Syscoin 3.0: A Peer-to-Peer Electronic Cash System Built For Business Applications

Jagdeep Sidhu, Msc.
Syscoin Core Developer
Blockchain Foundry Inc.
Email: jsidhu@blockchainfoundry.co

***Abstract*—Syscoin 3.0 introduces a novel implementation of a decentralized marketplace, an asset infrastructure, masternodes providing bonded validators for a PoW/PoS hybrid consensus model and instant pseudo-interactive zero-confirmation cryptocurrency transactions with double-spend protection.**

1. Introduction

Syscoin 3.0 builds upon Syscoin 2.0[Sid] with additional implementation of an asset infrastructure, a zero-confirmation double-spend protected instant settlement scheme, a hardened decentralized marketplace, and masternodes providing bonded validators for a Proof of Work (PoW) / Proof of Stake (PoS) hybrid consensus model, leveraging the blockchain's immutability and audibility.

1.0.1. Z-DAG. Z-DAG (Zero-Confirmation Directed Acyclic Graph) is an instant settlement protocol functioning across all Syscoin services. Syscoin services consist of Alias Identities, Certificates, Escrow, Offers and Assets. Each service is controlled via an Alias, in which ownership is proven through a private key that matches each unique address. Z-DAG organizes transactions based on dependencies to build the state in a deterministic fashion. This helps protect against double spends where an Asset is transferred falsely by creating multiple transactions through multiple nodes within a short time.

Figure 1 shows the difference between a regular blockchain and Syscoin's Z-DAG implementation. Syscoin now has two consensus layers: In the first layer, a Z-DAG graph of transactions are represented in the mempool without a block, providing settlement in real-time. The secondary layer provides confirmation and conflict resolution preventing double-spend events through existing Proof of Work consensus.

Verifying client nodes creates a graph of transactions by looking at the sender/receiver list of Asset transfers. A circuit detection algorithm is applied (read below Hawick cycle detection) and used to remove cycles from within the graph to create a DAG (Directed Acyclic Graph). Once the DAG is created it is topologically sorted with an Asset transfer consensus code being processed in sequence, to form a deterministic state among the entire network in consensus.

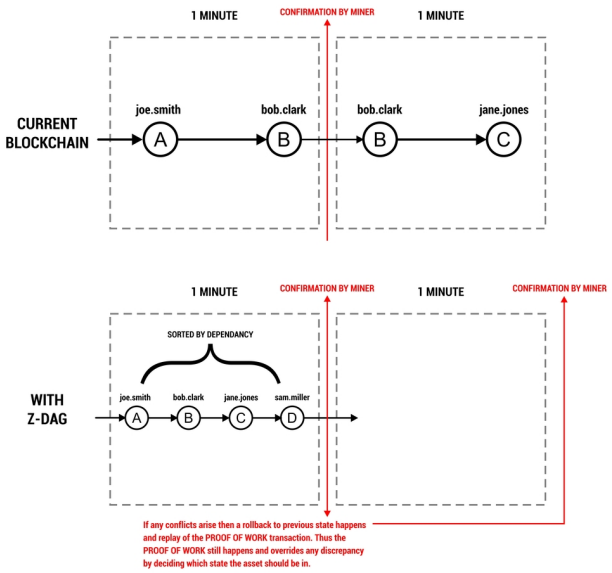


Figure 1: Current Blockchain design vs Z-DAG

Z-DAG can be applied to state changes (similar to UTXO updates to Bitcoin). If any discrepancy occurs, Proof-of-Work (PoW) will override and replay the correct order of events according to the miner; this is ensured by saving the previous state on every block and rolling back to the previous state prior to every block.

A User Interface layer will notify the user of conflicts in real-time, making the transaction occur between 3-5 seconds (statistically the amount of time needed to ensure that network takes to notice that 2 double spend transactions are conflicting with each other). This is possible through Syscoin's Masternode layer. Every Masternode is connected to 25 or more peers providing high-throughput relay across the network, averaging one or more network hops to transmit from the sender to the receiver nodes.

The partition tolerance of the protocol is the PoW block timing which is set to 60 seconds on Syscoin 3. Every block, a new DAG is constructed from the transactions within that block. Unlike other implementations that have no PoW to fall back on, there is no case where the DAG tends toward an

incorrect state over time.To accurately detect double-spends, other implementations such as Phantom[SZ] or Spectre[Zoh] must use an algorithm to replace the longest chain rule to derive the order of events and attacker sequences in a graph. These implementations end up in a more complex game-theoretical situation that has not mathematically been proven to be accurate in all cases. Syscoin relies on the thoroughly tested Nakamoto consensus model to arrive at a consensus over time rather than simply relying on a DAG.

1.0.2. Hawick cycle detection. [HJ] Hawick and James were able to detect a circuit in a graph containing edges that start and end at the same vertex, as well as multiple edges connecting the same two vertices. A similar method is implemented in the Z-DAG functionality as clients verify, extract cycles and create DAGs in order to process Asset transfers sequentially.

1.0.3. Order of events preservation and conflict resolution. A topological sort is applied upon the DAG. This gets created as a result of the Hawick detection and removal of cycles from the graph of transactions representing Asset transfers. Upon receiving a transfer, a timestamp is added to the Asset structure.

