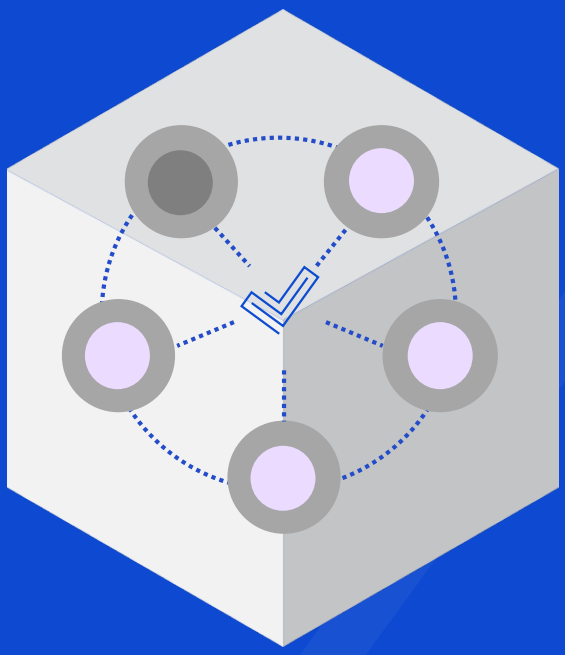
Lucidity

Decentralized Ad Analytics for the Enforcement of Digital Measurement Standards

JULY 5, 2018 - V0.5.16



ALEXANDER VOLOSHKO
MIGUEL MORALES

TABLE OF CONTENTS

TABLE OF CONTENTS

ABSTRACT

In this paper we present an implementation of a trustless system of measurement and enforcement of advertising metrics. Specifically, our implementation uses a sidechain composed of a decentralized consortium of verifiers.

This implementation provides a decentralized and democratically governed mechanism for the codification of measurement standards. This implementation enforces the standards for computing advertising metrics based on signals received from disconnected programmatic supply chain participants.

A simple standard could be codified that enforces the computation of an attribution.

It also enables a mechanism in which supply-chain participants may be paid using cryptocurrency safely and at scale.

Thus, the system we describe implements the trifecta of blockchain-based tracking and billing of programmatic advertising: campaign insertion orders, supply-chain transparency, and payments.

In this paper, we will describe the management of participant identities, a Plasma based sidechain architecture, and support for payments. We also describe a Proof-of-Stake consensus along with a rewards and penalties system used to perpetually incentivize and enforce the correct function of the network.

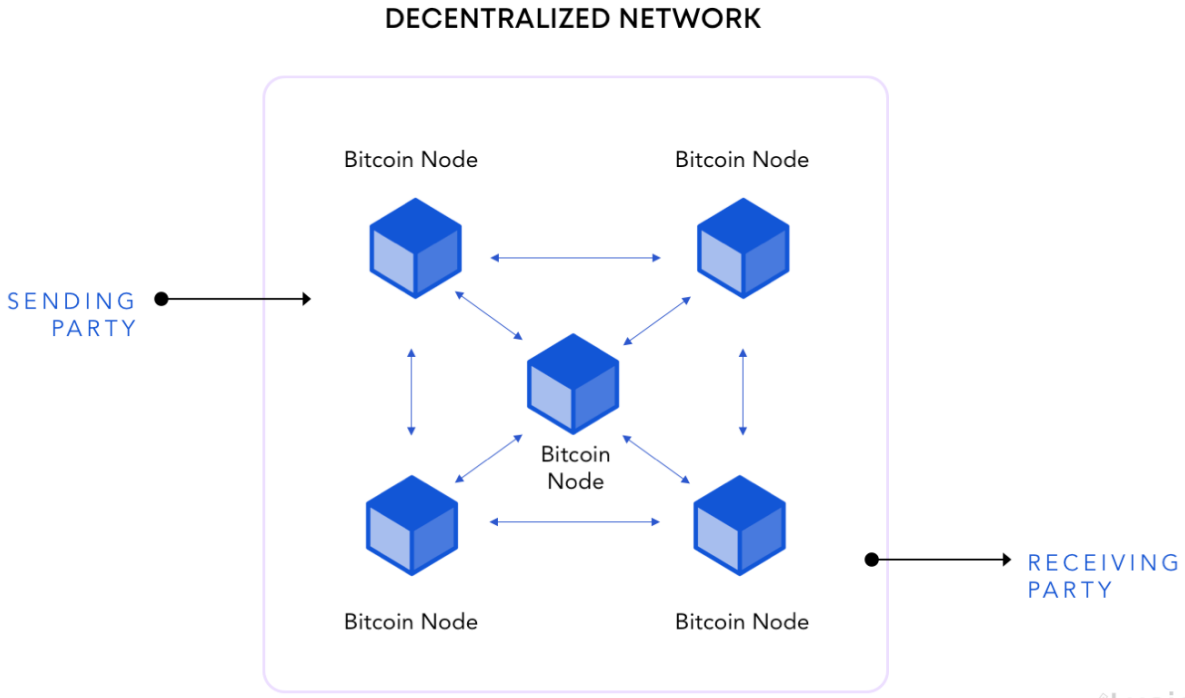


Figure : How digital currencies work

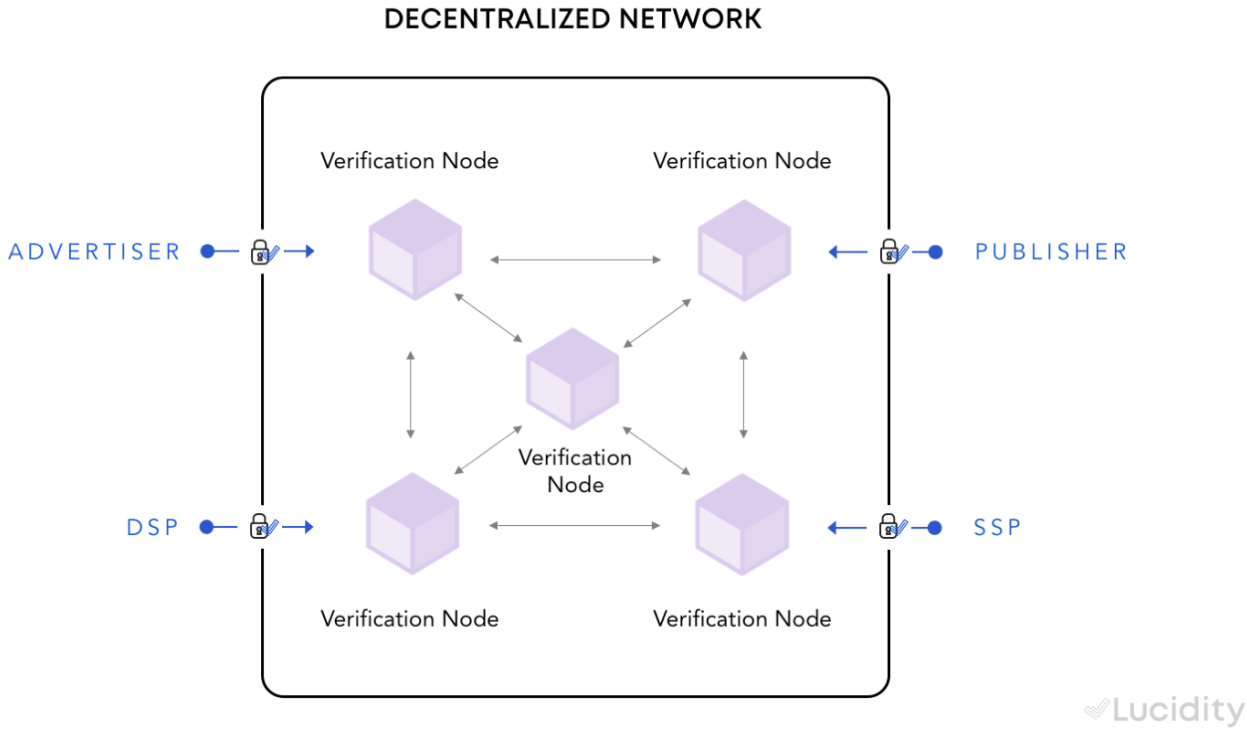


Figure: Advertising trust through decentralization

1. INTRODUCTION

1.1. Problem description

Current digital ad measurement standards are difficult to enforce and certify. Becoming certified involves the process of being audited by a certification vendor. This process makes the implementation and upgrading of measurement standards difficult - with many companies choosing to forego such certifications altogether. There are a couple reasons for this, including:

1. No sample implementation exists, so every company must re-implement every standard. This is inefficient as each standard must be implemented at least once for each implementing company.
2. When there is desire by the community to update a standard, there is a long waiting period as companies update their implementation and get re-certified. The implementation period varies depending on the urgency of the community to accept and implement the proposed changes.

Because companies independently track and compute advertising measurements, this creates discrepancies that are costly to audit and fix.

1.2. Shared ledger for advertising events

In order to implement a system in which digital measurement standards can be enforced, we create a shared ledger that takes signal events from supply chain participants to compute standardized metrics.

Currently, each party in the programmatic supply chain tracks and computes metrics differently, using their own algorithms. This leads to data discrepancy issues and limited visibility into the programmatic supply-chain. No mechanism currently exists for these unconnected parties to effectively communicate with each other, making trust difficult to establish.

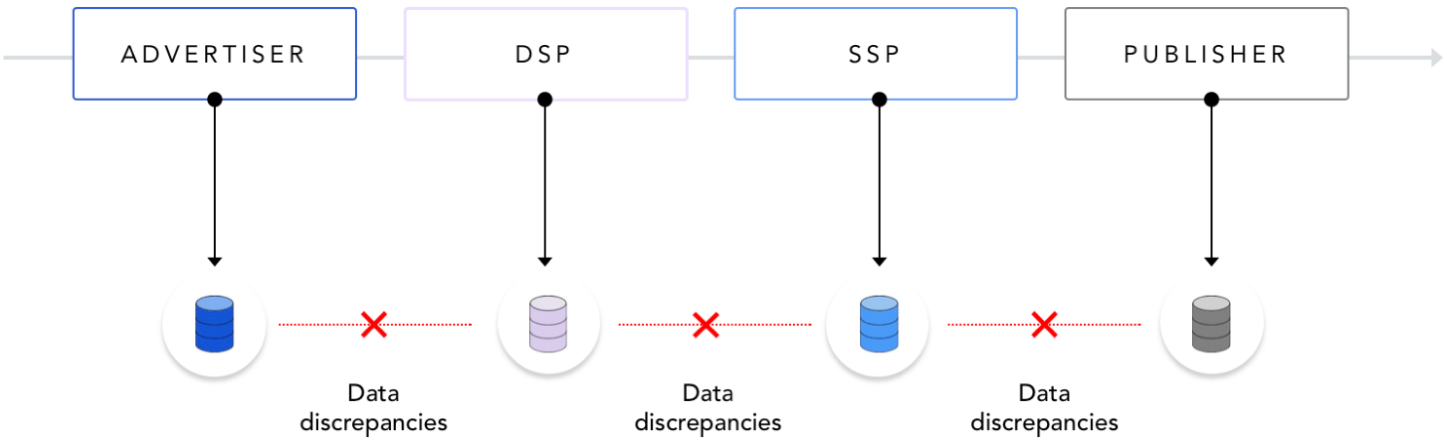


Figure: Simplified viewed of disconnected programmatic supply-chain participants

1.3. Guaranteed fairness through decentralization

The reason that no shared ledger currently exists is because supply-chain participants have no incentive to share their data. Furthermore, participants do not trust their data with centralized parties. With centralized shared ledgers, all the power of billing and discrepancy reconciliation is given to those centralized parties.

