KAPPI.network

Hybrid blockchains

support@kappi.network

Introduction

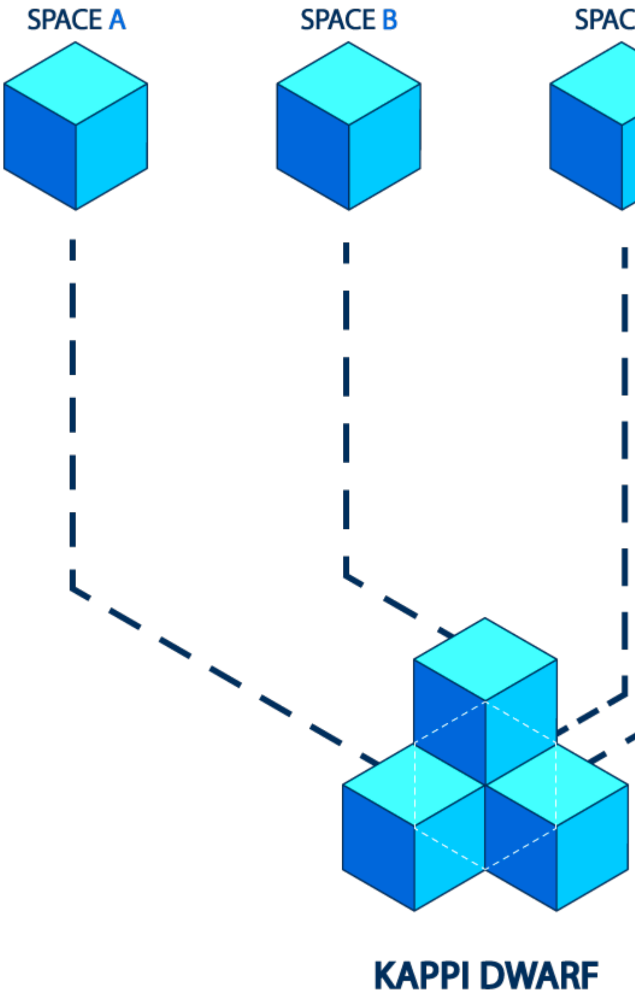
Creating success through the use of decentralized file-sharing in an open-source ecosystem to run a public cryptocurrency has allowed more people to understand how such infrastructure can improve basic social economics. Bitcoin, and Zero cash are two examples of specialized blockchain applications, and the best example of a smart contract platform would still be Ethereum, which allows for many applications of the Ethereum Virtual Machine (EVM). There have been drawbacks to the different blockchains that have been created thus far, such as the lack of energy efficiency, lack of any type of well thought out governance mechanisms and limited or poorly performing blockchains. Scalability was not thought of when Bitcoin was incepted, and now there are proposals to create scalability throughout Bitcoin's transaction process.

These include Bitcoin and Segregated-Witness, which are both vertical scaling models which are still bound by a lack of capacity within any single machine. This approach is necessary for pre-existing blockchains that didn't consider their scalability due to the requirement to maintain a complete auditability. There is also the Lighting Network, which could be used instead of alternative options to bring scalability to Bitcoin transaction volume. This would be accomplished in the Lightning Network by reducing the number of transactions that are on the ledger. This approach is suitable for privacy-preserving payment rails and for micropayments but might not suit more generalized scaling requirements.

Multiple blockchains running in parallel could use this solution advantageously, allowing for interoperation and still keeping all security properties of the blockchain. Proof-of-work approaches make this all but impossible. For merged mining, work is allowed to be done for the securing of a parent chain to be used again on a child chain but does not take away the need for each individual transaction to be validated by each node in order. Merged mined blockchains are also open to attack from a hashtag power majority on the parent chain, where there is no active merging of the child chain. This is where KAPPI comes in. KAPPI is a unique blockchain which has network architecture that addresses each of the above problems.

KAPPI works as a network of several different independent blockchains, which are called spaces. Each of these spaces are powered through a KAPPI DWARF, ensuring that there are a consistent, high-performing and secure PBFT similar consensus engine wherein the accountability is guaranteed through forks. The KAPPI algorithm is scalable and can be used for proof of stake, public blockchains. Within the fist space on KAPPI is the KAPPI DWARF. The KAPPI is a cryptocurrency that operates a multi-asset proof of stake and has a simple governance system that allows for upgrades and is generally adaptable. The KAPPI DWARF can connect to other spaces allowing it to be extended.

