaces these diminishing token injections, allowing the reward pool to become entirely self-sustaining without altering monetary policy.

4. **Aligned Long-Term Incentives:** Absent perpetual inflation, the principal way to capture upside is to help the network grow—either by building, securing, or using it. Early and ongoing contributors therefore share directly in the rising value of $CENT rather than relying on short-term emissions.

In essence, the token model places $CENT at the centre of a self-reinforcing economic flywheel - fees convert to rewards, rewards motivate contribution, contribution drives usage, and usage generates more fees—all without inflating the supply.



The detailed mechanics behind that flywheel follow in the next section; here, it suffices to note that **$CENT's immutable supply and fee-recycling role provide the monetary bedrock on which the Incentiv economy is built**.

8. Economic Layer – Incentiv+ Engine

8.1 Economic Design Principles

The Incentiv+ economic engine is designed as an integral part of the blockchain rather than an add-on, ensuring that incentive mechanisms are embedded at every layer of the network. This integrated approach means that whenever participants act to maximize their own rewards, they are simultaneously strengthening the ecosystem. In essence, this **mechanism design** results in a carefully calibrated environment where rational self-interest naturally leads to outcomes that benefit the whole community. Instead of relying on ad-hoc solutions based on token allocation or inflation, Incentiv's core protocol operations inherently drive positive-sum behavior, aligning individual optimization with collective growth.

8.1.1 Unified Reward Pool Principles

At the heart of this system is a unified **reward pool** that aggregates value from all network activity and redistributes it to contributors based on their proven contributions. Rather than splitting rewards into isolated silos for miners, developers, and other stakeholders, **all** transaction fees and other economic inputs feed into a single communal pool. This eliminates the fragmentation seen in some other blockchains where different groups vie for separate rewards, and ensures that as the network grows, *all* participants share in the upside proportionally.

The reward pool is grounded on the following core principles:

- **Pool Funding Sources:** Every transaction on Incentiv contributes to the same reward reservoir. Fees charged on user operations (calculated from the economic value of transactions) are converted into $CENT and deposited into the pool. In addition, the pool is bootstrapped by an upfront allocation of tokens reserved specifically for community rewards (the "reward pool pre-load"), as well as any supplementary early subsidies from growth funds. Other sources like revenue from protocol-owned services or penalties (e.g. slashed stakes from misbehaving participants) also flow into this pool. Consolidating all value into one pot creates economies of scale in distribution and means the incentive for contributing is tied directly to overall network success, not just one's individual role.
- **Algorithmic Distribution:** Rewards from the pool are paid out on a fixed **epoch cycle** (weekly, as detailed below) according to transparent on-chain events rather than manual allocations. At the end of each epoch, scoring nodes that pick up on these events calculate every participant's share based on their contribution scores (across all roles) and sign on-chain allocations in the smart contract that releases rewards accordingly. This algorithmic approach guarantees predictability and equitability: participants can trust that rewards are determined solely by objective metrics and preset formulas, without any centralized discretion. It also allows the incentive logic to adjust to changes in participation or activity levels each epoch, keeping rewards well-calibrated to current network conditions.
- **Gradual Fee-to-Reward Transition:** Importantly, the unified pool model is built to guide the network from a **subsidy-driven** early stage to a long-term **fee-driven** economy. In the beginning, on-chain activity may be low, so the pool is largely filled by the pre-allocated Community Rewards tokens – ensuring meaningful payouts from day one. As usage increases, however, naturally collected fees will make up a larger and larger portion of the pool. The protocol is designed for a smooth handoff: early on, reserved tokens dominate rewards, but over time fee inflows organically take over without any abrupt drop-off in total rewards. This avoids the "cliff" effect that plagues some networks when initial token emissions run out. Instead, Incentiv's reward pool gradually

transitions to being self-sustaining through actual network usage, all while maintaining consistent incentive levels for participants.

8.1.2 Progressive Fee Model

Incentiv's transaction fee model is fundamentally **value-based** rather than purely resource-based. Instead of pricing solely by computational effort or gas used, the network charges fees as a small percentage of the economic value of each transaction. This means a transaction securing a high value on-chain pays slightly more, whereas a low-value transaction pays very little – making the network cost-effective for everyday use. The fee schedule uses **progressive marginal brackets** akin to a progressive tax system: different portions of a transaction's value are charged at increasing rates, so that larger transactions contribute more in fees without pricing out smaller ones. For example, a million-dollar DeFi trade might incur a higher total fee (in $CENT) than a $10 transfer, reflecting the greater value secured by the network, but both are charged in proportionate tiers so that both transactions remains affordable.