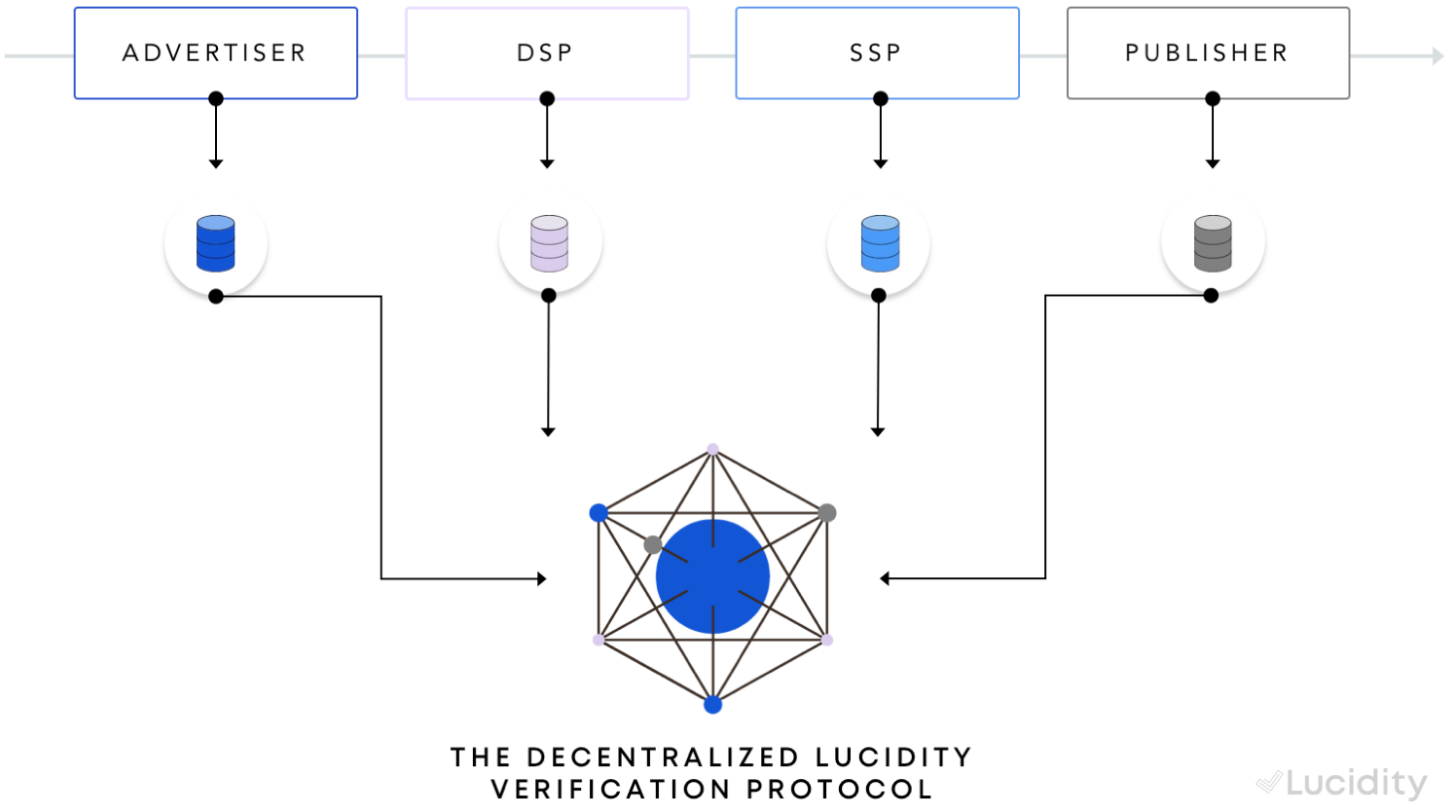


Figure: Shared ledger for enforced computation of multi-party advertising metrics

1.4. Challenges

With the invention of blockchain technology and smart contracts, we can construct a trustless and decentralized system for the computation of advertisement metrics. The system we propose uses a set of on-chain smart contracts for scalable Proof-of-Stake consensus that current systems - such as systems that implement practical byzantine fault tolerance (PBFT) - cannot provide.

Systems that implement PBFT are not well suited for programmatic advertising. As events enter the system and as the number of verifiers increases, the performance of the network greatly decreases. Furthermore, PBFT has a limit of about 100,000 transactions per second, which is insufficient to satisfy the goals of performing event-level measurement enforcement of programmatic advertising events.

1.5. Ethereum

Ethereum allows us to implement Proof-of-Stake which is tied to an economically incentivized set of tokens. Proof-of-Stake consensus, as described in this paper, allows us to process billions of events per second using a gas-efficient set of smart contracts. We also utilize tokenized incentives and penalties to enforce a self-sustaining system with real economic consequences for malicious actors. This is important because, as bluntly stated by Joseph Poon, "if you have a blockchain that's not enforceable, you're just roleplaying.".

1.6. Technology differentiator

In this paper, we describe an event-level decentralized system for the computation of advertisement analytics. We do not rely on client-side aggregation, which we believe to be a form of centralization. Additionally, we propose centralized aggregation cannot properly guarantee the computation of session-level digital measurements. Session-level metrics are those that are computed based on event-level logs.

1.7. Game theory

We also create an incentive-based game for advertisers, DSPs, exchanges, publishers, etc. In the game, advertisers have the money and want supply-chain transparency. Participants down the supply-chain wish to get paid and prove that they provide real value to the supply chain. By incorporating payments into the system, we can enforce payments down the supply chain and the sharing of truthful data. Because the system is decentralized, yet private, no central party may break the rules to their favor.

2. TOKENS

2.1. Payment vs. Staking Token

Herein, we introduce two types of tokens: a payment token and a staking token. Both tokens are standard ERC20 tokens deployed to the Ethereum blockchain.

The payment token described here is Dai. Dai is an existing ERC20 stablecoin pegged to the US dollar. A stable coin is chosen because it doesn't exhibit the same volatility that traditional cryptocurrencies exhibit, this makes it more appealing as a payment mechanism.

Lucidity is also introducing a staking token known as the Marketing Analytics Token (MAT). MAT is a staking, consensus, and governance token whose value reflects the work being done by the decentralized network of verifiers.

A protocol specific token is necessary to create the proper incentives for verifiers to operate the network. If the staking token were Ether, there would be no correlation between the price of Ether and the work being done by the decentralized verification system. Therefore, an independent, protocol specific token is necessary.

3. THE REGISTRIES

3.1. Registry Smart Contracts

The registries are a set of smart contracts with information about each entity in the registry. Every time an entity is added or updated through the smart contract, a log message is emitted which is listened to by verifiers.

This allows verifiers to cache information from the registries and only refresh their cache when a change has been made to the registry.

3.2. The Participant Registry

The participant registry holds all the entities who are participating in the sidechain. This includes: advertisers, Demand Side Platforms (DSPs), exchanges, Supply Side Platforms (SSPs), measurement/anti-fraud third parties, Data Management Platforms (DMPs), publishers, and other technology vendors in the digital advertising supply chain. Each participant also specifies a public key which can be used to validate any data the participant sends to the sidechain.

3.3. The Verifier Registry

The verifiers are stored in a registry in the form of a smart contract. Verifiers are whitelisted by the chain operator. Using the verifier registry, verifiers are able to associate an IP address to their verifier address. The verifier's IP address is used to route network data to those verifiers.

Verifiers also place MATs at stake using the verifier registry. A minimum number of MATs are required to participate in verification, staking, and fee collection.

3.4. The Campaign Registry

An advertiser creates campaigns to track advertising metrics. Each campaign consists of off-chain and on-chain components. The on-chain component is represented by the general information on the campaign and its configuration.

Name	Type	Description
ID	bytes32	The unique ID of the campaign
owner		The address of the owner of the campaign
enabled	bool	The advertiser may enable or disable the campaign

4. FUND MANAGEMENT

4.1. Banking Smart Contracts

Funds are managed through a set of smart contracts on-chain. There is a mechanism for depositing funds for a given campaign and being able to withdraw earnings as a verifier. It is inspired by Plasma Cash. However, unlike Plasma Cash where each leaf in the Merkle tree represents each unique token and its owner, here each leaf is a unique ID representing an account or campaign and its token balance.

4.2. Campaign Balances

The campaign's off-chain part is merkelized and managed by verifiers. Verifiers bill campaigns every voting round and agree on the updated balances. Fraud proofs are used to challenge invalid balances and the campaign state updates.

Each campaign has a unique ID generated by the smart contract. A transaction updating a campaign must be included in the position in the Merkle tree corresponding to the campaign's ID.

4.3. Verifier Balances

Like campaigns, verifier balances are merkelized and their root is stored in each block header.

Each verifier has a unique ID as well, which is their Ethereum public address. Transactions updating the verifier balances must be included in the position in the Merkle tree corresponding to the verifier's ID. Verifiers have two balances, one for their payment token earnings and one for their staking tokens.

4.4. Participant Balances

Participants of the sidechain are also able to receive payments in the system. Like verifier and campaign balances, each participant has a unique ID generated by the smart contract.

4.5. Fund Withdrawals

Fund withdrawals require that the advertiser or account holder (verifier, DSP, etc.) have proof of ownership of the account and Merkle proof. A withdrawal request will include the amount being withdrawn, the current account's balance, and a Merkle proof. The fund management smart contract will verify the proof and the current balance against the requested withdrawal amount. Once the proof and the balance has been verified against the latest state, the tokens are released.

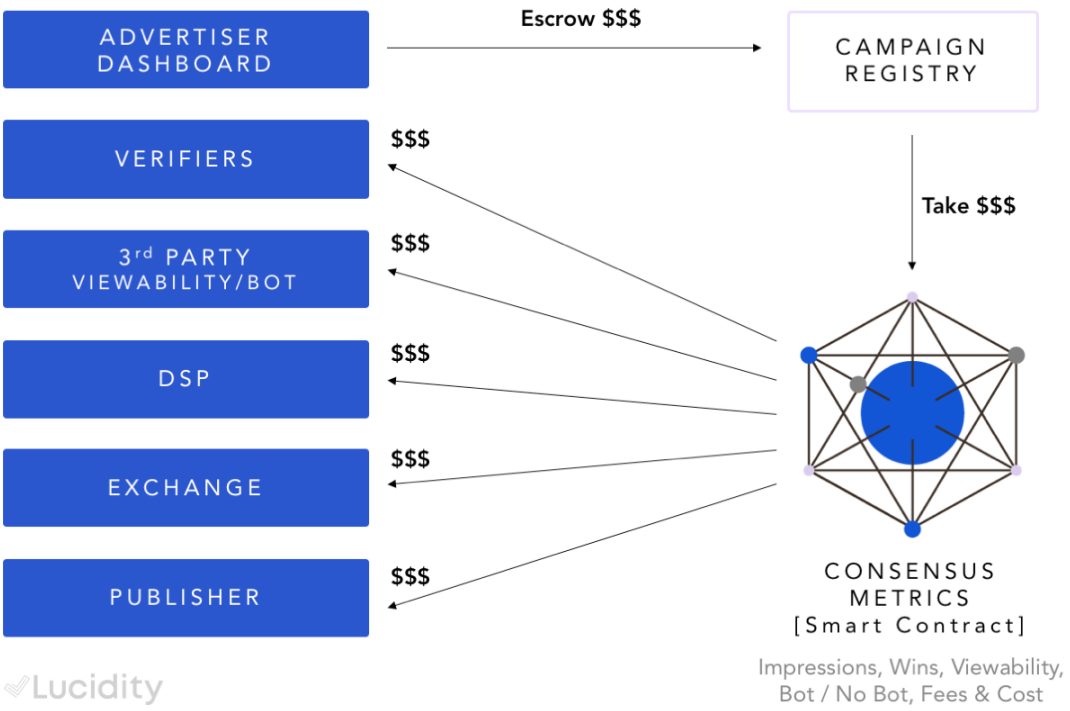


Figure: Payments flow

5. ARCHITECTURE

5.1. A Sidechain Composed of Replicated State Machines

The sidechain is composed of replicated state machines - known as verifiers. Verifiers operate open source software which implements a set of codified standards for the computation of various metrics based on eventful data. The verifier state machine can process billions of events which are represented by a 32 byte hash on-chain.