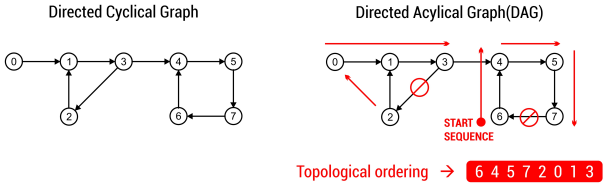


Figure 2: DAG with topological sorting

When creating a block, the miner is tasked with ordering Asset transfers from oldest to newest. There is a non-enforced 10-second delay applied to transfers, preventing Aliases from sending an Asset within the minimum latency period; this allows the transactions to arrive in order at the miner. If an Asset transfer is detected within the minimum latency period, a conflict state is set within the consensus code and users will have to wait for the next block to confirm their transfer. Asset transfers are ordered on a first come, first served basis; thus, clients choosing transaction fees on their dependent Asset transfers that are higher than transfers they depend on will not alter the order that they are put in the block.

Figure 2 shows us the typical cyclical graph that can represent the payment structure of assets between individuals. However to create a DAG we need to remove cycles and topologically sort the graph in order of dependence. In this example, 2 circuits exist and removing the edge's 3 to 2 and 7 to 6 allows us to create a DAG with a valid topological sort of 6 4 5 7 2 0 1 3.

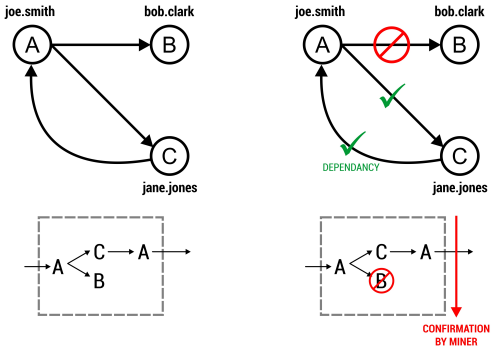


Figure 3: Conflict Resolution within Z-DAG

Figure 3 illustrates three parties, (A, B and C). A sends the same coins at the exact same time to both B and C. A part of the network would interpret the transaction as B receiving the funds, the rest of the network would see C as receiving the funds. With standard blockchain implementations, the discrepancy would be flagged as a conflicting transaction in real-time. Both parties would wait until the next block, resolving the conflict. In this case, the miner first sees the payment from A to C and the transaction from A to B is classified as a double-spend and discarded.

1.1. CAP Theorem

The CAP theorem[Bre] states that it is impossible for a distributed data store to simultaneously provide more than two out of the following three guarantees: consistency, availability or partition tolerance. Bitcoin tries to provide a guarantee that transactions are settled, but mathematically it is not able to do so. Z-DAG trades some consistency for some availability. Since it is unlikely that a double-spend or re-organization will change the state of someone's balance, we do not wait for settlement finality by waiting for blocks to confirm. This drastically increases the usability in point-of-sale applications.

The allowance of instant settlements and increased availability requires paying additional attention to users simply trying to double-spend or send transactions too quickly, which may cause the miner view to change from the general network view. The DAG will order the dependency graph of transactions and process in sequence, allowing for better availability when it comes to instant state changes. The same CAP constraints as Proof-of-Work will provide a more responsive money transfer mechanism that may be used as a transaction processor instead of simply a settlement layer.

1.1.1. Point-of-sale applications. The combination of using Assets with Z-DAG will allow for point-of-sale applications, providing service in real-time exchange for cryptocurrency Asset tokens.

1.2. Assets

Syscoin Assets provide a layer of tokenization on top of the Syscoin network. Use cases include loyalty points, coins backed by physical assets, and service backed currency. Syscoin assets can also track individual minted tokens as inputs during transfer of ownership. The creator/owner of the asset can mint and allocate tokens to other alias identities, which can then subsequently transfer ownership of tokens to other alias identities. The tokens can be divided up to a definable amount of decimal places, allowing for a flexible unit-of-account model depending on the use case.

For example, in a gold-backed asset token, one may want to track ownership of entire gold bars which are stored and audited in a vault with serial numbers. Syscoin assets can track inputs when sending tokens, so you can place ownership of the gold bars and link them to the external serial numbers through Syscoin's off-chain data-anchoring mechanism. An off-chain data file associating the token inputs to serial numbers will anchor them cryptographically to the asset object from the asset owner. A central issuer would allow redemption of the bars based on ownership of token inputs.

Generally the assets are defined with a quantity and a minimum dividable quantity (a precision field). The tracking of individual inputs and anchoring of the subsequent data model to the asset through off-chain anchoring allows for external services/assets/business practices to be distinguishable when linked to cryptographically secure and verifiable ownership of tokens.

Sending allocations of an asset involves the Z-DAG protocol to allow for real-time point-of-sale applications in a token to service model. The sender would send the receiver some tokens and receiver would in real-time detect double-spends within roughly 10 seconds. This provides a way for the receiver of a micro/low value transaction to be ensured that it is statistically likely that their funds will confirm on the network, and that the contract between sender and receiver can complete.