To balance network growth with economic sustainability, fee collection will be introduced gradually. **At genesis, fees start at 0%** – meaning early users pay no fees at all – in order to encourage adoption and activity. As the user base and on-chain value grow, the protocol will phase in fees over time, moving toward the target fee rates in stages rather than all at once. This phased rollout (for example, beginning with a small fraction of the full fee rates and then incrementally increasing over successive epochs) allows the community to observe how fees impact usage and adjust parameters if needed, ensuring that fee implementation does not hinder growth. It also means early adopters benefit from a period of essentially free usage, while still earning rewards from the pool – a powerful incentive to join early.

Behind the scenes, the network's **fee conversion mechanism** ensures that fees can be paid in any token yet still support $CENT and the reward pool. Users may transact and pay fees in ETH, stablecoins, or other tokens via the unified token mechanism, sponsored by the paymaster and then routed to conversion into $CENT at market rates. Thus, all fees ultimately enter the reward pool as $CENT value, creating a constant baseline demand for the token without forcing every user to acquire it in advance.

Finally, the fee model includes safeguards to prevent manipulation. **Anti-gaming measures** are in place so that users cannot evade fees or inflate their reward share through unnatural behavior. For instance, splitting one large transaction into many small ones will not avoid fees – because each transaction has a base cost, splitting into many pieces can incur even *higher* total fees. These protections ensure that fee contributions and reward calculations remain fair and rooted in genuine economic activity.

8.1.3 Weekly Epoch Cadence

All economic calculations and distributions in Incentiv follow a fixed **epoch cycle,** which will be set as a weekly epoch at launch. Each epoch spans the seven-day equivalent in number of blocks, after which the network "close" the period, compute contributions, and allocate rewards for claim before the next cycle begins. This cadence is chosen to strike a balance between feedback timeliness and data reliability. A week provides enough time to gather meaningful activity metrics and smooth out day-to-day noise, yet it is short enough to keep participants engaged with a regular reward rhythm. In other words, contributors can claim rewards frequently (every week), which reinforces active participation, yet each reward reflects a substantial period of work.

Coordinating on weekly epochs also brings **predictability and synchronization** to the system. All roles and processes – from miners producing blocks, to users making transactions, to developers deploying new code – operate within the same repeating schedule. Scores are calculated and reset on epoch boundaries, and protocol parameters or allocations can also be adjusted at these boundaries if needed. This predictable cycle creates a regular heartbeat for the network that participants can plan around, while simplifying implementation by aligning updates to a common timetable.

Equally important, the weekly reset prevents anyone from accumulating a permanent advantage. Each epoch, everyone effectively starts fresh in terms of contribution ranking, so past high performers do not carry an unfair lead into the next round. Consistency is rewarded – someone who contributes well *every* epoch will earn steady rewards –

but no participant can simply coast on a previous epoch's success indefinitely. This fosters a competitive but fair environment where newcomers always have a chance to prove themselves, and ongoing effort is required to stay at the top. Overall, the seven-day epoch structure keeps incentives timely, recurring, and aligned with natural engagement cycles.

8.2 Role-Based Contribution Scoring

Incentiv's economic engine rewards a wide range of contributors by quantitatively measuring their contributions across different roles. The network recognizes **five key roles** that together drive the ecosystem's success, and it maintains a separate contribution score for each participant in each relevant role. By segmenting contributions this way, the system ensures that, for example, developers are compared appropriately against other developers, and miners against other miners, rather than everyone competing on a single metric. Every role is vital in its own right, and the scoring framework is designed to compensate each category of work according to the specific value it brings to the network.

8.2.1 Roles and Value Contribution

The five participant roles acknowledged in the Incentiv protocol are:

- **Miners:** The blockchain's security providers. Miners commit computational power to validate blocks and process user operations. Beyond mere block production, they maintain the trustless foundation upon which all other activities are built. Without reliable miners, no other role could function effectively, so miners are rewarded for securing the network's integrity and performance.
- **Developers:** The creators of applications and smart contracts. Developers expand network utility by building dApps and services; without them, blockchain infrastructure would remain an empty shell. Successful applications drive transactions and attract users, so developers who bring valuable, widely used dApps to Incentiv directly contribute to network growth and are compensated accordingly.
- **Users:** The end-users who generate on-chain activity and network effects through their participation. Users transform potential utility into actual value by using dApps, initiating transactions, and engaging with the network. They effectively create demand for block space and services. Active users are thus rewarded for contributing transaction volume and engagement that make the ecosystem vibrant.
- **Bundlers / Wallet Providers:** Specialized actors (enabled by account abstraction) who package user operations and provide streamlined transaction services. Bundlers optimize how transactions are submitted to the chain and often cover gas fees, while wallet providers/clients improve UX on behalf of users. In doing so they increase the network's efficiency and accessibility. They earn rewards for adding this operational value and helping more user transactions get on-chain smoothly.
- **Liquidity Providers (LPs):** Participants who supply liquidity to decentralized exchanges and other market mechanisms within the ecosystem. By locking up assets in liquidity pools, LPs enable efficient token swaps and price discovery. This role is crucial for $CENT's economy, as deep liquidity ensures that the fee conversion (from various tokens into $CENT) is frictionless and that markets remain stable. LPs are rewarded for committing capital and facilitating the economic infrastructure that other participants rely on.