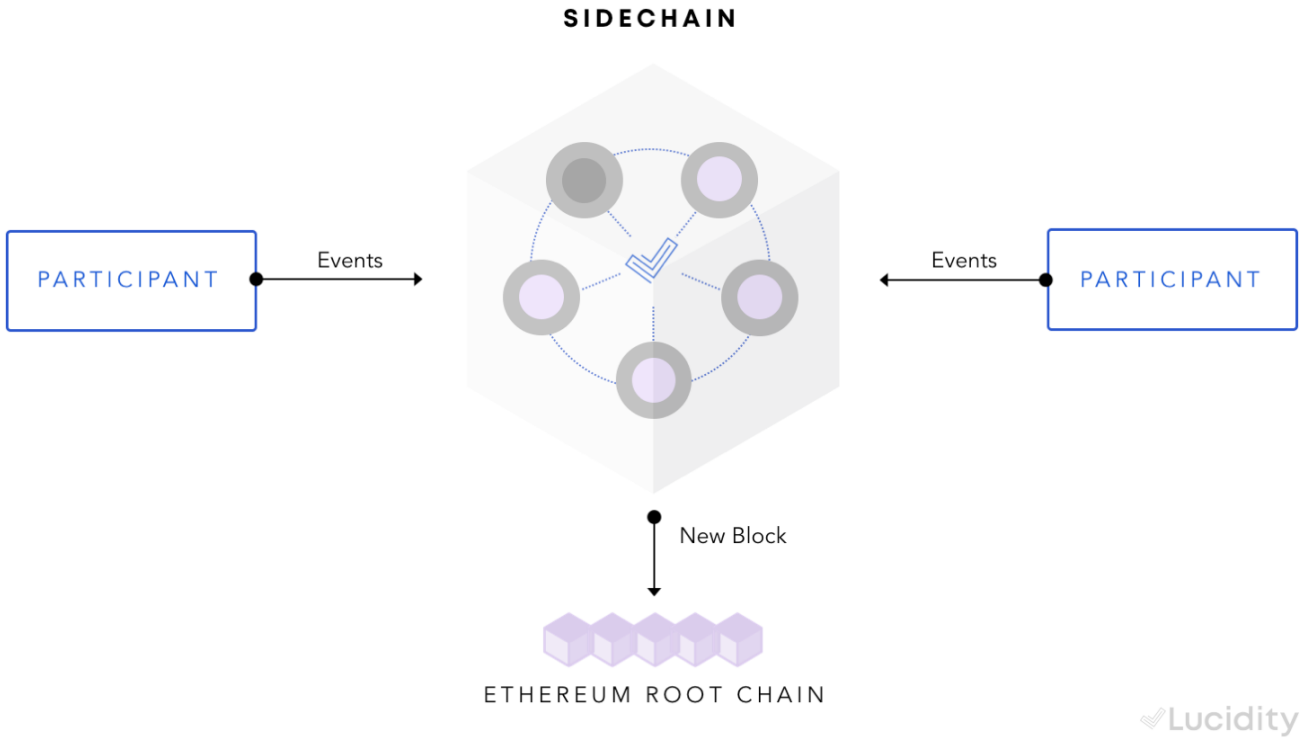


Figure: Sidechain overview

5.2. The Verifier State Machine

Each verifier state machine independently tracks, stores, and computes advertising metrics. This is done by storing a log of all received and computed metrics within the verifier's internal ledger.

Within the verifier, there is specialized code to implement measurement standards. For example, there is an industry definition for loaded impressions. A loaded impression is defined as an impression confirmed by both the advertiser and the DSP. This specification is codified into the verifier as a digital measurement standard. Thus the verifiers guarantee that measurement standards, such as loaded impressions, are followed and enforced.

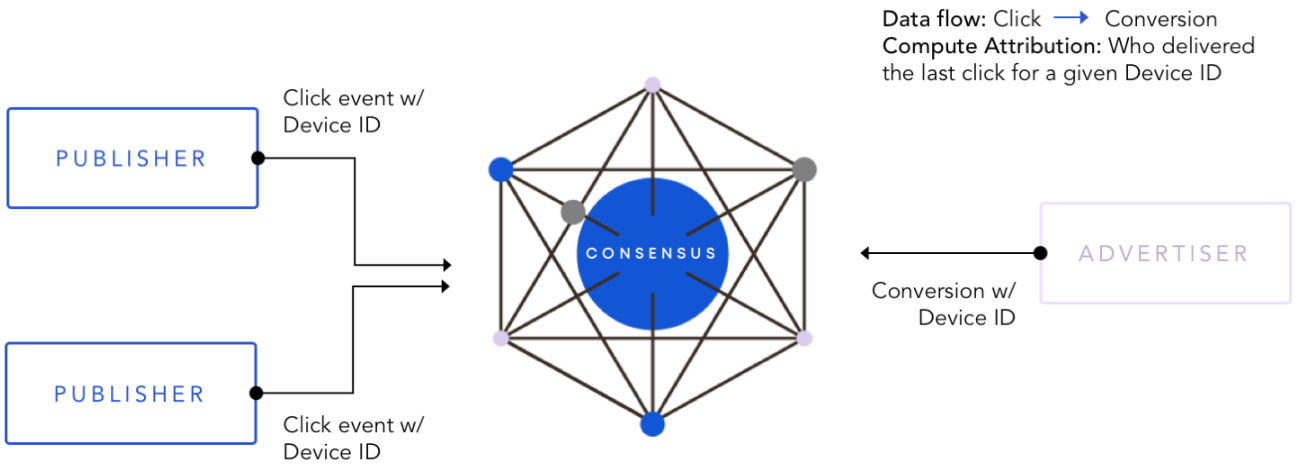


Figure: Decentralized advertising metrics: attributions

Lucidity

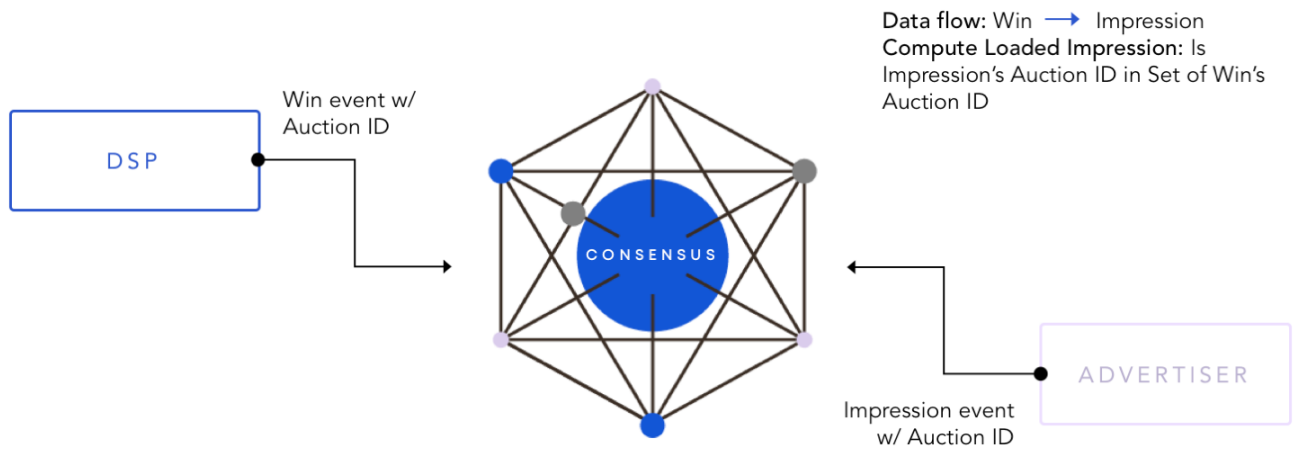


Figure: Decentralized advertising metrics: loaded impression

Lucidity

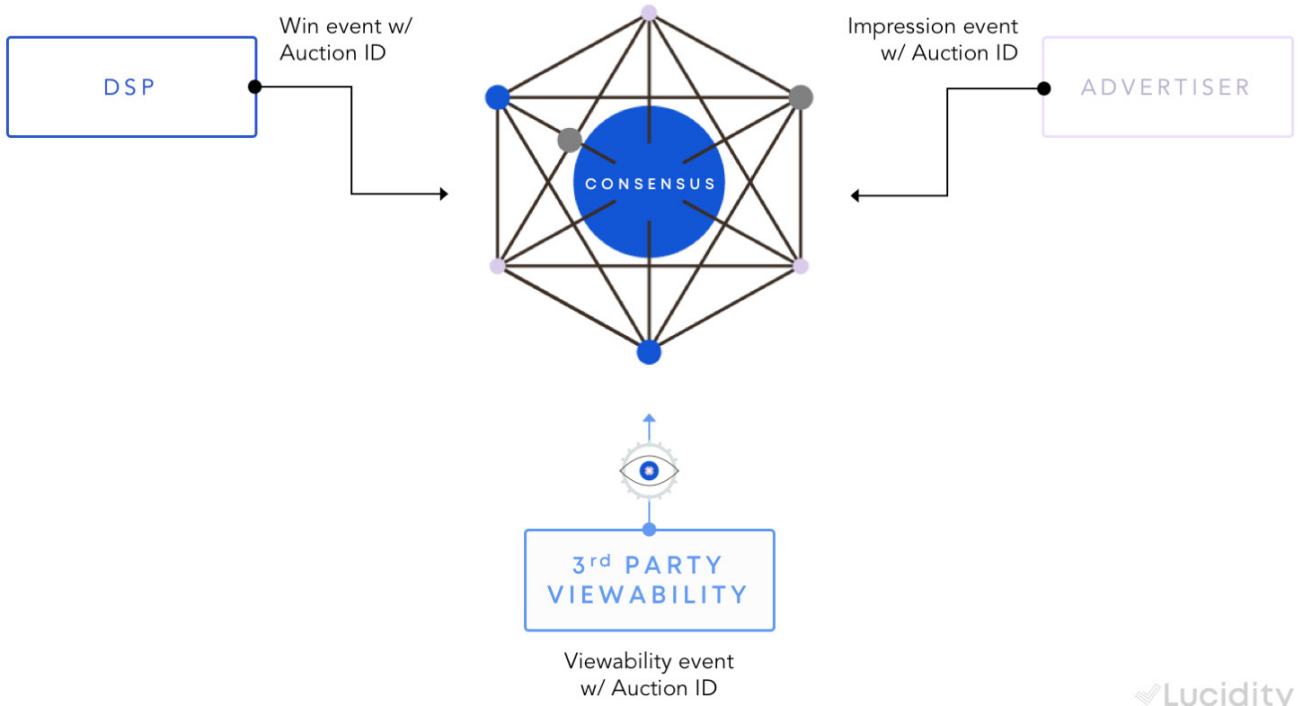


Figure: Decentralized advertising metrics: viewable loaded impression

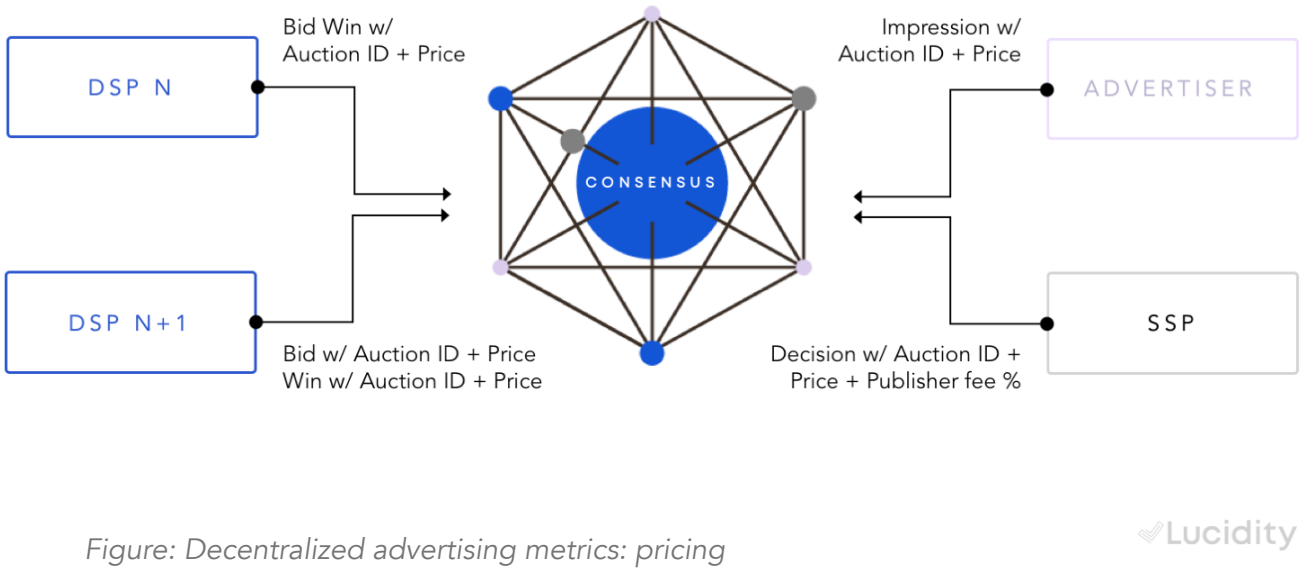


Figure: Decentralized advertising metrics: pricing

5.3. Network routing

The chain's participants can send data to the network by digitally signing data and sending it to every node in the verification network.