For efficient operation of non-fungible tokens (tokens that track ownership individually like the UTXO coin model) we developed a way to store the ownership of these tokens as input ranges. Input ranges represent a range of ownership tokens that an asset or an alias identity owns such that when transferred or received it will split and merge the ranges respectively.

$$S_A \supseteq S_B$$

Let S_A be the superset range of inputs of the sender and S_B the strict subset inputs range being sent to the receiver. Then it follows that we can derive S_C as follows:

$$S_C = S_A \setminus S_B$$

For receiver to send a set back to sender we need to prove that S_B^1 is a strict subset of S_B:

$$S_B^1 \supseteq S_B$$

In addition, we apply to S_A^1:

$$S_A^1 = S_C \cup S_B^1$$

and it follows that if S_B^1 is the same as S_B then:

$$S_A == S_A^1$$

We may apply this generally to any range sending/receiving of assets with input ranges. Please refer to Appendix D for a simple example.

1.3. Offers and decentralized marketplace

We have developed a marketplace where users can securely and reliably buy and sell a variety of items. Syscoin users will be able to directly create and manage their online stores through the web-portal or offline on their desktop or mobile wallets.

Users will be able to view listing descriptions, price, geo-location based listings and services (if enabled by the merchant), seller's profiles, and reputation.

Availability of merchants will be filterable in the context of searching for offers. It will also allow for buyers to quickly get in contact with merchants and acquire support for a product in real-time prior to purchase.

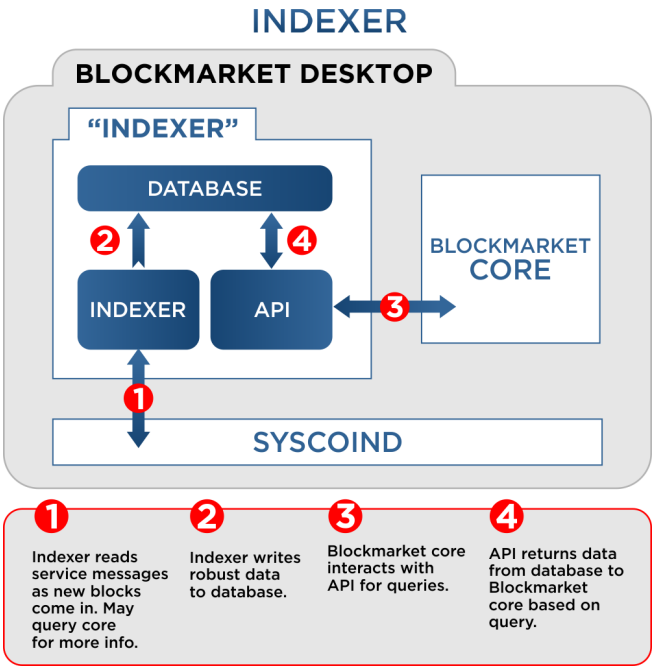


Figure 4: Syscoin Service Indexer

Sorting and filtering will be provided by supplementary off-chain databases indexed by the Syscoin core through ZeroMQ messages as it sees offer transactions in real-time. This will allow for greater interactivity with data than previous implementations which relied on the blockchain reference software for storage and querying/filtering.

1.3.1. Digital sales. Certificates may be sold in conjunction with offers to create sales of digital ownership. A certificate may hold private information such as codes or registration keys that are redeemed for some service by the buyer of the offer. Certificates can be automatically transferred to the buyer upon completion of sale.

1.3.2. Auctions. Offers on the Syscoin platform may be created as an auction with a fixed countdown time, minimum bid and reserve price. Offers can now be Regular Offers, Timed Auctions or Timed Auctions with a Buy it Now option.

In our auction system, we allow merchants to configure multiple options. They may require a deposit (this imitates how current Real-Estate transactions work, a deposit is often required to show earnestness), a witness or notary, and a reserve price. The escrow system works in conjunction with the offer and identity modules to ensure the correct offer is sold to the winning bidder under authorization of the seller. Because these are smart contracts running through consensus on the blockchain, auctions are just as secure as regular transactions.

1.3.3. Reselling with whitelists. Merchants may leverage a whitelist feature to offer re-sellers the chance to sell their offers for a commission. This allows drop-shipping of goods and services while offering provable sales through the decentralized marketplace. The merchant who created the offer controls the whitelist and can add a discount level on a per entry basis for each re-seller. If the merchant sets their offer to private, then end users must purchase the item through one of the participating re-seller offers.

1.3.4. Feedback and rating system. Escrows and offers sold through the marketplace offer a convenient way to rate and leave feedback on a per sale basis. For an escrow, one rating is accepted (a number from 1 to 5) to represent a users satisfaction level with a transaction, with 1 being the least satisfactory and 5 being completely satisfied and recommending the user to others. Ratings and feedback can be given to and from arbiters, merchants and buyers.

1.3.5. Multiple payment options. Syscoin offers the direct conversion, transfer and ownership of hundreds of cryptocurrencies today. The Syscoin marketplace will allow offer payment via the complete set of coins by adding an abstraction layer converting coins to Syscoin upon purchase and locking the Syscoin in escrow as part of the normal offer purchase flow. Native support can be for any coin that supports the same private key and signature scheme format as Syscoin does; this allows client-side conversion from the native Syscoin address to the desired address such that the same private key can unlock payments in the other supporting blockchains where the private key can be used to sign off on spending those coins. Direct support for Syscoin and other Assets will work differently. Instead of sending money to an address, an Asset transfer would happen between identities since Asset transfers use the identity as input instead of coins.