The spaces and the DWARF of the KAPPI network communicate between themselves through an inter-blockchain protocol, which is like a virtual UDP for blockchains. This makes it easy to transfer tokens between zones in a fast and secure way without the requirement of exchanging liquidity between each zone. In this approach, the tokens that are transferred between zones will go through the KAPPI DWARF that keeps a running tally of all tokens within each space. The KAPPI DWARF acts like a surge protector that isolates each zone from any failure experienced in any other zone. Anyone can connect new zones to the KAPPI DWARF, the spaces are capable of being compatible with blockchain innovations in the future.



KAPPI Consensus Protocol

The KAPPI consensus protocol and the interface which was used to build the application is backed by nodes which retain voting power which is no-negative. This is different than the classic Byzantine Fault Tolerant (BFT) algorithm that has each node carrying the same weight. In the KAPPI consensus, validators are able to participate in the consensus protocol through broadcasting cryptographic based signatures, which are referred to as votes, that agree on the next block. The voting power of each validator is determined at genesis and can be altered deterministically within the blockchain, which would be dependent on what the application is. An example of this is that the staking tokens can be bonded as collateral to determine the voting power. Unless all the validators have an equal weight, fractions would not be used to refer to the validator but only for the total voting power.

KAPPI runs as a partly synchronous BFT consensus protocol that was created from the DLS consensus algorithm. KAPPI is known for being simple, working at a high level, and for the ease which fork-accountability creates. The protocol only needs a fixed known set of validators that are all individually identified by their public key. The validators will work at gaining consensus on a block by block basis, where each block contains a list of transactions. Block consensus voting is done in rounds, where each round will have a proposer, or a round-leader, which proposes a block. Validators will then vote in trenches to decide whether or not each proposed block will be accepted. The round proposer will be chosen in a deterministic way based on the validator order list and always in proportion to their voting power.

The security on KAPPI is created through using an optimized Byzantine fault-tolerance through super majority voting as well as a locking mechanism. Combined they act to formalize that one third of the voting power must be through Byzantine for a safety violation to be caused, any time more than two values are committed. Whenever a set of validators either attempt to or succeeds in violating safety, they will be identified through the protocol. This protocol is inclusive of voting for conflicting blocks and trying to broadcast an unjustified vote. KAPPI is able to process thousands of transactions per second and can order latencies within two seconds. This allows for over a thousand transactions a second to be performed constantly, even where there are strong adversarial conditions such as maliciously broadcasting votes or crashing validators.

One of the benefits to the KAPPI consensus algorithm is its client security, allowing a versatility that can be used for any of the internet of things or mobile applications. Whereas a client of Bitcoin has to sync chains of block headers to find the one with the highest proof of work, the KAPPI clients only have to follow the validator set changes and verify any PreCommits within the newest block to create the latest state. Proofs in KAPPI also allow for inter-blockchain communication. Proactive steps are included in KAPPI that prevents specific notable attacks like censorship or double spends. The KAPPI consensus algorithm is opened in a program called KAPPI, which is an application-agnostic consensus engine that turns any black box, deterministic application into a replicated distributed blockchain. The KAPPI DWARF connects to other blockchain applications through the Application Blockchain Interface, which can be programmed with any language regardless of the language used in the consensus engine. The application blockchain interface also allows for an easy changeover of any existing stack.

An analogy can be made with Bitcoin, which is known as a cryptocurrency blockchain that operates by having each node carrying a fully audited Unspent Transaction Output (UTXO) database. If someone wished to create that type of system on an ABCI foundation the KAPPI DWARF would have the responsibility to share transactions and blocks between each node. This would establish an immutable or canonical order to the transactions in the blockchain. The ABCI application would maintain the responsibility for:

- The UTXO database.
- Validating all cryptographic signatures within a transaction.
- Stopping transactions from spending funds which do not exist.
- Allowing client query through the UTXO database.