Each of these roles addresses a different dimension of network health, and **all are needed** for Incentiv to thrive. The scoring and reward system therefore allocates rewards to each role separately (using governance-set percentages) and then further divides those rewards among individuals in the role based on their personal contributions. This way, a high-performing miner isn't directly competing with a high-performing developer for the same reward pool, for

example – each is rewarded from their own role's allocation, reflecting the distinct value they bring. Participants can fulfill more than one role.

8.2.2 Multi-Metric Score Composition

To measure performance in each role, the contribution scoring system evaluates multiple metrics rather than relying on any single number. Every role has a tailored set of quantitative indicators that reflect meaningful contributions for that category of work, and each participant's score is a weighted blend of those metrics. Notably, one common factor across **all roles** is the participant's stake of $CENT tokens, which serves as a fundamental alignment measure. Holding $CENT demonstrates long-term commitment to the ecosystem, so the scoring formulas give a modest boost to contributors who hold the native token (and may, in parallel, use it to provide liquidity in parallel) . This boost is calibrated such that holding tokens complements active contribution but does not replace the need to actually participate. For four of the five roles, only $CENT holdings count toward this alignment factor; Liquidity Providers are the exception in that **any** liquidity they provide (even in non-$CENT tokens) is counted as part of their contribution – recognizing the value of all assets they put at the service of the network.

Beyond token alignment, each role's score includes role-specific performance metrics. For example:

- **Miners:** Measured by factors such as blocks and transactions processed (e.g. a miner's share of total block production, including any uncle blocks they help include) A miner's $CENT holdings in their coinbase address reflect their vested interest in the network's success. Additional miner metrics like uptime and availability or contributed hashpower relative to the network could be incorporated as the infrastructure allows.
- **Developers:** Measured by the usage and impact of their deployed applications. Key metrics include the number of transactions flowing through the developer's smart contracts, the number of unique active users engaging with their dApps each epoch, and the total volume of value those contracts handle. These indicators reflect how much real utility and adoption a developer has driven. The developer's own $CENT stake is also included, indicating their alignment with the platform's long-term growth.
- **Users:** Measured by on-chain activity and engagement. Important metrics for users are the frequency of transactions (regular participation is valued over sporadic bursts), the volume transacted, and the fees contributed to the protocol. Additionally, interacting with a diverse range of dApps (as opposed to only one application) can boost a user's score, since it indicates support for the broader ecosystem (this metric is also used as an anti-gaming measure). Holding $CENT over time (instead of immediately selling all rewards) is another positive signal of a user's commitment.
- **Bundlers:** Measured by service efficiency and reliability. Metrics include the number of UserOperations successfully processed, the gas efficiency achieved (i.e. how much they optimize gas usage for the network), and their success rate in getting operations included in blocks. Handling a variety of operation types or serving many different dApps can also improve a bundler's score, as it shows versatility. As with other roles, any $CENT held by the bundler contributes to their score as an alignment indicator.
- **Liquidity Providers:** Measured by the liquidity and trading activity they enable. Key metrics are the total trading volume facilitated by the liquidity an LP supplies and the number of transactions (trades) that occur through their pools. The duration for which liquidity is locked or continuously provided is also considered – an LP who remains in a pool for a longer term (indicating commitment) can score higher than one who adds and removes liquidity frequently. The total capital deployed in the pools (depth of liquidity) is of course a major factor, as larger pools provide more value to the network. (LPs inherently get credit for non-$CENT tokens they stake in pools, but their $CENT holdings are counted too if they provide $CENT liquidity or hold rewards.)

By considering a combination of metrics, the system paints a well-rounded picture of each participant's contribution. It prevents any single factor from dominating – for example, a user who generates huge transaction volume but only in one short burst might score lower than another user with slightly less volume spread out steadily across many apps. The exact weight of each metric is initially set based on modeling, and can be adjusted via governance as needed to fine-tune incentives.

8.2.3 Score Normalization and Equitability

Raw performance metrics are processed through a normalization pipeline to ensure equitability and comparability across participants. First, **data collection** is automatic and trustless: whenever relevant events occur on-chain (transactions, contract calls, blocks mined, etc.), they emit structured events that are captured by specialized scoring nodes. Thus, scores are computed from verifiable on-chain data without requiring any extra effort or off-chain reporting from participants.

Within each role category (miners, developers, users, bundlers, liquidity providers), the normalization pipeline follows a consistent approach:

- The top contributor establishes the maximum reference point normalized (100)
- The minimum contribution is always set to zero (0)
- All other participants are plotted on this scale according to their relative performance

This normalization ensures that participants are evaluated proportionally within their role cohort, regardless of the absolute values or units of the underlying metrics. **The normalized scores establish a consistent basis for applying the gradient distribution formula during reward calculation.**

Multiple independent scoring nodes (minimum 5-10) process these metrics in parallel, providing cross-validation to ensure accuracy and prevent manipulation. While preliminary scoring occurs in real-time throughout the epoch, allowing participants to track their approximate standing, final score calculation is determined at the epoch boundary after cross-validation is complete.