1.3.6. Shipping notification system. A payment acknowledgement button on escrow and offer payments allows a multi-use notification system to the buyer that either the merchant acknowledges payment and/or they are about to ship the product. Tracking and other shipment information can then be sent via the encrypted messaging system.

1.3.7. Marketplace moderation and Private Offers. Marketplace moderation is done through the safe search feature, allowing blacklisting of offers that are in obvious violation of an ethical and moral code of conduct. Merchants may allow one-on-one deals by setting the offer to private.

1.3.8. Cryptographic security through blockchain unique user identities. Any Syscoin user that updates his offer listings or digital certificates must sign an input of their blockchain anchored identity. This is a cryptographically secure means to ensure the provable ownership and modifications of those services. Consensus code at the blockchain layer ensures the proper identity signature matches that of the public address associated with the authorizing identity.

We have applied domain-name like rules to user identities, allowing only unique case-insensitive names. Users are now able to send coins and encrypted messages to an alias using any case formatting desired; the recipient will always be the user who owns the lowercase version of the alias.

Within the identity context, users may post public and private details including social profiles, profile pictures and other information that users may want others to see.

1.4. Data anchoring

The diagram below depicts an innovative way to have a blockchain based anchoring mechanism that authenticates to your blockchain identity. Anchoring is a way to reduce the footprint of data on the blockchain yet at the same time allow for the cryptographic verification of that information linked to your blockchain based identity.

Figure 5: Data anchoring infographic

The client attempting to write the first data signs the payload using their private key; this establishes that they control the piece of data which they are trying to modify. Next, the raw payload, signature and public address are sent to the server. The server uses a local independent connection to a full node to validate the signature based on the address submitted. If the signature is valid then the write operation is executed, if not then the write is prevented and an error is returned to the client.

1.4.1. Identity specification. By storing identity data off-chain with a blockchain-anchor, we can increase the amount of data users are able to store within their Syscoin Identity without bloating the blockchain. By storing these entities off-chain we are also able to evolve the specification for Identities without having to fork the network, another costly piece of overhead if we were to store the actual data on chain.

```
export interface SyscoinIdentity {
  publicIdentity: SyscoinPublicIdentity;
  privateIdentity?: SyscoinPrivateIdentity;
  encryptedPrivateIdentity?: string;
}
```

1.4.2. Public Identity specification. We're starting with what most modern-day platforms attach to an identity simply for the purposes of enhancing the current Blockmarket (Syscoin desktop and web wallet) experience and making marketplace interaction even easier. Some of the items you can store in your Syscoin Identity include an avatar image URL, first name, last name, Facebook URL, Twitter URL, PGP public key, bio, and more.

```
export interface SyscoinOnChainIdentity {
  avatarUrls: string[];
  firstName: string;
  lastName: string;
}

export interface SyscoinPublicIdentity
extends SyscoinOnChainIdentity {
  location?: string;
  pgpKey?: string;
  bio?: string;
  facebookUrl?: string;
  twitterUrl?: string;
  instagramUrl?: string;
  bitcointalkUser?: string;
  trustedArbiterNames?: string[];
  requireTrustedArbiter?: boolean;
}
```

1.4.3. Private Identity specification. Identities also feature a private data field which is secured against public access using ECIES encryption. Properties like shipping address and PGP private key can be stored in this section of the identity with confidence of data safety because it is stored encrypted and can only be decrypted with the owners alias and password.

```
export interface SyscoinPrivateIdentity {
  shippingAddress?: string;
  pgpPrivKey?: string;
}
```

1.5. Instant Encrypted Messages

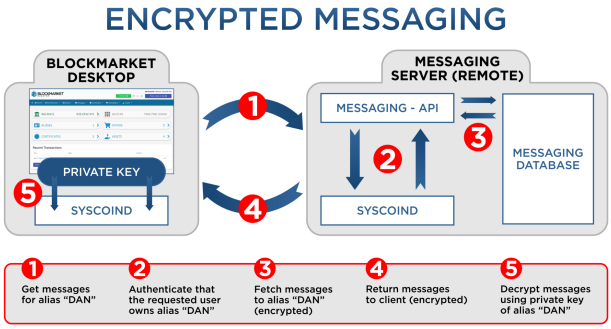


Figure 6: Authenticated encrypted messaging service

Since users of the marketplace operate through a blockchain unique identity, an encrypted messaging service is easily plausible such that both sender and receiver are authenticated providing cryptographic certainty that User A received a message from User B and only User A can decrypt and read that message using his unique Syscoin identity. Encrypted messages are secured with the Elliptic Curve Integrated Encryption Scheme (ECIES) symmetric key negotiation from public keys to encrypt arbitrarily long messages. Multiparty encryption is also possible through the use of multi-signature identities. Messages are not stored on the blockchain. However, blockchain based identities are used to decrypt and read these off-chain messages as well as cryptographically verify who the sender is.