KAPPI is capable of decomposing the blockchain design through offering an easy API which works between the consensus process and the application process. KAPPI consists of a network of parallel blockchains that are all independently powered by classic BFT consensus algorithms like KAPPI. The first blockchain in the network is the X DWARF, which connects to multiple other zones or blockchains through a unique inter-blockchain communication protocol. The KAPPI DWARF is able to track many types of tokens and keeps a record of all the tokens in each zone that is connected. Tokens are able to be transferred from one zone to another zone quickly and securely without requiring any physical liquid exchange between the zones. This is made possible due to the fact that all coin transfers undertaken inter-zone will all go through the KAPPI DWARF, which acts as a solution to many of the issues facing blockchain in contemporary time.

Several problems are solved using this architecture, including scalability issues, interoperability and seamless upgradability. Regardless of where the zone derived, including if it was created on Bitcoin, CryptoNote, Go-Ethereum or 7Cash, among others, can plug into the KAPPI DWARF. This allows for infinite scalability into the future and shows that these zones are perfect for operating a distributed exchange. KAPPI isn't just a single distributed ledger, but instead creates a protocol for any open network of distributed exchange. Transparency, sound economics, accountability and consensus theory are all encompassed. KAPPI validators operate much the same as a miner would in Bitcoin, except that it uses cryptographic signatures for voting. KAPPI validators are dedicated and secure and responsible for the commitment of blocks. A non-validator can delegate staking tokens, referred to in the KAPPI as atoms, to any validator and earn a part of the atom rewards and block fees.

If the validator is hacked or is seen to violate the protocol in any way, it will be slashed, which ensures a safety guarantee of KAPPI BFT consensus, and the overall deposit of validator and delegator stakeholders and ensure no violation to the protocol. This provides security that is quantifiable for both larger clients and for small nodes. Each zone is also able to have its own governance and constitution mechanism, such as for dealing with subjects like rollbacks. This interoperability between different policy zones gives users freedom and the ability to undertake permissionless style experimentation.

Scalability and Decentralization

KAPPI is comprised of a network of many blockchains that being powered by KAPPI. KAPPI allows many blockchains to be running concurrently with each other whilst retaining interoperability. At its DWARF, KAPPI DWARF manages multiple independent blockchain 'zones', that are also referred to by some as shards. With a constant stream of block commits coming from zones on the DWARF, it can keep up with each zone's information and its current state. In turn, the zones keep up with the DWARF, but not each other except through the DWARF. Information packets are sent from one zone to another through the DWARF through Merkle-proof posting showing that the information was both sent and received accurately.

Due to the inter-blockchain communication, any zone can be a DWARF for the purpose of forming an acrylic graph, if desired. The KAPPI DWARF blockchain consists of a multiple asset distribution ledger with tokens being individually used or used within a zone itself. Tokens can be moved between zones through a DWARF responsible for preserving any global invariance of the total token value across each zone. Sender, receiver, or DWARF blockchains can commit IBC coin packet transactions. The KAPPI DWARF is the central ledger for the entire system and its security is of primary importance. Each zone can have a KAPPI blockchain that is secured by no less than 4 (or less if not using the BFT consensus) and is secured by a set of validators that are globally decentralized to serve as being strong enough to stop any type of hack or attack scenario.

The KAPPI zone constitutes an individual blockchain which exchanges IBC messages to the DWARF. The DWARF would conceive a zone to be a multiple asset membership dynamic with a multiple signature account capable of sending and receiving tokens through IBC packets. Like any cryptocurrency account, zones cannot transfer a token if they do not have that token to send but are able to receive tokens. Zones can use one or more types of tokens, which gives it the ability to inflate token supplies. Atoms of the KAPPI DWARF can be staked by any validator of a zone which is connected to the DWARF. This could result in a double spend attack, but it would be slashed through the KAPPI fork accountability, a zone where voting power cannot create any invalid state. KAPPI DWARF will not execute or verify any transaction that is committed in another DWARF, so users must send tokens to trustworthy zones.