The epoch-based evaluation (approximately 7 days or ~40,320 blocks) creates natural temporal boundaries, ensuring that rewards reflect current contributions rather than historical performance. This encourages continuous effort - participants who increase their activity or improve their service quality will see their normalized scores rise within the current epoch.

Crucially, the system incorporates anti-gaming mechanisms to detect and penalize attempts at exploiting the scoring mechanism. This includes identifying Sybil attacks (e.g., splitting one entity's activity across multiple fake identities to earn multiple rewards) and flagging anomalous patterns that suggest artificial inflation of metrics (such as ping-pong transactions that generate volume without real value transfer). When such behavior is detected, the offending metrics can be discounted or the participant can be disqualified for that epoch. The guiding principle is that only genuine, value-adding contributions should count toward rewards. By combining robust normalization with these safeguards, Incentiv's scoring system maintains a level playing field and directs rewards to those who truly help the network.

8.3 Linear Gradient Reward Distribution

Having quantified each participant's contributions via scores, Incentiv allocates the weekly reward pool using a **continuous gradient model** rather than winner-takes-all or strict tiers. Traditional blockchain reward schemes often create hard cut-offs – for instance, only the top N contributors get a reward and everyone else gets nothing. Such systems can discourage those who just miss the cut, or encourage unhealthy manipulation of the metrics around threshold levels. Incentiv avoids these pitfalls by ensuring **every legitimate contributor receives a reward proportional to their contribution**, with higher performers earning more but lower performers still getting something for their efforts.

8.3.1 Continuous Reward Model

In the gradient model, rewards scale smoothly with performance. There are no arbitrary tiers or "all-or-nothing" jumps. If one participant's contribution score is, say, twice as high as another's, they will receive roughly twice the reward (subject to certain caps or minimums set by governance), whereas in a traditional model the lower contributor might get nothing at all. This creates a natural incentive for everyone to improve their contribution without the fear of falling off a cliff edge in the payout structure. Even the lowest-ranked active contributors earn a meaningful share, which keeps them engaged and striving to earn more next time. Meanwhile, the highest-ranked contributors cannot rest easy because their reward advantage is only proportional – if others below them work harder and improve, the top performers must also continue to excel to maintain their relative share.

From a behavioral perspective, this continuous model is more motivating and sustainable. Research in behavioral economics indicates that all-or-nothing reward structures can demotivate those in the middle or bottom, whereas a graded reward structure gives *everyone* a reason to push a bit further. Middle-tier participants see clear, incremental progress – every improvement in their contribution yields a tangible increase in rewards – and top performers remain incentivized to innovate rather than becoming complacent. Moreover, motivation does not follow a strictly linear relationship with reward size; moderate, continuous incentive structures often produce more sustainable performance than extreme winner-take-all schemes. Incentiv's approach creates a dynamic yet inclusive ecosystem with constant encouragement for innovation and improvement at all levels.

This model also maximizes network growth and participation. By rewarding all genuine contributors (even smaller ones), the system harnesses network effects: many participants adding small amounts of value can collectively have a significant impact, and none of that effort is excluded from rewards. No contributor is made to feel that their input is too minor to matter. In summary, the gradient reward model fosters healthy competition without discouraging the majority of contributors, leading to a more robust and cooperative growth trajectory for the network.