1.6. Blockchain pruning

To combat blockchain bloat, the underlying infrastructure implements a novel pruning mechanism to remove marketplace listings and other unused data based on the expiration of Syscoin user identities. If a user is inactive, their data will be removed. The data on the blockchain thus represents active users using the Syscoin marketplace. The data for the service is stored in the OPRETURN output and not hashed to the transaction and thus can be safely removed upon expiry of the connected Syscoin Identity. To better understand the technical nature of pruning, please refer to the Syscoin 2 whitepaper.

Syscoin assets are not prunable as they are considered money and are placed under the same policy as normal coins.

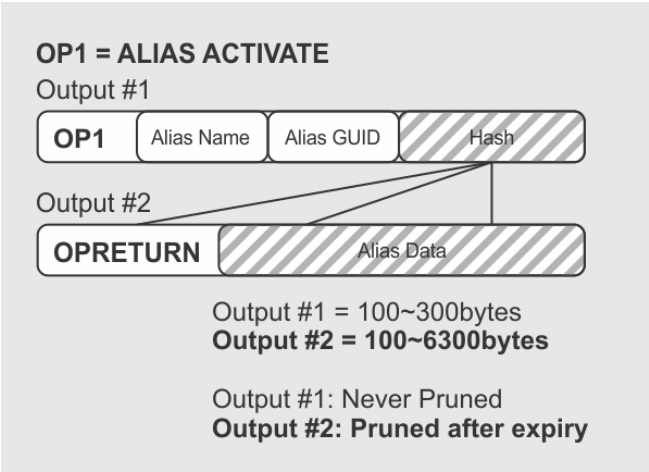


Figure 7: Blockchain pruning

1.6.1. Zero-knowledge alias authentication. Alias private keys can be generated deterministically by supplying a password hashed with a generated random 256bit number known as ”password salt”. Password salt is stored in local storage on the browser and offline. When the user logs in, their private key can be derived using their password and locally stored salt, then checked against the public address of the alias through an identity information lookup from within any full node. This enables wallet-less on-chain controls and authenticated spending of coins/services without requiring a transfer of credentials over any network.

1.6.2. Expiration. Identities expiry happens based on time. The blockchain protocol acts as a decentralized time server which stamps blocks based on height and time. Services expire when the identity related to it expire and escrow will expire if, and only if the arbiter, buyer and, seller identities involved are all expired.

1.7. Certificates

Digital certificates within Syscoin infrastructure are useful for many kinds of applications - from storing bits of data to creating data that may be sold and transferred upon purchase - all with provable ownership via the blockchain.

1.7.1. Public and private data. Certificates, like aliases, have public and private data. Private data can be accessed by foreign aliases either through creating a multi-signature alias and including other aliases or by transferring ownership of the certificate to the new owner.

Using a multi-signature approach allows certificate owners to maintain control of their certificates while still allowing decryption of private data by other users. For example, an owner could change the alias of the certificate to a multi-signature alias, then assign two aliases owned by the owner and one alias owned by another party. By requiring two of three signatures, the owner would be able to edit the data, the third party would not.

1.7.2. Transfer of ownership. Certificates can be transferred to other identities. New owners will receive reading rights for any private encrypted data and the transfer can be configured to allow editing of certificates upon completion.

1.8. Escrow

Syscoin’s integrated escrow service allows safer payments of offers by securely holding a buyer’s tokens in escrow until the terms of the sale are met and the buyer releases payment to the seller.

The system uses an arbiter-based system, whereby arbiters act as trusted third-parties between buyers and merchant for a sale in the decentralized marketplace. An arbiter is paid based on a dynamic fee set in the rates peg for the offer that is sold. Once the escrow process is complete all parties can be rated and given feedback related to the sale.

Arbiters are chosen by buyers when accepting an offer. Typically the buyer and seller would agree on an arbiter before an offer is accepted. In most cases no dispute is filed and no arbitration is needed.

If a merchant does not ship goods, the arbiter refunds the buyer. If the buyer receives goods as described but does not release payment, the arbiter releases funds to the merchant. The feedback and rating system helps prevent irrational behavior by aligning incentives allowing actors to benefit from honest actions.

The fees paid to arbiters are only applicable if the arbiter acts to sign off on a refund or release payment to merchant. The fees are set by the buyer upon purchase. These fees are adjustable, but the fee rates requested by the escrow agent should be taken into consideration when buyer adjusts the default fee rates when purchasing. Escrow agents with better reputation scores and more transactions arbitrated can charge higher fees as a result. However, a market equilibrium will present itself between the supply and demand of escrow agents and their fees.

Once an escrow is created, the users involved are provided specific controls based on their role. These controls allow the user to initiate specific functions on the escrow through the user-interface.

2. Open-API Specification

We will develop syscoin-api 3.0, an Open-API compatible specification for software developers. Syscoin-api provides a clear and concise solution and is provided in the language of their choice. With syscoin-api, developers can easily build custom blockchain applications.