Inter-zone Communication

Using an example, let's say there are three blockchains, one of which is the DWARF. We want to produce a packet that is destined to arrive at one of the 2 non-DWARF zones. For a packet to be moved between blockchains it is first posted on the receiving chain; the proof will state that a packet was published by the sending chain for the destination. For this proof to be checked by the receiving chain it has to keep speed with the sender's block headers. This is quite similar to sidechains that require interacting chains to be aware of each other through bidirectional chains through the use of datagram proof of existence transactions. The IBC protocol is able to define two types of transactions, a packet which facilitates the blockchain proof to observers of the last block hash and one that allows the blockchain to prove that the sender's application published any given packet through the Merkle-proof. Since these mechanics are split into different transactions there is an allowance for the native fee mechanism of the receiver chain to determine what packet is acknowledged whilst allowing freedom for the sending chain to send any number of outbound packets.

These exchanges are not as vulnerable to internal or external hacks since it is a distributed exchange. Common decentralized exchanges are capable of running because of atomic cross chain (AXC) transactions; where users on different chains can make different transactions that are committed together on both ledgers, and atomically on no ledger. The benefit of AXC transactions is that trust does not need to be present between users, but both parties will have to be online to finish the transaction. Mass replicated distributed exchanges that run on their own blockchain are another type of decentralized exchange. This method allows the user to turn their computer off and know that the order will still be executed without their need to be online. A centralized style of exchange has the potential to make large orderbooks and as such will attract more trades. Poloniex is the current cryptocurrency exchange with the most popularity, followed by Bitifnex. AXC based exchanges are unlikely to win over such strong network effects.

It is necessary for decentralized exchanges to support large orderbooks and institute limit orders for the exchange to compete with one that is centralized. This can be accomplished by integrating a distributed exchange on blockchain. KAPPI brings further benefits to this scenario by being consistently fast, able to finish transactions either in IBC token transfers or in exchange order transactions as fast as is possible with modern technology. Transaction throughput capacity in KAPPI is comparable to centralized exchanges. Limit orders can be submitted by traders that can be finished without the need for both parties, and either party, to be online. Transfers across zones are also accomplished in a comparably fast way.

Bridge Zones

A bridge is what the relationship between the DWARF and the zone is called. Both have to keep up to date information on the other's blocks for the purpose of verifying proofs when tokens move between the two. The bridge zone has an indirection which lets the DWARF logic to stay agnostic and simple to the other blockchain's consensus algorithm strategies. Every bridge zone validator will operate an KAPPI powered blockchain that includes an ABCI bridge app and a full node of the original blockchain. As new blocks are mined the bridge zone validators reach agreement on committed blocks through signing and sharing each perspective view of the blockchain tip origin. When payment is passed through a bridge zone on origin and there have been enough confirmations a corresponding account is created on the bridge with that balance. The bridge zone can share validator sets on networks such as Ethereum.

Bitcoin has a comparable concept with the exception of each UTXO being controlled by a multiple signature pubscript instead of a single bridge contact. Ether can be transferred on the bridge zone between the DWARF and will be destroyed later on when it is sent to a specific Ethereum address. The bridge zone will then have an IBC packet that can be sent through the bridge, which allows the Ether to be withdrawn. Restricted scripting is hard to mirror in an IBC transfer mechanism. UTXOs can be compressed and decompressed when necessary to keep the overall UTXO number down, which works will since each has its own pubscript. Rogue validator sets are risky where there are bridging contracts. Forks can be caused by Byzantine voting power when ether is withdrawn from the bridge contract when it is kept on the bridge zone. Ether can be stolen through Byzantine voting power by having a deviation to the original bridge zone bridging logic.

This can be addressed through the design of the bridge, by requiring acknowledgement on the bridge zone that allows for all transactions to be challenged and verified either by the bridge contract or by the DWARF. The relationship between the DWARF and the origin should be such that it allows the bridge zone validators to post collateral by either the bridge zone or the DWARF. The origin and the DWARF should also allow token transfers out but have enough time to allow for all transfers to be verified. KAPPI can commit blocks faster than Ethereum's proof of work, with KAPPI consensus and bridged ether operations offering a better performance than blockchains created on Ethereum. The KAPPI DWARF doesn't allow for any contract logic execution that is arbitrary but can be used to create the token movements between different zones, creating a scalable and shard-able token focused Ethereum. The KAPPI zone operates random application logic that is defined when the zone is created and has the ability to be updated over time.