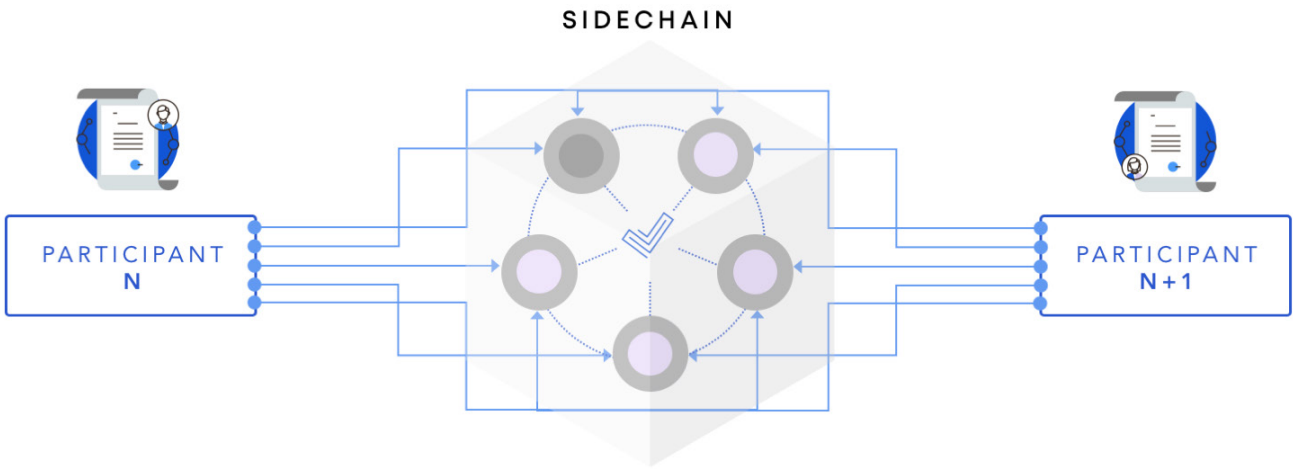


Figure: Networking flow

Lucidity

5.4: Digitally signed data messages

Received eventful data messages are digitally signed by the participant generating the message.

An event entering the sidechain must be signed using an ECDSA keypair. Specifically, the Lucidity protocol uses the "secp112r2" elliptic curve algorithm. This mechanism was chosen to be easily compatible with the IAB's ads.cert digital signing of programmatic bid requests (the IAB is the Interactive Advertising Bureau, an organization that develops standards for the digital advertising industry).

An event's ID is expected to be a universally unique string. Every field described in the eventful message schema is recursively hashed (SHA-256) and is used to generate a cryptographic signature which is included in the event's request body.

Each verifier that receives the event message validates its signature against the participant registry. The content of every eventful data message is also checked against the event's signature. This creates a trail that tags the source of advertising events in the system.

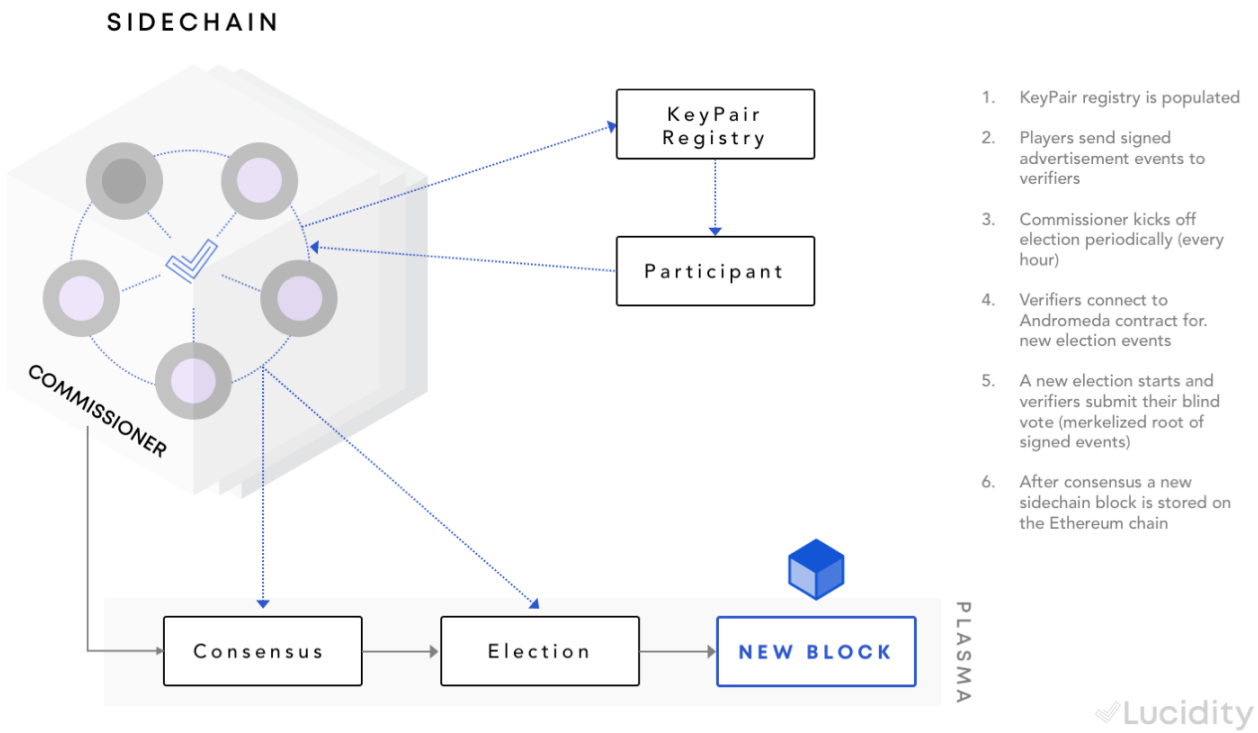


Figure: Sidechain consensus overview

5.5. Proof-of-Stake consensus

Here we describe a majority based byzantine fault tolerant consensus mechanism using Proof-of-Stake.

In the Proof-of-Stake system described, one token receives one vote. Thus, voting is not based on the number of participants but rather on the number of tokens at stake.

Currently, the system uses a commissioner to initiate and count votes, upgrades to this system will be implemented to remove the commissioner. The commissioner is operated by the Lucidity team and is set when the official sidechain on-chain consensus smart contracts are launched.

Below, we explain the voting mechanism when a sidechain's commissioner initiates a consensus election every hour.

The sidechain's commissioner will initiate a new consensus election at the 30th minute of every hour. Each election covers the previous hour from the time the election is initiated. Elections occur as follows:

1. The commissioner can only initiate elections and tally votes. The commissioner has no influence in the results of the election.
2. During each election, each verifier submits a blind vote of the root hash of the set of events collected during that election's time period.
3. The election votes are counted by the commissioner. Verifier vote weighting corresponds to the percentage of tokens a verifier has at stake.
4. When majority of the verifiers arrive at consensus, a newly minted sidechain block is then created, which contains the list of verifiers along with the consensus root hash of the tracked and computed metrics.

5. Verifiers are responsible for updating balances off the main Ethereum chain. They scrupulously and continuously carry out calculations of advertising metrics and participate in blind voting to agree on the updated balance sheet and calculated metrics.

Initially, the advertiser deposits money to the relevant advertising campaign. The deposit is locked in a smart contract. In order to withdraw tokens from a campaign, it is necessary to provide the proof that the money is present in the system (Merkle proof, campaign ID, available balance, advertiser's address).

Before an advertising event gets to the verifier, the event is signed by the relevant entity (advertiser, DSP, SSP, publisher, exchange, or other third party) and ordered by timestamp assigned by the signer. Events are processed in 10-minute consensus rounds. The current round includes events cast during the previous 60-minute interval. That provides enough time for the verifier to receive events as well as updates in any of the registries or fund management smart contract within the time frame.

Every included metric requires a positive balance on the corresponding campaign. If the campaign balance falls to zero, then the unprocessed events are kept pending for 24 hours. When the metrics are computed, the respective amount is deducted from the corresponding campaign balance sheet.

Verifiers monitor events and changes in the registry and fund management smart contracts as well as translate the changes to the off-chain state. These changes are organized as Merkle trees (advertiser balances, verifier balances, and advertising metrics). Only Merkle roots show up on the root main chain. This enables the system to be scaled up to millions of events a second.

The verifiers themselves use ERC20 staking tokens to vote for the next state of the system. To prevent multiple votes by the same token, the smart contract blocks transfers while voting is in progress. Each token gives the right to one vote. The verifier participates in voting every 60 minutes through an on-chain consensus smart contract. Every round the verifier submits an encrypted tuple to the on-chain consensus smart contract (advertising metrics Merkle root, advertiser balances Merkle root, verifier balances Merkle root).

To encrypt the tuple necessary for blind voting, the verifier pseudo-randomly generates a 256-bit number (salt) and hashes (SHA3) the number with the actual data. The polling stage lasts 5 minutes. In the following 5 minutes, the verifier reveals the actual data by submitting the inputs. The tuple which gets the most number of votes is the winner. The winning tuple is a hash that represents the agreed state of the sidechain at that point in time. This tuple is used for various proofs and as a checkpoint of the system.

Economic forces compel verifiers to make correct calculations and also not to withhold the actual data (balance sheet and computed metrics).

5.6. Sidechain blocks

Each block produced through the consensus system is composed of on-chain and off-chain parts. The on-chain parts are called the block header, and the off-chain parts are the block transactions. The block header stores small bits of data used to identify the block and a set of Merkle roots to validate block transactions.