3. Masternodes

Masternodes are a P2P extension of the Syscoin protocol; allowing decentralized governance, decentralized mixing, instant transactions and Z-DAG. The Syscoin masternode protocol is similar to the Dash crytocurrency’s implementation, the first to introduce the masternode technology[Duf]. We have further innovated this protocol by

providing a 50/50 split of transaction fees to miners and masternodes. A seniority model has been created to provide larger incentives to masternode hosts. The seniority model was created after in-depth analysis of masternode up-time behavior; supplying the network with a more efficient mechanism.

For additional information about the specific Dash features that Syscoin inherits, you may refer to the Dash whitepaper. We will be focusing on the innovations done on Syscoin pertaining to masternodes and how else we provide utility using the masternode layer.

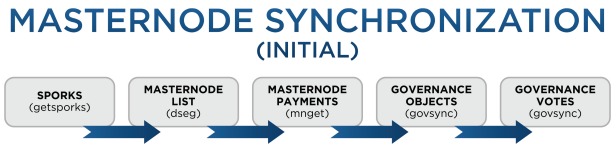


Figure 8: Masternode P2P Protocol Basics

Refer to Appendix A for flow diagrams of how the P2P layer functions and what messages are being passed for the correct functioning of masternodes.

3.0.1. Mechanism design through masternode seniority. Bitcoin provides two incentives for miners: block subsidies (through rewards) and transaction fees. As Bitcoin rewards wind down the network could become unstable due to degrading miner incentive to act in the best interest of network security. Issues such as selfish mining and undercutting are discussed in greater length in a paper presented at the ACM CCS [Nar16]. To prevent mining incentive from degrading, usage of services and length of bonding are tied to an inflation metric for block rewards. Transaction fees remain to provide incentive to mine and relay transactions but rewards depend on the demand for using the Syscoin network. Bonded validators serve network utility in exchange for a mining reward adjusted based on seniority. The longer the term of the bond contract the more deflation-adjusted return on investment, the user would make up to a maximum rate. As utility increases and bond holders remain, a seniority metric slightly adjusts the supply base for the increased incentives. For every 4 months of blocks that one bonds coins and provides masternode services, a 3 percent increase in ROI - on average - is accounted for in the mining reward. This can carry forward for up to 3 years in blocks or 27 percent maximum deflation-adjusted ROI.

3.1. Syscoin Governance

Syscoins governance ensures that there is a decentralized, unbiased system to manage and fund the platform. Unlike previous not-for-profit foundations that were tasked to maintain core protocol and promotion of the coin, Syscoin governance is administered by masternode operators that are heavily invested in the future of Syscoin. These masternodes are the most dependant to date, as they, unlike miners, cannot reuse their asset for any other coin or purpose.

Syscoins decentralized governance system powered by masternodes, ensures that the promotion and betterment of Syscoin is their number one focus. Masternodes must vote through a transparent public portal, on the development and expansion of the Syscoin network, and budgeted funds are allocated accordingly. Each project that gets voted on and is passed, is taken from the total budget and paid directly to the person who won the proposal to carry out the proposed project. The payment is paid out in a decentralized fashion, using a superblock, once a month. A superblock receives 10 percent of each block reward saved into a monthly budget for proposals. Governance promotes a fast turnaround and essentially indicates an approval to proceed on that specific task.

To ensure long-term stability, allocating funds through Syscoin ensures masternode operators become contractors who work for, and are compensated through the network. A portion of the block reward is held in escrow, in the names of the operators, and not released until voting is complete. This maintains the promotion of the coin and the network, as operators who sell their coins can be picked up by more interested individuals who will then acquire the right to vote. This system allows for the network to be capable of sustaining itself, maintaining growth and allowing for appropriate adjustments and changes which are not dependent on specific operators.

4. Specs

General specifications for Syscoin 3.0:

- Max Coins: 888 million
- Deflation: 5 percent per year until Max Coins
- Consensus: PoW/PoS Hybrid. PoW is SHA256 Merge-mined with Bitcoin
- Block time: 60 seconds target
- Rewards: 38.5 Syscoin deflated 5 percent per year of which 10 percent is allocated to governance proposals (3.85 Syscoin). 75 percent of the result goes to masternode and 25 percent to miner.
- Difficulty algorithm: Dark Gravity Wave
- Masternode collateral requirement: 100000 Syscoin
- Masternode seniority: 3 percent every 4 months until 27 percent over 3 years
- Governance proposals payout schedule: every month
- Governance funding per round (168630 Syscoin per month)

5. Future work

We will work with the Blockchain Foundry team to innovate and bring value to the token holders of Syscoin in a variety of ways, including scaling and better third-party escrow solutions for trustless e-commerce.

5.1. Encrypted Messaging enhancements

With the new messaging system, it is now possible to implement optional email notifications for specific events on Blockmarket such as notification of a sale or an escrow request. We may also add image support, enhanced (HTML) messages, attachments or even support direct phone calls and text messages. You may also export messages to a file for safekeeping.

5.2. Lightning networks

We are looking to develop an off-chain transaction mechanism whereby we are able to provably move Syscoin Assets in high volume without fees and without affecting blockchain bloat.[Fyo]

5.3. Offers/Escrow

Proof-of-shipment is something we have innovated and are expanding upon the shipping notification system from within the escrow and offer service layers. A video can be taken by the merchant, hashed and included in a data-field from within the shipping notification transaction, assuring arbiters and buyers that any disputes would be quickly and efficiently resolved. Arbitration and insurance would become cost-effective means to insure true buyer protection as markets form as a result of the technology. There is currently a proof-of-concept under development for this proof-of-shipment mechanism.