This type of multiple ability lets KAPPI zones become bridges to various cryptocurrencies and allows derivatives to be created of those blockchains, using the same codebase and integrating a different initial distribution and validator set. This creates the ability to link to other frameworks using the KAPPI engine as a common network. Within a multiple asset blockchain a singular transaction can include different inputs and outputs, wherein any input can be a different type of token. This enables KAPPI to operate as a decentralized exchange platform directly. Zones can also act as fault-tolerant distributed exchanges that is better than other centralized cryptocurrency exchanges in the sense that it is not prone to hacking.

Zones are also able to act as enterprise and government systems that is backed by blockchain where different traditional services can be ran as an ABCI application on any zone. A problem with consensus favouring algorithms like KAPPI can be that any partition in a network that creates a lack of partition within voting power can stop the consensus entirely. The KAPPI architecture addresses this issue through the use of global DWARFs that have regional autonomous zones that each have the zone voting powers distributed across geographic regions. This will allow real features to be added in when designing fault tolerant systems. NameCoin was an early adopter of solving naming resolution problems when adapting to Bitcoin with federated fault tolerant systems.

As it turned out there were issues with this approach, such as not being able to confirm early public keys without downloading every block since the name was last updated. This is since the Merkle model that Bitcoin uses with their UTXO transactions are block hash which allows for existence to be proven but not any updates to names. For this reason, it is not possible to know what the newest value of a name is unless the full node is trusted or unless higher costs are accrued through the requirement of downloading the whole blockchain. If a Merkle-ized search tree were created in NameCoin it would depend on proof of work to finish light client verification. Bandwidth requirements would be large since the light clients would need to download full copies of the headers on all blocks within the blockchain or at least all that were updated to a name. This would create a linear scale in line with the needed time. Name changes on such blockchains also require time, that can run to an hour.

For KAPPI, the only necessary requirement is the latest block hash signed by validator voting power combined with a Merkle proof to the value currently associated with the name. This allows for fast and secure light client verification through name values. In KAPPI this concept is taken further; each zone for name registration can hold another associated domain, such as .org or .com, and each zone can create their own registration and governance rules. Since the KAPPI DWARF is a multiple asset distribution ledger, it has its own token called a KAPPI Token. KAPPI Tokens are licensed to the holder to validate, delegate or vote to other validators, and can be used to pay for any transaction fee. Block transaction fees and KAPPI tokens are rewarded to delegators and validators. Any proportionate amount of KAPPI tokens can be recovered from the reserve pool.

If there is any intentional or unintentional deviation resulting from sanctioned protocol, there will be penalties imposed on the validators. When this happens, a validator will lose its good standing and its proportionate share of tokens at stake will be slashed. Validators are not always available, due to either a disruption or power failure, or other reason. If the validator timeout window gets blocked for any reason, the validator's commit code will not be included in the blockchain more than X amount of times. It will then be inactivated and subsequently lose its stake.

Where there is malicious behaviour that is not immediately recognized by the blockchain the validator will coordinate outside the band to create a timeout on the malicious validators where there is a supermajority consensus. Whenever the KAPPI DWARF halts in respect of a coalition greater than one third voting power, or where voting power censor evidence of malicious behaviour is attempting to enter the blockchain, the DWARF will have to recorder through a hard fork reorganization proposal.

KAPPI DWARF validators can take any type of token, and any combination of token types to be used as fees for processing transactions. Every validator will be able to subjectively set to any exchange and decide which transaction it wants to choose so long as the limit is not exceeded. Collected fees will have taxes deducted and then be redistributed to stakeholders proportionately in relation to their bonded atoms every sixty minutes. For encouraging the early discovery and immediate reporting of any vulnerability found, the KAPPI DWARF asks hackers to publish any successful exploit through a Report Hack transaction that will state the validator got hacked and allow an address to be listed for bounty to be sent.