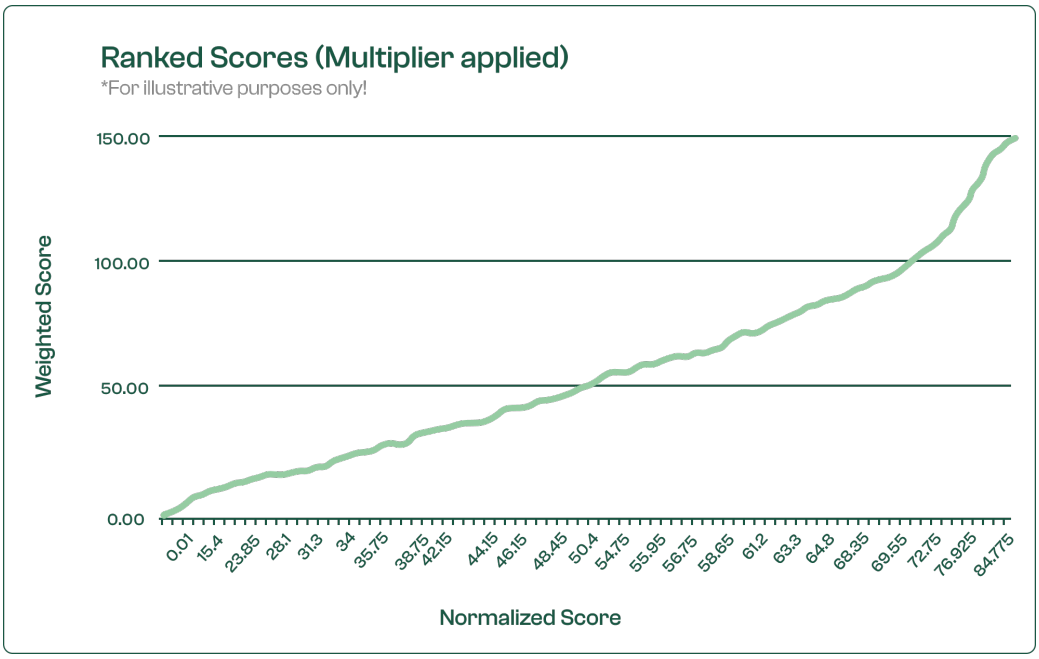
8.3.2 Distribution Mathematics

Under the hood, the gradient distribution follows a simple, transparent mathematical formulation to ensure proportionality and predictability. First, for each epoch the total reward pool (composed of fees and any token subsidies for that week) is divided among the roles according to preset **role allocation percentages**. These allocations – e.g. X% of the pool to miners, Y% to developers, etc. – are set via governance and can be adjusted within reasonable bounds as the community deems appropriate. This step ensures that the reward split aligns with the network's priorities (for example, if liquidity becomes more crucial, the LP allocation could be increased, and so on).

Next comes the distribution within each role's share. Incentiv uses a **rank-based multiplier** approach to slightly favor the top performers in a role while still giving every participant a portion of the rewards. Each contributor in a role is assigned a multiplier based on their rank (position by contribution score in that role for the epoch). The formula for the multiplier is a linear interpolation between a set maximum and minimum value:

```
Multiplier(rank) = Max - (Max - Min) × (rank - 1) / (n - 1)
```

Where *n* is the number of contributors in that role for the epoch, *rank = 1* for the highest scorer, and *rank = n* for the lowest. For example, if the top multiplier Max is 1.5× and the bottom Min is 0.5×, then the best performer gets a 1.5× boost to their score, the last person gets 0.5×, and others get a multiplier in between proportional to their ranking. This linear gradient ensures a smooth gradation of rewards from top to bottom without sharp drop-offs.



The specific Max and Min multipliers can be tuned per role via governance (within limits – for instance, top multipliers might range from 1.2× up to 2.0×, bottoms from 0.3× up to 0.8×) so the community can decide how egalitarian vs. top-heavy the reward curve should be.

After applying the rank multipliers, each participant has an **adjusted (weighted) contribution score** for the epoch:

```
Weighted_Score = Raw_Contribution_Score × Multiplier(rank)
```

Finally, the rewards for each individual are calculated as their proportion of the total weighted score in their role, times that role's reward pool:

```
Individual_Reward = Role_Pool × (Individual_Weighted_Score /
Sum_of_All_Weighted_Scores_in_Role)
```

This means that within a given role, participants are essentially sharing the role's reward pool in proportion to their weighted contributions. If one miner's weighted score is 10% of the sum of all miners' weighted scores, that miner gets 10% of the miners' allocated rewards for the epoch. Because of the weighting, a top-ranked contributor's score is amplified relative to a lower-ranked one (per the linear multiplier above), giving the top performer a larger slice of the pie than the raw scores alone would dictate – but every active contributor still gets *some* slice.

This two-step allocation – first by role, then by relative contribution within the role – ensures that the distribution is both **role-proportional** and **performance-proportional**. Role-proportional means each category of work (security, development, usage, etc.) receives its intended share of the rewards, and performance-proportional means that within each category the payouts scale smoothly according to how much each participant contributed. The use of a linear gradient and transparent formulas provides predictability: contributors can estimate how improvements in their score or rank will affect their rewards, and there are no sudden jumps or cut-offs to worry about. All calculations are executed automatically by the protocol's smart contracts at epoch's end, making the process trustless and free of human bias.

8.4 EIP-1973 Hybrid Distribution System

While the prior sections cover how reward amounts are determined, the **distribution mechanism** addresses how those rewards actually reach participants' hands. Simply transferring tokens to tens of thousands of addresses each epoch would be impractical and costly on-chain. To solve this at scale, Incentiv employs a **claim-based payout system** inspired by Ethereum Improvement Proposal 1973 ("Scalable Rewards")[9]. Instead of pushing rewards out to everyone (which doesn't scale well), the network creates a cryptographic proof of each participant's reward and allows individuals to **claim** their $CENT when convenient. This approach drastically improves scalability and flexibility, ensuring that reward distribution remains efficient even as the contributor base grows.