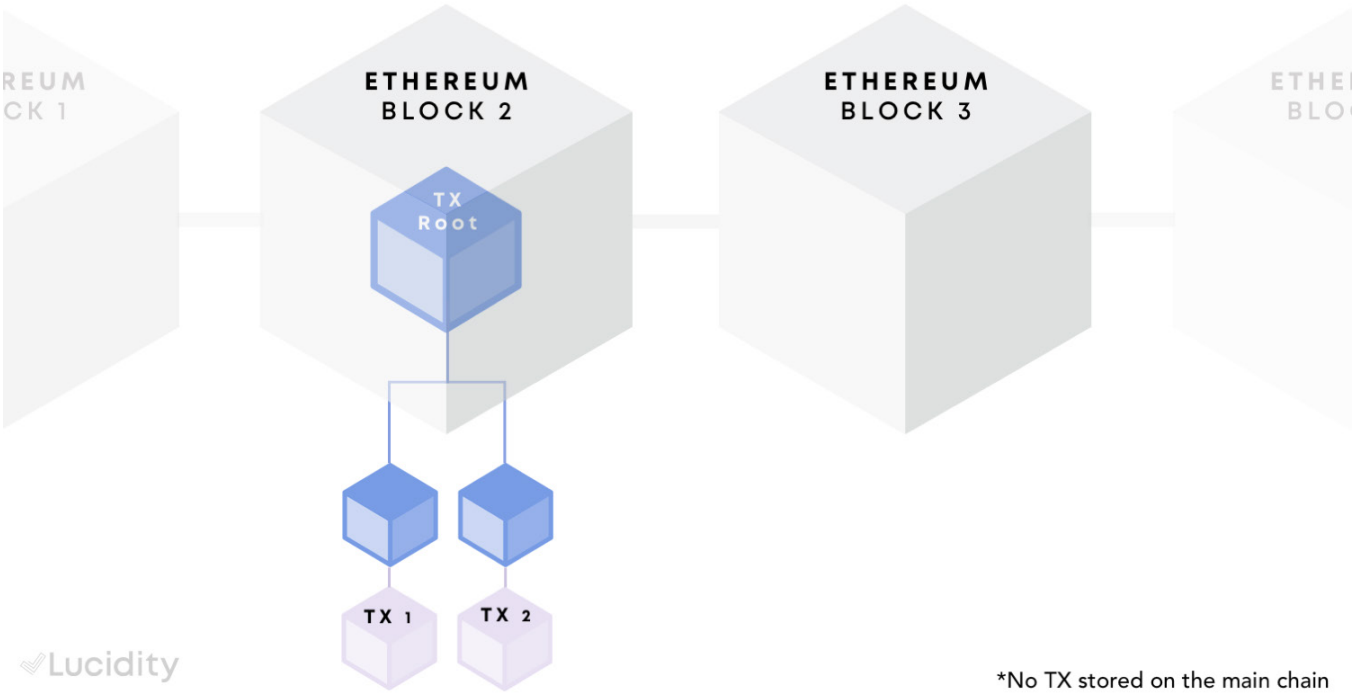


Figure: Sidechain blocks

Below is a list of all the fields in the block header and their description.

Field Name	Field Name
blockNumber	The Ethereum block number at which this block's election took place
startsAt	The start date of the set of data included in this block
endsAt	The end date of the set of data included in this block
seed	A random seed generated by the commissioner
stateRoot	Merkelized time-series segments for the election's start and end time
participantBalancesRoot	The Merkle root of payment token balances for participants
campaignBalancesRoot	The Merkle root of campaign payment token balances
verifierBalancesRoot	The Merkle root of payment token balances for verifier
verifierStakingBalancesRoot	The Merkle root of staking token balances for verifiers

5.7. The State Merkle Root

The state root provides a fingerprint for validating the transactions within an election. The state is computed by each verifier independently based on the data events they received from participants. The block's state root is the root agreed upon by the majority of verifiers.

In order to compute the state root, each verifier first computes all the compound metrics built from participants. Then each verifier performs a time-series rollup by a set of grouping dimensions (advertiser ID, exchange ID, app ID, etc.) and computes the metric values in a map/reduce style set of functions. These grouped and rolled-up analytics are referred to as segments, inspired by Apache Druid's definition of time-series segments.

Each segment has a predictably computed ID which is composed of its dimensions, values, and the election's smart contract address. A Merkle tree is then built by each verifier in which each leaf is a non-null value indexed by the segment's ID. Events are rolled up and event-level measurements are applied by each verifier; thus, there is decentralization of the enforcement of those measurement standards.

Because the IDs of the segments are predictably generated based on its expected election and content, a segment displayed to a participant can be proven to be valid by the participant viewing their specific set of analytics. The participant computes the expected segment ID and checks it against that election's state root.

6. SHARDING

6.1. Decentralization

The architecture described above is an essential first step. However, we must expand upon this to create a decentralized system capable of handling millions of events per second with thousands of verifiers. In order to achieve this, we describe a sharding mechanism that is applied to the underlying sidechain architecture to scale throughput and decentralization.

By utilizing a sharding mechanism, the sidechain network is split into chunks. Each chunk is a subset of verifiers who receive and verify only a subset of all the data in the entire network. By using sharding we decrease the amount of resources each verifier must use to participate in verification. By creating a protocol and architecture that is able to scale (without slowing down) as the number of verifiers increases, we introduce a system that has no theoretical scaling limit.

A key differentiator in the system implementation is that there is no client side aggregation. Every event is routed to the sidechain and independently computed by all the verifiers at the user-session level.

6.2. Network Routing

The primary limiting factor of scaling blockchains is the peer-to-peer gossiping that happens between nodes. The goal of sharding is to break up the verification network into groups. Each group is in charge is validating only a subset of events passing through the sidechain.

We first define a routing ID which is currently a bid request's transaction ID. The mechanism employed to determine to which verifiers a certain event is routed is known as consistent hashing. It is a concept popularized by content delivery networks in which the hash of a file's contents maps to a set of nodes.

With consistent hashing, there is a keyspace in which routing IDs are predictably mapped. Each shard is a point in the keyspace and each routing ID maps to

a shard number. Consistent hashing is advantageous, because the number of verifiers can increase by n+1 and each routing ID will consistently route to the same shard.

Because the routing ID is known by all participants, without directly being connected with each other, every event that is related to the same user session is sent to the same shard. This makes it possible to predictably calculate session-level digital measurements from event-level data without advance knowledge about other programmatic supply-chain participants. Programmatic supply-chain participants are not all connected with each other, even if they participate in the same bid request as it makes its way up the supply-chain. Participants in the programmatic supply-chain only know about their neighboring participant to the left and the neighboring participant to the right.

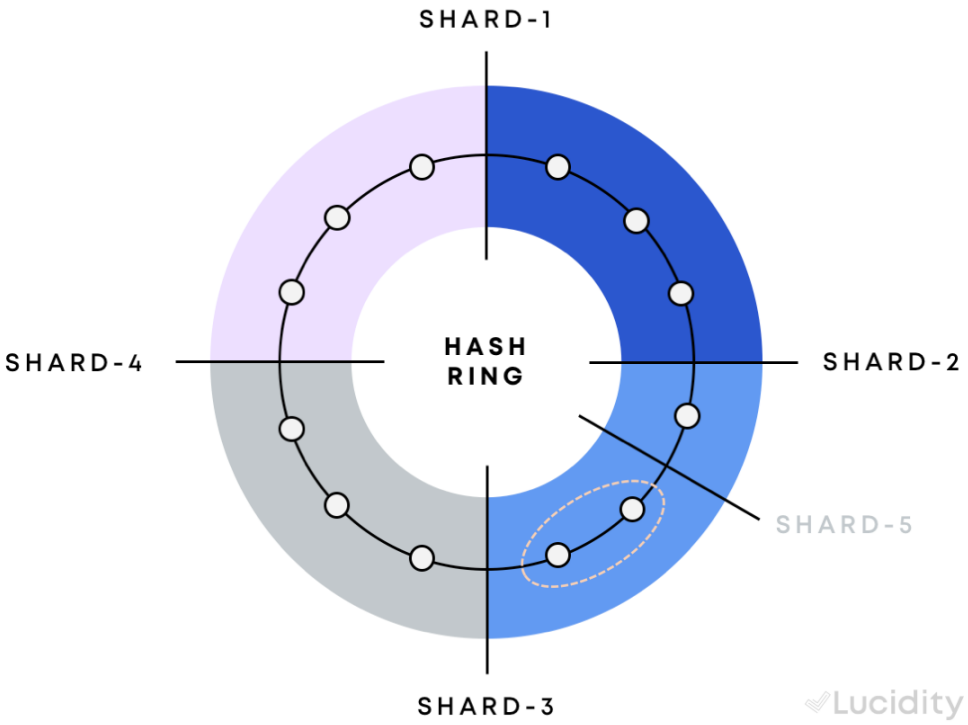


Figure: Consistent hashing ring

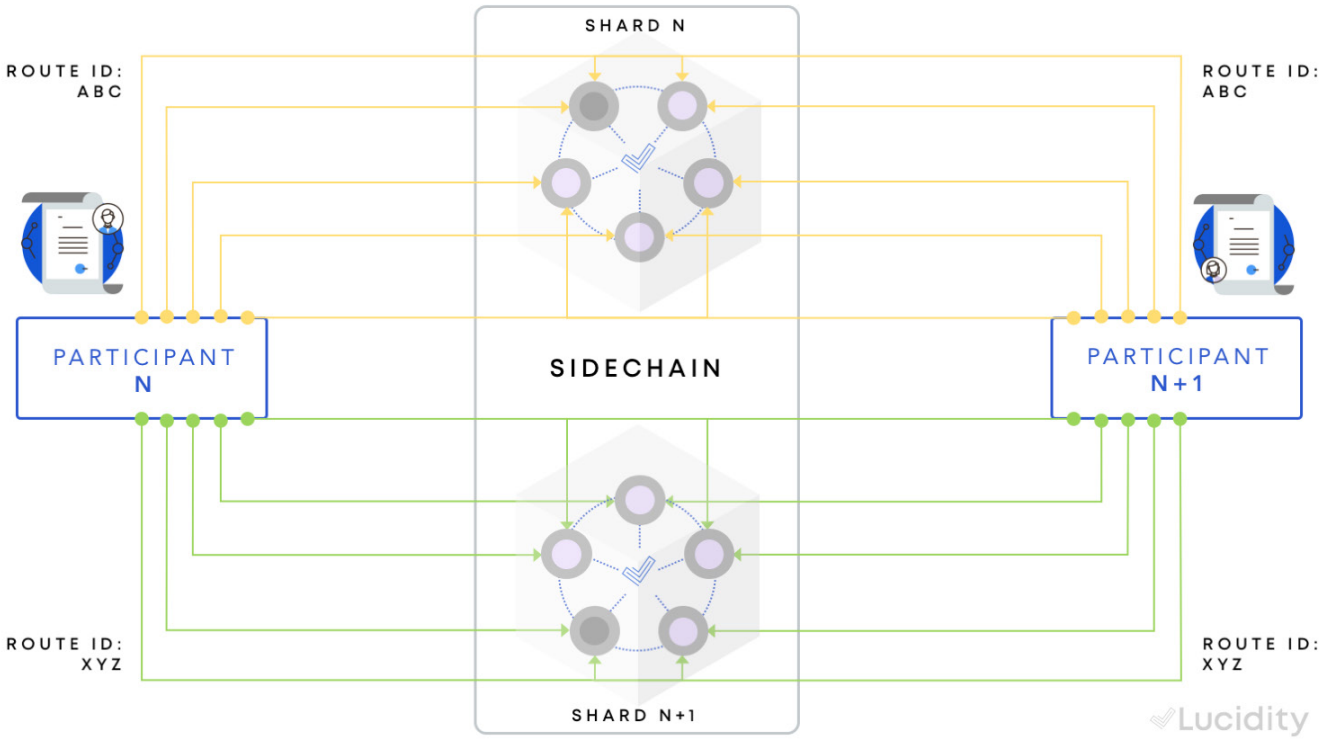


Figure: Load balanced networking

6.3. Per-Shard Consensus

Each shard is composed of a subset of verifiers. Each verifier in a shard receives a digitally signed copy of events from participants. This is similar to sharded replica-sets in existing clustered database systems. At every consensus election round, verifiers within each shard agree on the state of their shard.

The on-chain block headers are then separated by shards, each shard has its own state roots along with their own balance roots. Campaign balances are split into each shard. Because verifiers belong to only one shard, they only hold balances relevant to their specific shard. Therefore, the architecture outlined above is partitioned by creating Merkle roots per shard.