When such an exploit occurs the delegator and validator will be inactivated and 5% of everyone's tokens will get sent to the hacker's bounty address. The validator will then have to recover any tokens that remain through the use of their backup key. For the prevention of this feature being abused, a portion of unvested vs vested tokens and delegators before and after the hacker report will stay the same and the hacker's bounty will be inclusive of any unvested tokens. Practical Byzantine Fault Tolerance was the original blockchain consensus, but KAPPI consensus is simpler to execute and use. This is because the blocks in KAPPI have to commit sequentially, which supersedes PBFT's view changes. In our blockchain there is no need to make a block commit if the original block has yet to commit. If it transpires the reason why n doesn't commit within the KAPPI zone it will not help to integrate bandwidth sharing votes for N+I block. N+I will not commit if the reason is due to an offline node or a network partition. Block batching transactions facilitate Merkle-hashing of the application state, which works better than PBFT's checkpoint and runs faster transaction commits that are provable inter-blockchain communication.

The KAPPI DWARF also uses several features and optimizations that are higher than what is specified in PBFT. The KAPPI DWARF does not run in any assumption mode relating to connectivity between points and can run on the weakest connection within a P2P network. Proof of stake blockchains were better known and understood after BitShare was created, where stakeholders had to order and commit transaction and were also required to coordinate software updates and any other change of parameter. BitShare improved from the 1.0 with their 2.0 that achieved a higher level of performance within any ideal condition, and every block signed a single signer, leaving a long space of time for final transactions to be created. For KAPPI proof of stake, there is no requirement to have delegators or large witness collateral. KAPPI built on the Ripple approach through refining Federated Byzantine agreement models to create a situation where the process consensus participation doesn't create any fixed or globally known set.

Instead, process nodes create quorum slices that each constitute a trusted process set. A quorum can be defined as a node set that contains at least one quorum slice per node, which facilitates agreement being reached. The KAPPI mechanism security runs by assuming that any quorum intersection would not be empty, and that node availability is contingent on at least one quorum slice to be entirely full of correct nodes. This allows for a trade off that uses either smaller or larger quorum slices that might not be able to balance without the imposition of significant trust assumptions. Nodes will have to choose quorum slices that are adequate for fault tolerance through a hierarchical strategy. This is comparable to the Border Gateway Protocol (BGP) which is used in top level ISPs for establishing global routing tables and is also used when browsers try to manage a TLS certificate. Any criticism of the KAPPI proof of stake system is mitigated here by the token strategy, which sees our new token represent future fee portions and rewards. This is advantageous in the ease it provides whilst maintaining the guaranteed security that is necessary. Bitcoin proposed improvements through their BitcoinNG that creates vertical scalability without costing more, which happens to create larger impact for smaller miners. This is accomplished through separating leader election from the transaction broadcast with leaders being elected through proof of work within a micro block, then allowing the broadcast of transactions until another micro block is found. This method tends to reduce the bandwidth requirements that are needed to compete in the PoW race, which lets smaller miners compete fairly and also allows for regular transactions by the last micro-block miner.

Ethereum proposed a different proof of stake consensus algorithm called Casper, which is ran by a 'consensus by bet' which sees validators bet iteratively on what block they think will be committed based on past commitments and finality can be eventually achieved. The team are actively researching this method and it has created some challenges in the construction of a betting mechanism which can be a provably stable strategy. Casper's main benefit when compared to KAPPI might be in offering a consensus that is focused on availability rather than consistency by default of not requiring quorum voting power. This method will not be the fastest on offer and might be more complex than it needs to be. The inter ledger protocol is also not a solid scalable solution as it gives ad hoc level of interoperation between ledger systems through a bilateral relationship network, which operates like the lightning network. The inter ledger protocol can facilitate a payment but it will focus on payments that are made across different ledger types and will extend the atomic transaction mechanism to be inclusive of both hash locks and notary quorum. This last mechanism operates similar to the KAPPI light client SPV mechanism.

This would include ILP connector notaries not supporting changes by membership and would not facilitate any flexible weighting between notaries. It would however work better on blockchain since it was created for it with differently weighted validators and with the ability to allow for membership changes. ILP payment receivers have to be online, just like with the Lightning network, in order for the sender to receive a confirmation. In an IBC token transfer, the receiver's validator blockchain is responsible for confirmation, not the receiving user. The largest difference would be that the ILP connectors do not keep any authoritative state and are not responsible for the payment information, whereas in KAPPI the DWARF validators act as this is the innovation that allows transfers of tokens between zones to operate securely. ILP payments must be backed through an exchange orderbook since there isn't a symmetrical transfer of coins between ledgers.