8.4.1 Scalable Reward Distribution

At the close of each weekly epoch, the network produces an immutable record of all rewards earned. Specifically, the set of scoring nodes that computed the results will submit a **Merkle root** (a cryptographic commitment) to the EIP-1973 distributor contract on-chain, which encapsulates the reward amounts for every eligible address. This single hash represents the entire reward ledger for the epoch. The detailed breakdown (which address earned how much) is available off-chain in the form of a Merkle tree data structure, but it does not need to be posted on-chain in full.

With this mechanism, any participant can later claim their rewards by presenting a **Merkle proof** to the distributor contract. The proof is a short piece of data that verifies that a particular address and reward amount were included in the Merkle root for that epoch. The distributor contract, given the root and the proof, can efficiently confirm the claim without needing to store or loop through all entries. This "pull" model shifts the gas cost to individual claim transactions (initiated by each participant at a time of their choosing), rather than having the protocol pay the cost of distributing to everyone. It also means participants can bundle multiple epochs together: for example, if someone waits 4 epochs before claiming, they can combine the proofs for those 4 epochs and make one transaction to claim all foat once, saving on fees. Unclaimed rewards simply accumulate as claimable balances for 12 epochs – there is no loss in waiting a reasonable period, and it does not inflate the token supply or harm the system if people defer their claims.

This approach is highly **scalable and decentralized**. It avoids block gas limit issues that would arise from trying to pay out thousands of addresses in one transaction, and it does not rely on any central party to distribute rewards – each participant triggers their own payout when it suits them. The cryptographic structure (Merkle root and proofs) ensures that the process remains trustless: anyone can independently verify the entire reward distribution off-chain and ensure the on-chain root is correct. The result is a robust distribution method that can handle a large number of participants without sacrificing security or efficiency.

8.4.2 Flexible Claiming Options

The distribution contract offers several claim methods to accommodate different participant preferences and to encourage long-term alignment. When claiming their accumulated rewards, participants have a choice in *how* to receive them:

- **Immediate Liquid Claim (with Discount):** A participant can claim their rewards instantly in liquid $CENT, but only receive a portion (for example, 50%) of the earned amount. The remaining portion that is "discounted" or forfeited isn't kept by any third party – instead, it is returned to the communal reward pool for future epochs. This option provides immediate liquidity to those who need it, as the participant immediately receives tokens that can be freely traded or used. However, this comes at a significant cost, effectively burning half of the value by returning it to the communal reward pool. It functions as a voluntary contribution back to the community, benefiting those who do not cash out immediately since the forfeited value is redistributed via the pool.
- **Direct Utility Claim (Paymaster Bonding):** A participant can claim **100% of their rewards** directly into their personal paymaster account. In this mode, the full amount of $CENT is granted, but it is locked for use on the

network itself – specifically, it can only be spent on paying transaction fees (gas fees or value fees) via the paymaster system. This effectively pre-pays the participant's future activity on the network. It is beneficial for participants who intend to remain active on the network, since it fully covers their transaction costs by providing the entire reward amount for future fees, while ensuring the value continues to circulate within the ecosystem rather than being withdrawn.

- **Vested Claim:** As a third option, a participant may claim the entire **100% of their rewards** with no immediate discount, but the tokens will vest over a predefined schedule (just as an example example, linearly over 3 months). The claimed $CENT is locked in the distributor contract (or a dedicated vesting contract) and gradually released to the participant over the vesting period. This option rewards long-term thinking: the participant eventually gets all their tokens, but by delaying full access they demonstrate commitment and help reduce sales pressure. This option is essentially equivalent to holding the tokens under a time lock, aligning the participant's interests with the network's long-term success over the vesting period.
- **Hybrid Splits:** The claim system also supports splitting a reward into multiple modes. For instance, a developer might immediately take 30% in liquid $CENT for operating expenses and vest the remaining 70% for the long term. Or a miner might do a 50/50 split between an immediate claim and a direct-to-paymaster claim to cover upcoming costs. This flexibility lets each participant customize how they realize their rewards based on their needs and confidence in the network.

These options are carefully designed to balance individual liquidity needs with the network's incentive to keep participants invested. Those who need or prefer immediate liquidity can have it, but the protocol imposes a trade-off (surrendering a portion of the reward) that in turn bolsters the collective reward pool. Meanwhile, those who keep their rewards in the ecosystem – either as locked fee credits or vesting tokens – are effectively rewarded by getting the full amount. It creates a self-regulating system where short-term exits fund the long-term believers, and everyone has the freedom to choose the option that suits them best.