6.4. Global State

A global state derived from the partitioned network can be achieved by mapping each shard and reducing its states. As previously described, transactions within each block are rolled up time-series segments. Each segment is a combination of dimensional IDs and metric values. Like Plasma, we utilize map/reduce and root validation of each shard to achieve a safe global state. Segments across shards are mapped by their dimensional IDs and their values are summed during the reduce phase.

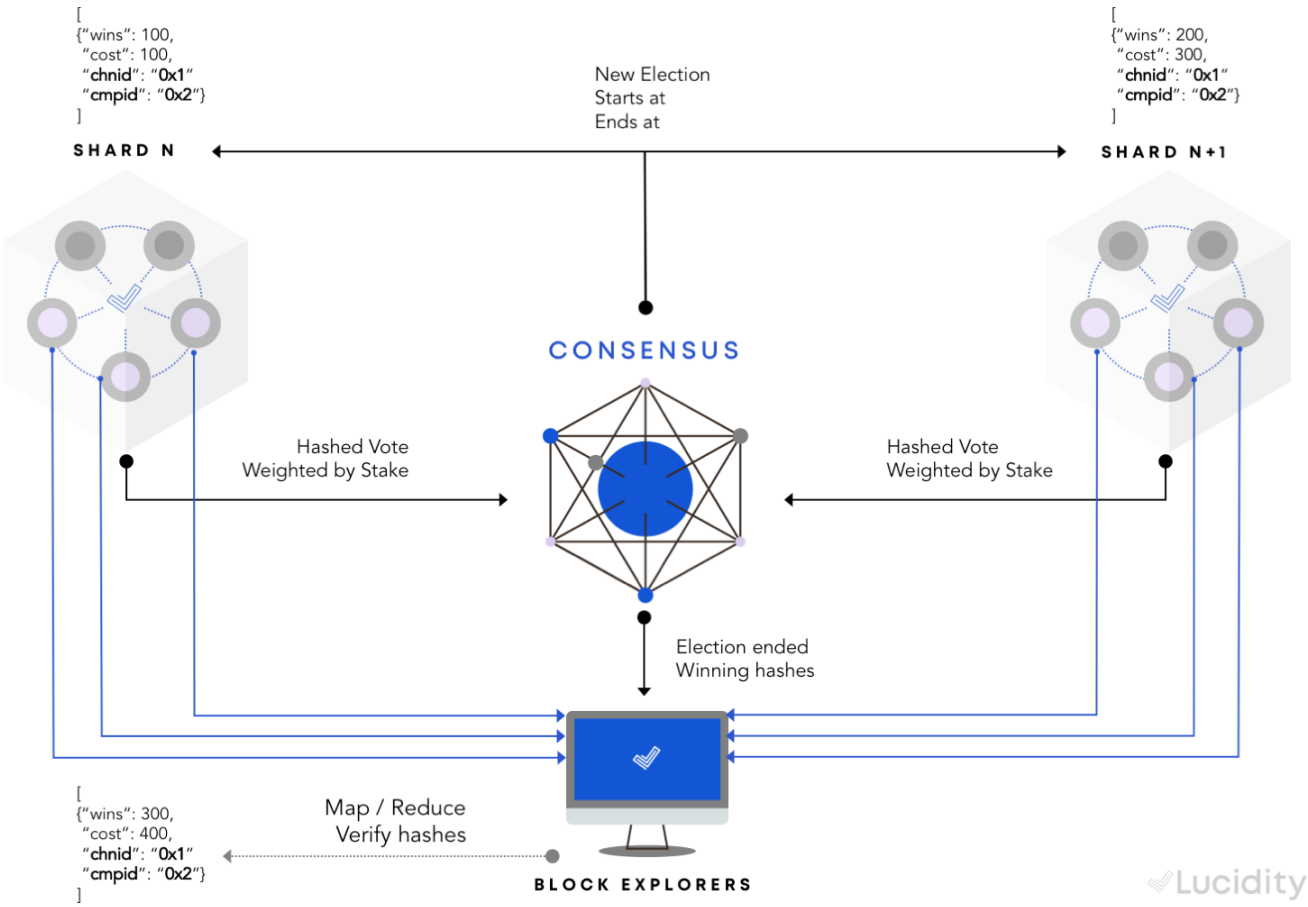


Figure: Sharded global state via map/reduce

7. PLASMA

7.1. Lucidity is Plasma

The architecture heavily uses the Plasma philosophy when it comes to using Merkle trees and hashes to define mechanics for decentralized and safe sidechains. Lucidity is Plasma XT is Plasma Cash is Plasma. This simply means that the Lucidity payments system is based on concepts of Plasma XT which itself is based on Plasma Cash which is based on Plasma.

Plasma provides two key mechanisms: a way of doing token transfers within a sidechain through on-chain smart contracts and a mass exit mechanism for exiting faulty chains.

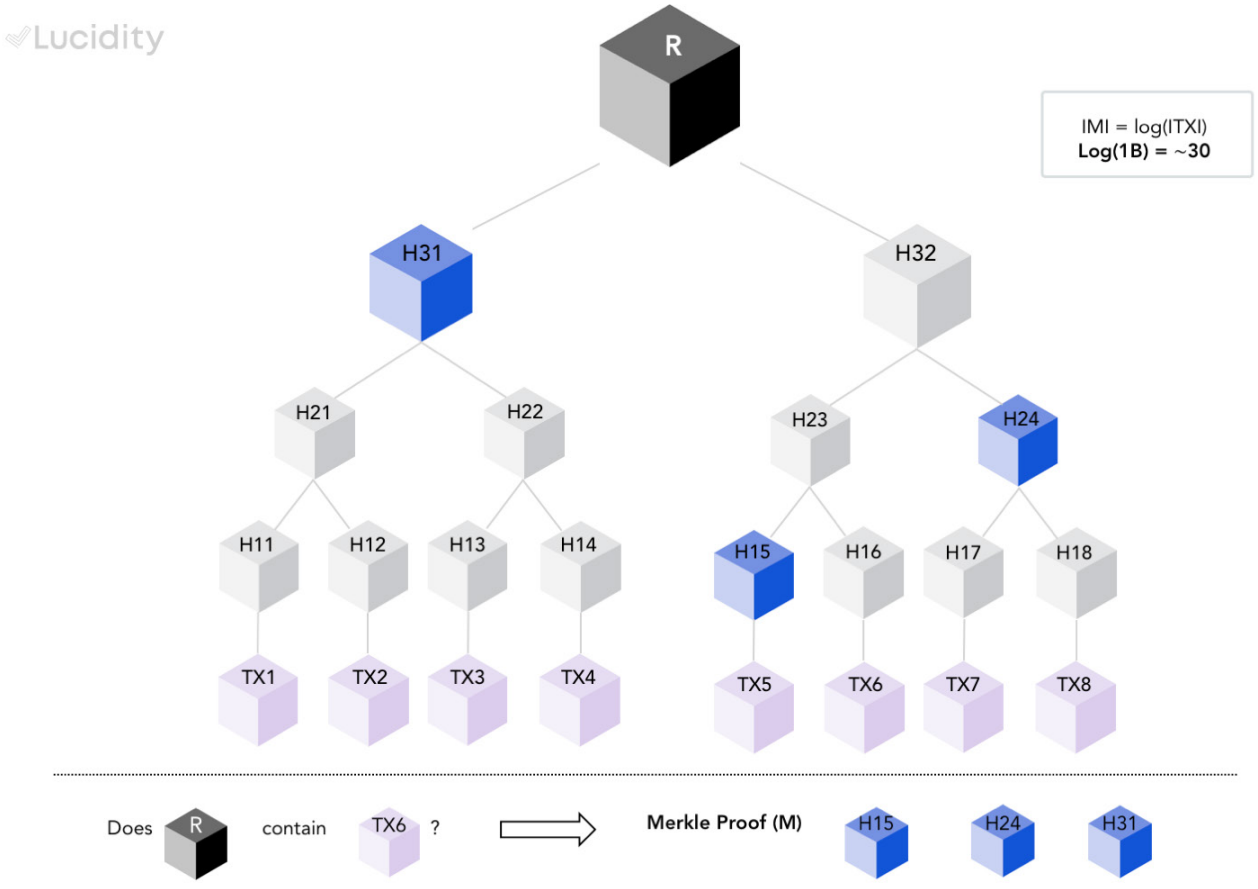


Figure: Merkle tree

7.2. Mass-Exits

One of the primary challenges of sidechain systems is data availability, if raw data is unavailable there is no way to prove that transactions within a given block are invalid.

Participants who have tokens deposited into the sidechain fund management contract can exit a faulty sidechain by waiting a number of blocks after a consensus election is finished for block data to be shared by the verifiers. If the verifiers don't share a block's data within a certain number of blocks, then participants are able to safely exit the fault sidechain.

The fund management contract has a number of blocks that must pass before verifiers are able to withdraw their tokens. This gives an overlap in time when the participants are able to withdraw their payment tokens before verifiers are able to withdraw tokens. When the verifiers are determined to be withholding data, participants are able to withdraw their deposited tokens safely and without disruption.

This mass-exit mechanism is guaranteed by the Plasma-enabled fund management smart contracts. With Plasma, there is an ordered transaction model, also known as unspent transaction outputs (UTXOs), to ensure that participants are able to withdraw their tokens before the verifiers are able to falsely claim ownership of the participant's tokens.

The UTXO model ensure that withdrawals from participants are processed before the faulty verifier's withdrawal request. The current implementation has been open sourced to promote wider adoption of the Plasma philosophy: https://github.com/luciditytech/lucidity-plasma.

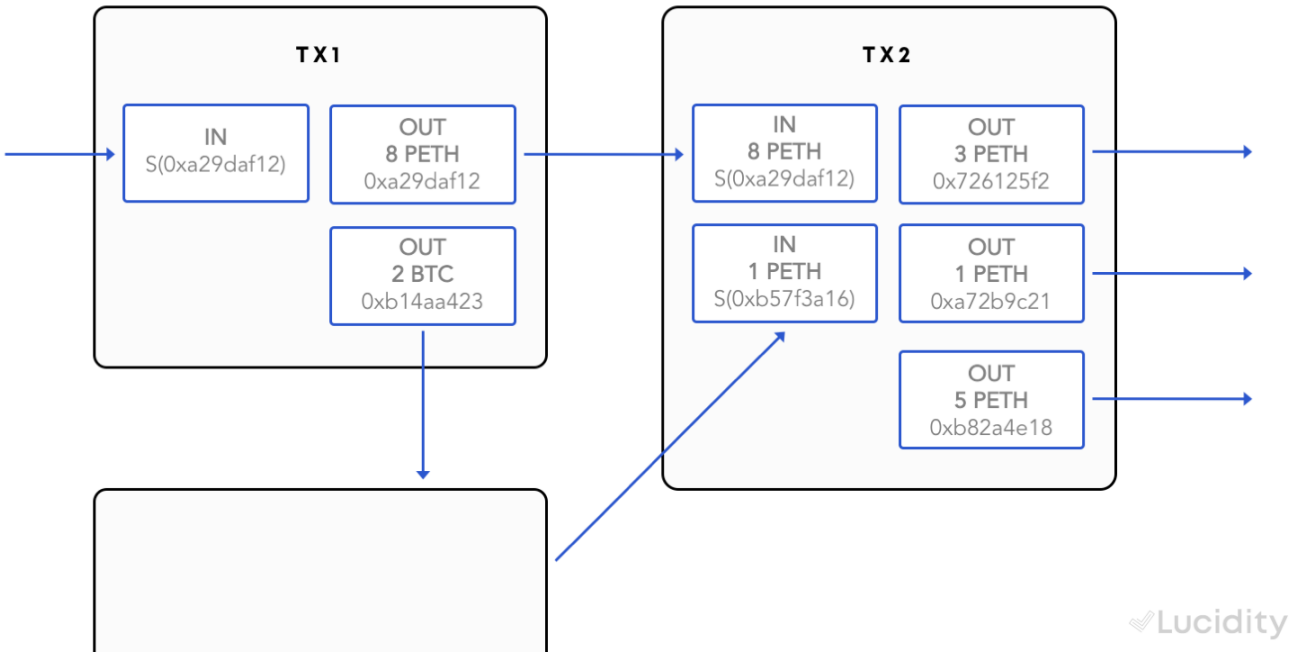


Figure: UTXO Model

8. GOVERNANCE

8.1. Governance Overview

Governance is composed of both on-chain and off-chain processes. The governance process will continue to be fine tuned to maximize trust in the sidechain.