We are working on the ability for autonomous agents to act as escrow agents and deliver goods to buyers within 5km of distribution centers. The food industry employs such services. We would simply adapt these to allow for a decentralized marketplace with tokens held in escrow that would be released upon acceptance of delivery.

5.4. Syscoin 4.0 - Next Generation

The next generation of Syscoin will focus on scalability. Work has begun on creating a platform for enterprise grade applications done in a permissionless blockchain environment. A public blockchain that serves as the backbone for a merkle computer[Rou] is the ideal case for subsequent economic development to rely upon without central actors to dissuade others from performing for the most efficient common goals for an economy. The ideal system would allow for an EVM implementation to run such that the Patricia Trie states[Fou]. The accounts would be stored in a scalable, permissionless blockchain running on masternode technology. They would also be performing sidechain Syscoin transactions to get in and out of the EVM environment through the help of Intel SGX or merge-mined sidechains. The computation would be run either as a quorum or with the help of hardware protected execution environments, so that single random or dual random nodes are all that would be needed to perform calculations for an EVM. Non-turing complete DApps can be done through a merkle-tree timestamping service, where many pseudo-transactions are aggregated into large merkle-trees and root-hash stored onto the Syscoin blockchain at set time intervals. Storage of the trees and the individual DApp state contracts can be stored on any content addressable storage system like IPFS. This would allow for an unbounded scalable system that benefits from the immutable source of truth that the blockchain provides. The idea is to separate the blockchain courts from the merkle computers. Computation and storage of the contracts can and should be done off-chain in content addressable means. The consensus critical hash of the data can be stored in the court for provable auditing purposes.

6. Conclusion

Acknowledgments

We would like to thank Satoshi Nakamoto, the Bitcoin Core developers, and the Dash core developers for their continued excellence in software engineering, which has made it possible for others to develop innovative products on top of their accomplishments.

References

[Nar16] Arvind Narayanan. "On the Instability of Bitcoin Without the Block Reward". In: *ACM CCS* (2016). DOI: https://www.cs.princeton.edu/~smattw/CKWN-CCS16.pdf.

[Bre] Eric Brewer. *Brewer's theorem*. URL: https://en.wikipedia.org/wiki/CAP_theorem.

[Duf] Evan Duffield. *Dash: A Privacy-Centric Crypto-Currency*. URL: https://github.com/dashpay/dash/wiki/Whitepaper.

[Fou] Ethereum Foundation. *Modified Merkle Patricia Trie Specification (also Merkle Patricia Tree)*. URL: https://github.com/ethereum/wiki/wiki/Patricia-Tree.

[Fyo] Jonald Fyookball. *Mathematical Proof That the Lightning Network Cannot Be a Decentralized Bitcoin Scaling Solution*. URL: https://medium.com/@jonaldfyookball/mathematical-proof-that-the-lightning-network-cannot-be-a-decentralized-bitcoin-scaling-solution-1b8147650800.

[HJ] K. A. Hawick and H. A. James. *Enumerating Circuits and Loops in Graphs with Self-Arcs and Multiple-Arcs*. URL: https://blog.mister-muffin.de/2012/07/04/enumerating-elementary-circuits-of-a-directed_graph/.

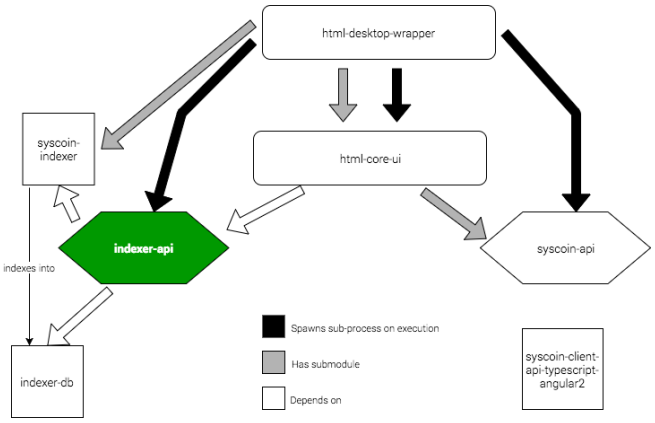
[Rou] Simon de la Rouviere. *Interplanetary Linked Computing: Separating Merkle Computing from Blockchain Computational Courts*. URL: https://media.consensys.net/interplanetary-linked-computing-separating-merkle-computing-from-blockchain-computational-courts-1ade201ecf8a.

[Sid] Jagdeep Sidhu. *Syscoin: A Peer-to-Peer Electronic Cash System with Blockchain-Based Services for E-Business*. URL: http://syscoin.org/whitepaper.pdf.