8.4.3 Anti-Gaming and Security

The reward distribution contract incorporates multiple layers of security to ensure that only legitimate rewards are claimed and that no one can exploit the system:

- **Merkle Proof Verification:** A claim is only successful if the provided Merkle proof matches an entry in the official reward root and passes verification by the contract. Moreover, the protocol requires a supermajority of scoring nodes (e.g. 7 out of 10) to sign off on each epoch's Merkle root attestation. This makes it virtually impossible for an incorrect or falsified root to be accepted for payouts, since it would require collusion by a large portion of the network's trusted (staked) actors. Any attempt to claim a reward that isn't actually earned (i.e. not in the Merkle tree) will fail because the cryptographic check will not pass.
- **No Double-Claiming:** The contract tracks which epochs each address has claimed rewards for, preventing any duplicate claims. Once an address has claimed its rewards for a given epoch, any further claim for that same epoch by the same address is automatically rejected. This simple measure protects against replay attacks or mistakes where someone might try to claim twice.
- **Slashing Integration:** If a participant behaves maliciously after earning rewards but before claiming them, the system can revoke those pending rewards in extreme cases. For example, suppose a miner earned rewards in epoch N but then was proven to have attacked the network in epoch N+1 before claiming epoch N's rewards. The governance or an automated slashing module could mark that miner's reward entry as void, preventing them from claiming what they earned prior to the misconduct. This ensures that participants cannot profit

from actions the network later deems harmful or disqualifying. It ties into the broader security model: substantial misbehavior can lead to the loss of not just future earning potential but also any unclaimed past rewards if the protocol rules warrant it.

- **Claim Time Limits:** To avoid indefinite accumulation or "zombie" claims lingering forever, rewards must be claimed within a certain period (for instance, 12 epochs after the end of the relevant the epoch). If they are not, those unclaimed tokens are returned to the communal reward pool to be recycled into new rewards. Since $CENT has a fixed supply and no inflation, recycling unclaimed tokens is important to keep the reward cycle robust over the long term. The time limit is set generously to give participants plenty of opportunity to claim, but it protects against a scenario where, say, many tokens remain unclaimed for years (which could complicate token economics or governance). It also provides a cleanup mechanism so that the distributor contract does not have to track claims indefinitely.

These security measures ensure the reward claiming process is reliable and trust-minimized. Participants can be confident that if they earned rewards, those tokens will be there for them to claim securely. Likewise, the community can be confident that the payout system will not become a vector for fraud or unintended token release. Everything is rooted in cryptographic verification and transparent rules, mirroring the integrity of the Incentiv+ Engine as a whole.

8.5 Transition to Self-Sustaining Economics

A core objective of Incentiv's economic model is to smoothly transition from an initial, incentive-boosted launch phase to a mature state where the network runs entirely on organic fee revenue. This is achieved through a phased approach that coordinates the tapering of token subsidies with the ramp-up of fee collection. The **Community Rewards** allocation (26% of the $CENT supply pre-loaded into the reward pool vesting over ~5 years) is the engine of early growth – it provides generous rewards in the beginning when fees alone would be insufficient. Over time, as the user base and transaction volume grow, fee recycling naturally replaces these token injections. The transition is designed to avoid any sharp drop in rewards by decaying the subsidy in a predictable curve while proportionally increasing fee contributions. The phases of this evolution can be outlined as follows:

1. **Genesis (Launch):** Transaction fees are **initially set to 0%**, so using the network carries no cost at the start, maximizing user onboarding. In this phase, rewards for all roles are almost entirely funded by pre-allocated token reserves – primarily the Community Rewards pool (with its hyperbolic release schedule) – supplemented by the Short-Term Growth Fund if needed. Thanks to this substantial subsidy, participants earn meaningful $CENT rewards from the very first epoch, bootstrapping miner hashpower, developer engagement, user activity, and liquidity provision long before natural fee income alone could support them. In fact, the reward pool is intentionally front-loaded: roughly **half** of the Community Rewards allocation is scheduled to be distributed in the first year of operation, ensuring early adopters capture a large share of the incentive pie when network activity is at its infancy. This creates a strong early pull to attract contributors and establish network effects quickly.
2. **Early Adoption:** As the network finds its footing, the protocol begins a **modest introduction of fees** – for example, enabling ~20% of the eventual target fee rates. This means users still enjoy heavily subsidized usage (an 80% fee discount relative to the long-term rate), and the reward pool starts to receive a trickle of real fee revenue while remaining mostly supported by token subsidies. The majority of rewards in this stage still come from the decaying Community Rewards reserve, maintaining high payouts to solidify growth. The partial fee introduction serves to gradually acclimate users to the fee model and to gather data on transaction patterns and elasticity, all while **retaining low economic friction** during this critical expansion period. Short-Term Growth Fund resources may also be used tactically here to boost certain activities, but overall the system is slowly shifting some weight onto actual network value generation.