Verifiers and participants are able to participate in on-chain voting to updates system wide parameters. Verifiers and participants are also able to participate in off-chain voting to update the definition of digital measurement standards.

8.2. Verifier registrations

Verifier registry happens through an off-chain governance process. The process of adding verifiers must be discussed by the existing verifiers and participants. Adding verifiers means that reward fees must be shared among the verifiers, but it also means adding more trust to the system and encourages advertisers to use the network.

As soon as the verifier is added to the registry, the verifier can start computing advertising metrics and voting.

8.3. Advertising measurement standards

The way that advertising metrics are defined is managed through an off-chain governance process. Advertising industry standards, such as the Interactive Advertising Bureau, may propose changes to each codified standard defined within the sidechain. Participants and verifiers who have stake within the sidechain are able to participate in the process of accepting updates to measurement standards.

When a new digital measurement standard, or update to an existing standard, is accepted a new version of the verifier software is released. An Ethereum block number is defined in which the verifiers will start to execute and add the new standard to the consensus rules.

This update mechanism is a key concept of adding a way of codifying digital measurement standards and updating them in an efficient manner.

8.4. Network upgrades

Token contract is implemented in the way which allows the owner to introduce further improvements. While the balances remain immutable, new methods can be added to ease integration with other smart contracts. To simplify data migration, all data is abstracted down and stored in primitive key-value pairs. A standard naming system along with the SHA256 hash algorithm is used to find the values of data. Such an approach will minimize an effort to migrate smart contracts and reduce gas and time needs for the deployment.

Consensus-related smart contracts might undergo hard and soft forks to improve the economy and adopt the most recent developments and Ethereum features. All forks are discussed on public forums and agreed in advance. Any fork will generally happen at a predefined block number.

The way the verifier computes advertising metrics might be changed over time. The logic is separated within the verifier's node and does not affect the protocol itself. The node upgrades must be published weeks in advance to allow the node operator to review the source code and upgrade their verifier node mindfully.

8.5. Global Parameters

Every token holder can participate in on-chain voting for global parameters. Global parameters affect the overall system. The most important parameters are listed in the table:

Name	Description
cpm_price	The cost an advertiser pays for one thousand ad impressions
vote_quorum	The percentage of total votes needed for the Supporting side to be declared the winner
commit_stage_length	The length of time the Voting Commit stage for a particular change will last
reveal_stage_length	The length of time the Voting Reveal stage for a particular change will last

8.6. On-chain voting process

Lucidity is in charge of initiating voting on the Ethereum blockchain after a proposal has been discussed on public forums. This allows every stakeholder to present proposals and reasoning for every change. Every token holder can participate in the voting. The voting consists of two rounds. The first round to cast a vote, and the second round to reveal the vote. Hashes are introduced to preserve the uncertainty of the election results.

A token locking mechanism is used to avoid a double vote attack. The mechanism is embedded into the token contract. Casting a vote locks tokens indefinitely until the voter reveals their vote.

9. INCENTIVES & PENALTIES

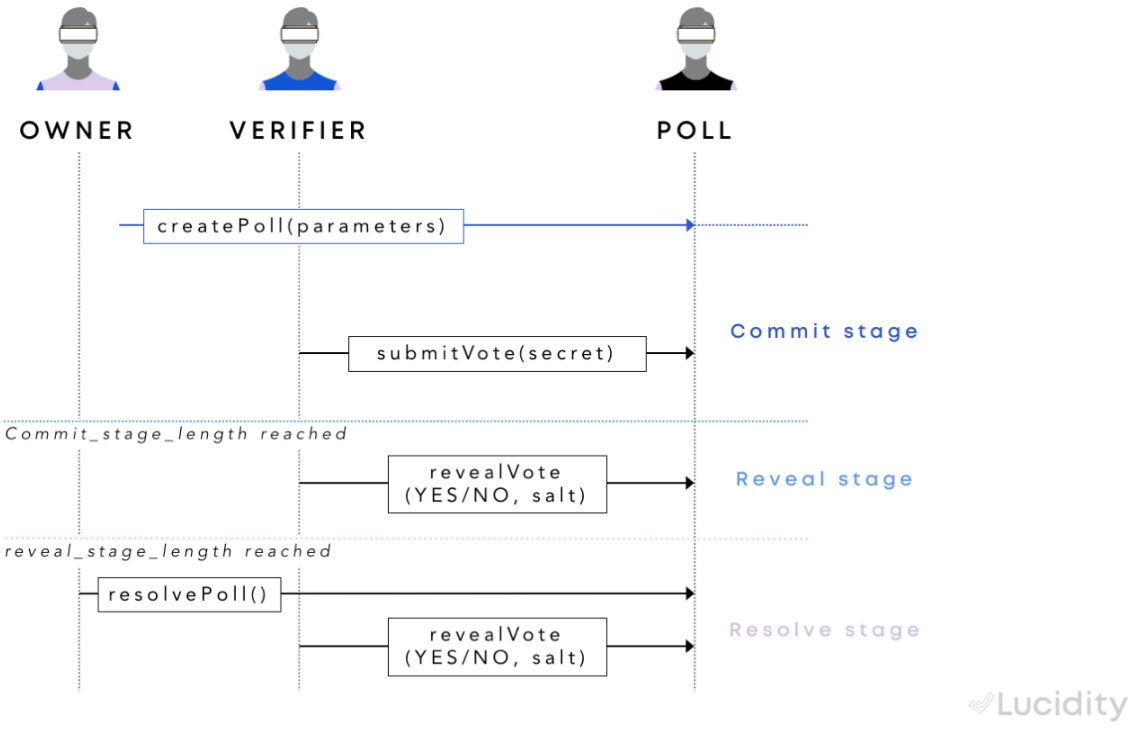


Figure: Voting process

9.1. Economic Model

Verifiers who correctly participated in consensus are paid in the payment token proportional to the number of staking tokens they have at stake.

The price per event computation per token is one of the global parameters of the system which is set through voting managed by staking token owners. Verification of an ad impression is an example of an event computation. Not only are verifiers able to vote for the next block and balance sheet, but also choose the right price per event computation per token according to various market factors.

Advertisers may pay using fiat currency, and a delegate in turn may pay for the campaign by utilizing a payment token. Ultimately, a payment token is used and goes to verifiers who compute correct metrics and share the results with others. They must deposit the payment tokens which will be used as form of payment to the verifiers for their work.

Verifiers' revenue for their computational work depends on the various factors of the system such as the price of staking and payment tokens, the number of voters, and the cost of running a computation service. However, in short, more event computed equals more revenue. And, the more billable events that the verifiers are able to compute, the higher the payouts.

Verifiers are highly concerned about being able to produce precise metrics since the right proposal gets the most votes, i.e., the proposal is agreed by the majority. Wrong proposals cause wasted computing power, the loss of potential earnings, and the slashing of the verifier's staked tokens. As a result, correspondent blocks are marked as invalid and all engaged in invalid voting get penalized.

Processing around 20 billion impressions per day (~250,000 per second) requires high-performance clusters working 24/7 and advanced cloud-based technologies. Verifiers' source code is available open-source. It is a highly-scalable and fault-tolerant solution developed by Lucidity and other contributors. The performance of the software should be constantly improved to reduce the service costs.

Staking token popularity depends on the market adoption and integrity of the system. The most obvious factors are:

- Number of active advertisers / campaigns;
- Number of processed metrics;
- Absence of errors;
- Data availability;
- Market liquidity;
- Simplicity of use

Simplicity of use is a critical factor. To become a verifier one would need to:

- Register in the global list of verifiers;
- Set up a node;
- Purchase staking tokens;
- Participate in voting

9.2. Yield

Below we provide the economic model based on the number of impressions verifiers could be paid for as computed events in the US in 2015. It is important to note that impressions are just one billable metric, otherwise called an event to be computed by verifiers, for the sidechain mechanism is able to support other billable metrics such as clicks, viewability, conversions, attributions, audience segments, etc. This model provides a simplified view of the projection of fees to be paid to verifiers using only impression events.

Based on the the average number of daily impressions, ROI can be calculated as follows at launch:

Daily Impressions	20 billion
US Ad Market Share	10%
Daily Impressions Tracked	2B
MAT Tokens in Circulation in 1st Year	180M
MAT Price	$0.15
$ / Token / Day	$0.056
$ / Token / Year	$0.203
Yield	**135%**

** Assumes all token holders are operating nodes*

Each token at stake will receive a fee of each event computed. The fee rewards are relative to the protocol's market share :

US Market Share (Impressions Only)	10%	20%	40%	60%	80%	100%
Yield	135%	270%	541%	811%	1081%	1352%

** Assumes all token holders are operating nodes*

9.3. Economic protection

One of the main reasons for choosing to use Ethereum as opposed to a private blockchain system is the availability of its ERC20 token definition. This allows the creation and distribution of MATs and the usage of the Dai stablecoin.

This allows verifiers to have skin in the game by requiring MAT ownership and staking to participate in verification. The purchasing and staking of MATs is a form of collateral in the case the verifier is proven to be acting out of consensus rules.

A verifier's invalid proposals during consensus are penalized. An invalid consensus is when 2/3rd of the verifiers agreed on a different state than the verifier. Every time a verifier is out of consensus, 1/3rd of their staked tokens are destroyed. If the verifier is continuously out of consensus, they will lose their staked tokens and be unable to further participate in consensus.

This solves the nothing-at-stake problem while providing economic security to the sidechain. Because the staked tokens have true economic value, there are financial consequences to verifiers for attempting to manipulate consensus rules.

10. THE BLOCK EXPLORER

10.1. A General Analytics Interface

The block explorer is owned and operated by the commissioner of the sidechain. It provides a layer of privacy while still being able to provide the trust and transparency through the public blockchain. The block explorer is designed to operate under strict operational conditions enforced by on-chain smart contracts.

The block explorer listens for consensus rounds through on-chain smart contracts. At the end of each election, when the root hashes have been determined, the block explorer requests the raw block transactions from the verifiers who correctly participated in consensus.

The block explorer can validate the raw transactions against the root hashes stored in the sidechain block headers. The block explorer is able to validate the data from the verifiers so that it cannot be manipulated by the verifiers themselves.

The block explorer is able to provide authorized access to the analytical state data generated by the verifiers. The block explorer's authentication mechanism ensures that only participants that participated in certain transactions may view those transactions.

Dimensions include: the advertiser, campaign, DSP, exchange, domain, app, etc. Metrics include both simple and compound metrics: impressions, clicks, loaded impressions, etc.

10.2. Proof-of-Inclusion

In order to enable trust that the block explorer is not manipulating data, while not being able to view all of a block's transactions as a participant, a proof-based mechanism for validating segments has been implemented. Data is verified by using the Merkelized state root hash for a given election that's stored on the public chain and providing the Merkle proof that each analytic segment that the participant is viewing exists within a given block. The ID is a recursive hash of the dimensions and metrics in each analytic segment. Each leaf in the state Merkle root represents that an element in the analytic segments exists by its ID.

The Merkle proof can be checked against the sidechain's on-chain block headers using MetaMask such that the participant can validate each analytical segment against the public Ethereum blockchain.