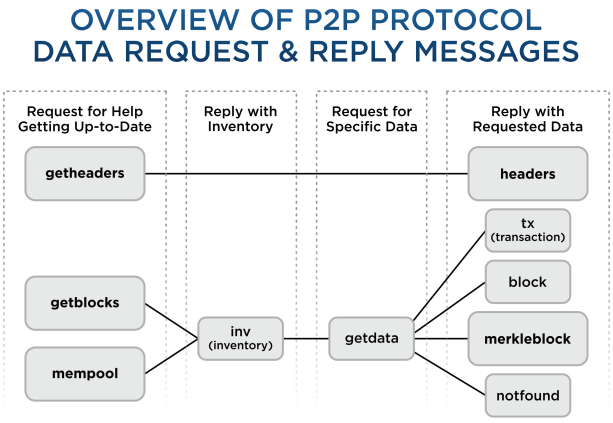
[SZ] Yonatan Sompolinsky and Aviv Zohar. *SA Scalable BlockDAG protocol*. URL: https://eprint.iacr.org/2018/104.pdf.

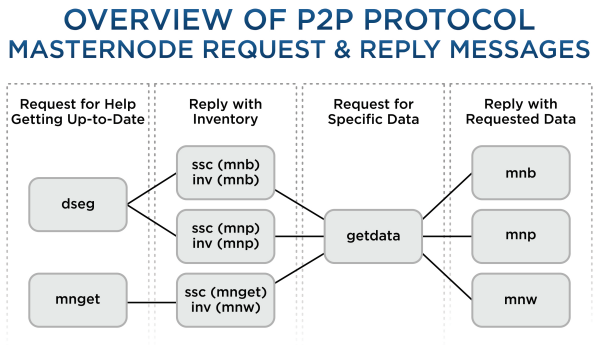
[Zoh] Aviv Zohar. *Serialization of Proof-of-work Events: Confirming Transactions via Recursive Elections*. URL: https://eprint.iacr.org/2016/1159.pdf.

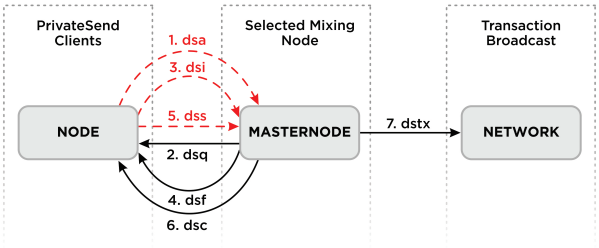
7. Appendix A: Syscoin Indexer Dataflow



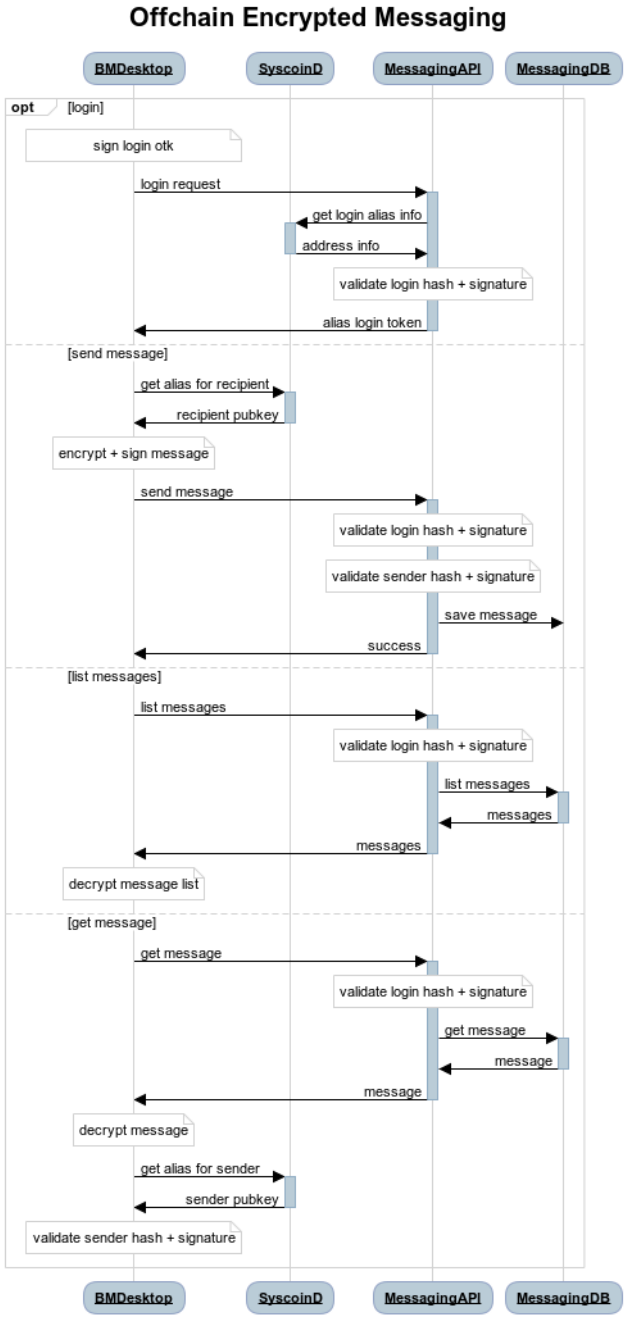
8. Appendix B: Masternode P2P Dataflow







9. Appendix C: Encrypted Messaging Sequence



10. Appendix D: Asset Input Range Sending