3. **Bootstrap Growth:** With increasing usage and a more robust ecosystem, fee rates are raised to around **50% of their target levels** (half of full fees). By this stage, on-chain activity is contributing a significant share of the reward pool – perhaps 30–50% of rewards are now funded by fees from real user transactions. The remaining share continues to come from the Community Rewards allocation, which is by now in the middle of its vesting period (the rate of token release is slowing each year, but still material). This phase represents a healthy balance: the network has demonstrated enough utility that it can begin paying its contributors out of its *own* revenue to a meaningful extent, yet the token reserves are still there to ensure total rewards stay high and competitive. Participants thus see that the model is working – rewards are increasingly backed by actual economic activity – but they also are still benefiting from token boosts that guarantee profitability and upside for engaging with the network. Any fine-tuning of fee parameters or role allocations can be done in this period based on the empirical outcomes observed so far.
4. **Expansion and Maturation:** As the network approaches maturity, the fee schedule moves toward **full implementation** (e.g. 80%–100% of target fees, as determined by governance). By now the usage of the chain is robust, and fees constitute the majority of the reward pool's inflows (on the order of 70% or more of weekly rewards coming from fees rather than token reserves). The Community Rewards injections have substantially tapered off — by year 4 or 5, the amount of tokens entering the pool each epoch from the reserve is much smaller than it was initially, both in absolute terms and relative to fee revenue. At this stage, the network is largely **self-sustaining**: the on-chain economy (users paying for transactions and services) is feeding the reward mechanism. Any remaining reserved tokens play a minor supplemental role or are held for strategic purposes, and the Short-Term Growth Fund will likely be exhausted or saved for exceptional interventions by this point. The economic system has effectively phased out subsidies without experiencing any shock from their removal – achieving a smooth transition to a sustainable state.
5. **Steady-State (Fully Fee-Driven):** Finally, the network reaches a point where **nearly 100% of rewards are funded by transaction fees and other organic protocol revenues**. The Community Rewards pool will have been entirely distributed (or any remaining vestiges are negligible), marking the end of reserved token subsidies. All $CENT payouts now come from value created on-chain in real time. The token supply is *fixed*, and by design, all tokens allocated for incentives are already in circulation by the time the network is mature – from here on, the system operates in a closed loop. Contributors' income is directly linked to network usage: if activity and fees grow, rewards grow; if they shrink, rewards adjust accordingly. Governance can still adjust parameters like fee rates or allocation weights as needed to optimize economics, but there is no reliance on any "war chest" of tokens to prop up participation. Incentiv's economy at this stage becomes essentially self-sustaining, with transaction fees circulating value to the contributors who keep the system thriving.

Throughout these phases, the changeover is **continuous and predictable** – there are no sudden cliffs where rewards drop off overnight. The hyperbolic decay of the Community Rewards allocation ensures a gradually decreasing token injection that smoothly hands off responsibility to fee revenue. Participants can see the trajectory well in advance and plan for it, knowing for example each epoch or each year how the mix of token subsidy vs. fee revenue is expected to evolve. This design motivates early participation (when token rewards are highest) but also reassures those in for the long term that the incentives will not disappear; they will simply come increasingly from the network's natural usage. Early contributors are thus richly rewarded – not only do they receive outsized token distributions in the beginning, but they also help bootstrap a network that will continue to reward them (via fee recycling boosting $CENT's value and ongoing usage-based rewards) for the long haul.

By the time the token subsidies fully phase out, the network should have achieved critical mass – with enough users, applications, and transaction volume – such that **fee recycling alone can uphold the reward cycle indefinitely**. In essence, the Incentiv+ Engine is designed to pay meaningful rewards from day one *and* to maintain those rewards perpetually in a sustainable way. The early token allocations act as scaffolding to help build the network's economy. Once the structure is self-supporting, that scaffolding is removed. Because no new tokens are ever minted beyond the

genesis supply, all participants' stakes in $CENT are never inflated away; instead, their value is tied to the real growth of network usage.

This creates a powerful alignment of incentives: continued prosperity for all participants depends on ensuring the platform is truly useful and attracts increasing usage, which in turn generates the fees that sustain the system. **The interests of miners, developers, users, and all other contributors are united in growing a healthy, fee-driven ecosystem**. The result is an economic flywheel where contribution drives usage, usage funds rewards, and rewards motivate further contribution – a self-reinforcing cycle built to deliver long-term value to the entire community.

Incentiv

References

1. Galbraith, J. K. (1958). The affluent society. Boston, MA: Houghton Mifflin.
2. Daly, H. E. (1996). Beyond growth: The economics of sustainable development. Beacon Press.
3. Raworth, K. (2017). Doughnut economics: Seven ways to think like a 21st-century economist. Random House Business Books
4. The Tragedy of the Commons (Garrett Hardin)
5. Nash, J. (1950) Equilibrium Points in n-Person Games. Proceedings of the National Academy of Science, 36, 48-49.
6. Peter D. Taylor, Leo B. Jonker (1978). An Analysis of Decision Under Risk
7. Kahneman & Tversky, (1980). Causal Schemas in Judgments Under Uncertainty
8. ERC-4337 Documentation
9. Scalable Rewards