10.3. Proof-of-Funds

Because the block explorer holds access to all the raw transactions generated by the

verifiers, it is responsible for generating Merkle proofs of balances. Unlike Plasma Cash/XT, this removes the burden of participants keeping the proofs. In order to withdraw tokens from the fund management smart contract, the participant or verifier must first get the proof provided by the block explorer. The proof, coupled with their private Ethereum key, allows the withdrawal of token earnings from the sidechain.

The block explorer, or its operators, are unable to take ownership of either payment or staking tokens. The block explorer is also unable to manipulate analytical data. This is guaranteed by the consensus and fund management smart contracts connected to the registries. Although the block explorer is owned and operated by a centralized organization, it is able to provide a trusted way of being forced to operate under strict conditions enforced by a set of public smart contracts.

11. TOKEN ALLOCATION AND RELEASE

Below are the current plans for token allocation and release. It is important to note that the company may make changes to the current plan based on a number of factors including, but not limited to: regulation, changing market conditions, and economic / token economic factors.

There will be a total of 1 billion Marketing Analytics Tokens (MAT). Any undistributed tokens will be burned within 5 years of tokens becoming freely tradable. Tokens will have different "vesting" schedules. n general, purchased tokens will be distributed before any bonuses or incentive tokens.

Locked tokens will have the same rights as Unlocked tokens except that Locked tokens must be held and will not be available for trade nor exchange. Otherwise, Locked tokens enable participation as a Verifier as well as voting governance.

11.1. Token Allocation

There are six noteworthy categories of participants in the MAT token allocation plan. The definition of each are:

Lucidity - Lucidity will retain a 30% stake of the total token pool on a non-diluted basis in order to retain significant influence over the governance of the protocol. Furthermore, Lucidity will utilize the revenue share from running nodes to fund its continued business and development efforts of the protocol.

Team - The Lucidity team, including founders, employees, and contractors will retain 25% of the total token pool on a non-diluted basis. Token grants (and Token option grants) will serve a purpose similar to employee stock option grants.

Advisors - The Advisory pool is equal to 8% of the total token pool on a non-diluted basis. Non-granted advisory tokens will be burned at the end of the token distribution process.

Partnership Reserve - Approximately 19% of the total token pool on a non-diluted basis is reserved for key strategic partners. The strategic partners are primarily either: (1) advertising industry companies, or (2) node operators.

The majority of the tokens from this pool are for allocations to advertising industry companies such as major agencies and advertising technology companies. These allocated tokens are intended to accelerate advertising industry participation in the protocol and to provide a lower risk and lower cost point of entry. Costs, if any, may vary by partner. In addition, these allocated tokens will encourage the voice and perspectives of the advertising industry in the corporate governance of the protocol.

Tokens are also set aside and allocated to incentivize large miners to provide the computational power necessary to operate the protocol.

Marketing and Bonus - The company has set aside tokens to be used for marketing purposes and to reward early token owners with bonuses in addition to the proposed $0.15 token price. Marketing allocation tokens will be used for activities and partnerships related to market making, platform adoption, and branding. In addition, the company will reward early and strategic token owners with tokens from this pool. Unused Marketing & Bonus pool tokens will be burned.

Strategic Round - Token owners from our Strategic Round received a Convertible Note along with a Token Rider that included the following equation: For every dollar invested on the Convertible Note, investor to receive tokens at the equivalent price of $0.15 per token with a bonus of 60%. Strategic Round participants include institutional funds from strategic industries/geographies/platforms, traditional venture capital companies (providing strategic guidance and their network), and companies from the advertising industry.

Private Round - Participants in the three Private Rounds will receive tokens at $0.15 per token and bonuses for the first two Private Round funding tranches.

"Lucidity Token Allocation Plan" is provided below:

LUCIDITY TOKEN ALLOCATION PLAN

	Price	*$ Bought*	*Tokens Purchased*	*Bonus*	*Bonus Tokens*	*TOTAL TOKENS*	*RUNNING SUM*
Lucidity	$0.15	$0	-	0.00%	300,000,000	300,000,000	$0
Team	$0.15	$0	-	0.00%	250,000,000	250,000,000	$0
Advisors	$0.15	$0	-	0.00%	80,000,000	80,000,000	$0
Partnership Reserve	$0.15	$0	-	0.00%	164,333,333	164,333,333	$0
Marketing & Bonus	$0.15	$0	-	0.00%	25,000,000	25,000,000	$0
Strategic Round	$0.15	$6,000,000	40,000,000	60.00%	24,000,000	64,000,000	$6,000,000
Private Round							
Round 1	$0.15	$5,000,000	33,333,333	30.00%	10,000,000	43,333,333	$11,000,000
Round 2	$0.15	$5,000,000	33,333,333	20.00%	6,666,666	40,000,000	$16,000,000
Round 3	$0.15	$5,000,000	33,333,333	0.00%	-	33,333,333	$21,000,000
TOTAL		**$21,000,000**	**140,000,000**		**860,000,000**	**1,000,000,000**	

11.2. Token Release

The table below illustrates the schedule by which tokens will be "released" to the various participants ("released" is equivalent to Unlocked). As mentioned earlier, purchased tokens will be made available before bonus and incentive tokens.

Lucidity - It is likely that Lucidity will hold a meaningful number of its tokens. These tokens will be subject to a lockup of 3 years after tokens are freely tradeable. After the 3 year "cliff", Lucidity's tokens will become available in tranches of 25% per quarter over the course of four consecutive quarters.

Team - The Team's tokens will have a 1 year cliff and an Unlocking schedule over the course of two years. More precisely, the first 12.5% tranche of Team tokens will be Unlocked after one year from the date tokens are freely tradeable. Thereafter, an additional 12.5% will be Unlocked every quarter.

Advisors - Advisor's tokens will be subject to the same Lockup as the Team: 1 year cliff and an additional two year Unlocking period.

Partnership Reserve - The Partnership Reserve's tokens will have a two year cliff with Unlocking over the course of the subsequent two years. More precisely, the first 12.5% tranche of Partnership Reserve's tokens will be Unlocked one year after tokens become freely tradeable. Thereafter, an additional 12.5% will be Unlocked every quarter.

Marketing & Bonus - Tokens from the Marketing & Bonus pool will be subject to a 1 year cliff with Unlocking over the course of one year. Lucidity reserves the right to change the unlocking schedule in particular for tokens used related to market making.

Strategic Round - The Strategic Round's tokens will have 25% of their tokens Unlocked at time tokens become freely tradeable. They will have an additional 25% of their purchased tokens Unlocked every quarter over the next three quarters. Strategic Round's bonus tokens will be unlocked one year from date tokens are freely tradeable.

Private Round - The Private Round's tokens will have 25% of their tokens Unlocked at the time tokens become freely tradeable. They will have an additional 25% of their purchased tokens unlocked over the next three quarters. Private Round's bonus tokens will be unlocked one year from date tokens are freely tradeable.

YEAR 1:

	Year 1				
	EXCHANGE	*M3*	*M6*	*M9*	*Year 1 TOTAL*
Lucidity	-	-	-	-	-
Team	-	-	-	-	-
Advisors	-	-	-	-	-
Partnership Reserve	-	-	-	-	-
Marketing & Bonus	-	-	-	-	-
Strategic Round	10,000,000	10,000,000	10,000,000	10,000,000	40,000,000
Private Round					
Round 1	8,333,333	8,333,333	8,333,333	8,333,333	33,333,333
Round 2	8,333,333	8,333,333	8,333,333	8,333,333	33,333,333
Round 3	8,333,333	8,333,333	8,333,333	8,333,333	33,333,333
TOTAL	**35,000,000**	**35,000,000**	**35,000,000**	**35,000,000**	**140,000,000**

YEAR 2:

	Year 2				
	M12	*M15*	*M18*	*M21*	*Year 2 TOTAL*
Lucidity	-	-	-	-	-
Team	31,250,000	31,250,000	31,250,000	31,250,000	125,000,000
Advisors	10,000,000	10,000,000	10,000,000	10,000,000	40,000,000
Partnership Reserve	-	-	-	-	-
Marketing & Bonus	16,416,667	16,416,667	16,416,667	16,416,667	65,666,667
Strategic Round		-	-	-	-
Private Round					
Round 1	-	-	-	-	-
Round 2	-	-	-	-	-
Round 3	-	-	-	-	-
TOTAL	**57,666,667**	**57,666,667**	**57,666,667**	**57,666,667**	**230,666,667**

YEAR 3 & 4:

	Year 3	Year 4	
	Year 3 Total	*Year 4 Total*	*TOTAL*
Lucidity	-	300,000,000	300,000,000
Team	125,000,000	-	250,000,000
Advisors	40,000,000	-	80,000,000
Partnership Reserve	82,166,667	82,166,667	164,333,333
Marketing & Bonus	-	-	65,666,667
Strategic Round	-	-	40,000,000
Private Round			
Round 1	-	-	33,333,333
Round 2	-	-	33,333,333
Round 3	-	-	33,333,333
TOTAL	**247,166,667**	**382,166,667**	**1,000,000,000**

12. HIGH FREQUENCY TRANSACTIONS & DISTRIBUTED LEDGER INFRASTRUCTURE

12.1. Supply-chain Transparency at Scale

While we have focused on a way of defining and enforcing measurement standards in digital advertising, the mechanisms described in this paper can be generally applied to implement high-frequency supply chain distributed ledgers. In other words, digital advertising is an ideal example of building industry specific apps on top of this infrastructure architecture, but it is not the only use case for the system we have built.

The core system describes a decentralized and economically protected mechanism for ingesting high-frequency data from dispersed entities. The generalized state machine could compute a wide variety of multi-party based metrics.

The verifiers can be generalized so that businesses could deploy their own custom business logic and set of governance rules to the sidechain.

12.2. Sidechain Advantages

A Plasma-style sidechain is used to avoid being bottlenecked by the root Ethereum blockchain, while still being able to use Ethereum's public token standard. The system described is root chain agnostic, meaning that it can be pegged to a different private or public blockchain and is not limited to Ethereum. A core blockchain concept, Merkle trees, is utilized to efficiently represent billions of eventful data points cheaply and efficiently on a public chain. By optimizing system implementation, we are able to avoid costly Ethereum gas fees.

In sum, the system described in this paper will benefit a number of industries. The unique architecture has customizable verifiers incorporated into sidechains utilizing Merkle trees. This advancement in distributed ledger infrastructure opens the door for industries to process multi-party, high frequency permissioned data at scale.

13. CONCLUSION

13.1. The Future

Based on on-chain and off-chain consensus, the network may be upgraded to continue to better serve its function of measuring business analytics. The system described in this paper aims to take the digital advertising industry, particularly programmatic advertising, to new levels of trust and efficiency for advertisers, technology vendors, publishers, and consumers.

Lucidity

The Blockchain Advertising Protocol for Complete Data Transparency

This document is available for information purposes only and does not constitute an offer for sale or any form of general solicitation or general advertising of interests in an investment vehicle managed directly or indirectly by KR8OS Inc DBA Lucidity or its officers/partners. Any such offer will only be made in compliance with applicable state and federal securities laws pursuant to offering documents which will be provided to qualified prospective investors upon request. Except as otherwise indicated, this Memorandum reflects events and conditions existing as of the date hereof. Neither the delivery of this Memorandum nor any sale made in connection with this Memorandum shall, under any circumstances, create an implication that there has been no change in the affairs of the Company after the date hereof. While the materials furnished to perspective investors include certain statements, estimates and projections of the Company with respect to the potential future performance of the Company, there can be no assurance that the Company's actual performance will meet these estimates and projections. The actual performance of the Company may be significantly and materially different from the estimates and projections, and the Company and its management make no representations herein or otherwise as to the Company's actual performance. Further, the assumptions upon which such statements, estimates and projections by management are based may prove to be incorrect. Such assumptions are inherently subject to significant uncertainties and contingencies, many of which are beyond the control of the Company and its management. **THIS DOCUMENT IS CONFIDENTIAL**