Bitcoin proposed sidechains to scale networks that are 'two way pegged' on the Bitcoin blockchain, which is their version of bridging. Sidechains facilitate the effective movement of Bitcoins back and forth to the sidechain and allows for varying sidechain features. Comparable to the KAPPI DWARF, Bitcoin and sidechain work as each other's light clients through the integration of SPV proofs which decide when coins will be transferred back and forth. However, since Bitcoin requires proof of work, this extends through to the sidechain as well, which creates problems. Bitcoin's solution will only work with the Bitcoin token as it does not support other tokens.

Sharding strategies are also being reviewed by Ethereum in an effort to address their own scalability issues. Ethereum is aiming to create a solution that maintains the abstraction layers that the EVM has across all shared space. This makes it clear that KAPPI and Ethereum operate different design goals, which KAPPI being about tokens and not being bound to the EVM. KAPPI will allow the zone creator to determine who is in a position to validate the zone. In KAPPI, anyone will be able to start a new zone, and the DWARF will act to isolate any failures within a zone for the purpose of preserving any token invariants that are preserved The Lightning Network is a token transfer system which has been proposed to operate one layer above the Bitcoin blockchain, or other public blockchain, and which will allow larger order throughput through pushing most transactions out of the consensus ledger into different payment channels through on-chain crypto scripts. This would allow bilateral stateful contracts where sharing digital signatures will act to update the state, and then at close the evidence will be added to the blockchain. The Lightning Network can easily extend across more than one blockchain for the purpose of transferring values in an exchange market, but it can't asymmetrically transfer tokens between blockchains. The KAPPI network does allow such direct token transfers.

A Bitcoin proposal link which aims to raise the per-block transaction throughput is called Segregated Witness, which might increase it by 2 or 3 times while also allowing for faster block syncing between nodes. This solution works within Bitcoin's current setup and would facilitate a soft fork upgrade. KAPPI has no design restrictions and different priorities in scaling. KAPPI will use a BFT round-robin algorithm that is created on the back of cryptographic signatures rather than mining and can commit blocks more frequently for vertical scaling. Well -designed consensus protocols must provide certain guarantees when any tolerance capacity is surpassed, and the consensus fails. This is very much a requirement within an economic system where the Byzantine behaviour can lead to a substantial financial incentive.

The fork accountability guarantee falls in this category, where identifying processes that fail is easy. If the legal system becomes too expensive to invoke then validators will need to make security deposits for continued participation. Those deposits will be able to be revoked if any malicious behaviour is detected. This is different from Bitcoin, where a fork is considered a regular occurrence due to how the probabilistic nature of finding hash collisions and network asynchrony are required. There isn't a reliable source of fork accountability in Bitcoin since it cannot tell the difference between an asynchronous fork and a malicious fork.

In KAPPI, voting stages are called PreVote and PreCommit. A block can be used for nil, or for a specific vote. A Polka represents a collection of PreVotes for a single block. If the PreCommit is 0 in the same round they move on to the next round. Determinism in protocol is strict and has a weak synchrony assumption since any faulty leaders would have to be detected and then skipped. For this reason, validators will wait for some time before they PreVote 0 and the value increases every round. Theoretically, this would require an adversary that was very strong to be able to stop the weak synchrony assumption, that is the ability to cause the consensus to actually fail in the committal of a block. Randomized values on each validator also make this more difficult. Further constraints make sure that only one block will be committed at each height. If a malicious attempt is identified a PreCommit block will have to be justified and if the validator already has a PreCommit block in the first round they will be locked on that block.

Validators will also have to pre vote any block they lock in on. These combined conditions will make sure that there are no PreCommits in validators that do not have justification, and that a validator can only contribute to one PreCommit, stabilising safety and 'liveness' within any consensus algorithm. KAPPI eliminates the requirement to sync all block headers due to the alternative chain of stake being able to be slashed. Slashing does require evidence of a fork to be shared, and light clients are advised to store any block has commits that it witnesses. Light clients will also be able to stay synced through validator set changes for the purpose of avoiding long range attacks.