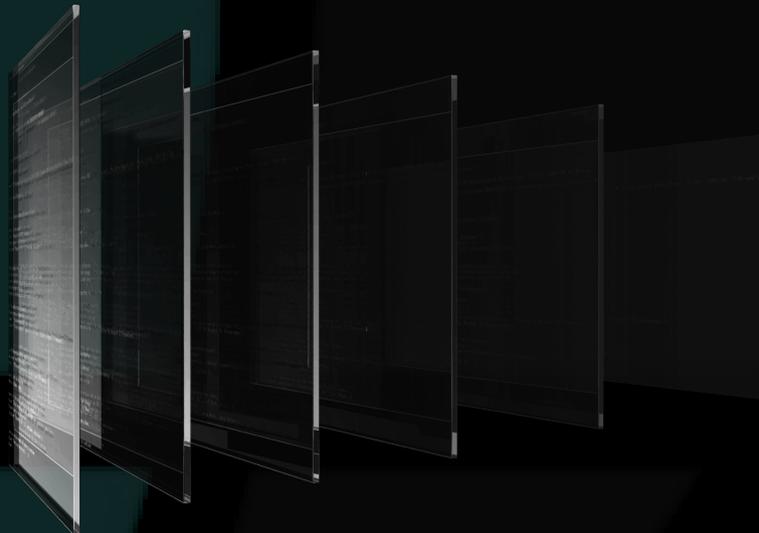




Multifactorial solution for
crypto world

Whitepaper

v1.3 August 05,2018



contact@powerchain.in

Contents

1.0. Why POWERCHAIN?

2.0. DAG Algorithm

2.1. Concepts In The DAG Blockchain

2.1.a. The Double-Spending Issue,
From a Probabilistic Perspective

2.1.b. The Width of the Network

2.1.c. Quick Transactions

2.1.d. No Mining Involved

2.2. DAG Working Method

2.2.a. Topological Sorting

2.2.b. Topological Sorting Demonstration

2.2.c. Critical Path Analysis

3.0. Blockchain vs Directed Acyclic Graph

3.1. Ledger Data Structures

3.1.a. Blockchain

3.1.b. Directed Acyclic Graph

3.2. Consensus

3.3. Confidence of Transaction Confirmation

3.4. Ledger Size

3.5. Scalability

3.6. Conclusion

4.0. POWERCHAIN Coin Technology

4.1. Contents of DAG

4.2. Blockless security

4.3. High Speed Transactions

4.4. Secret Miners

5.0. Token Metrics

5.1. Private sale

5.2. Presale

5.3. Public sale

5.4. Team

5.5. Bounty and Airdrop

5.6. Coinburn Program

6.0. PowerStack Wallet

7.0. PowerPay Credit Card

7.1. Pay With Crypto Instantly Anywhere

7.2. Powerstack-Powerpay Integration

8.0. POWERCHAIN and PowerExchange

8.1. A Decentralized Exchange

8.2. Sharing Exchange

9.0. Project Roadmap

9.5. References

1.0. Why POWERCHAIN?

Many sharing economy companies decide how personal data will be collected and how it is used. Controlling and dealing with personal data is integral to the activities of sharing economy platforms. Users may be required to share a range of information about themselves, including their location, address, job or the services they provide or use – and users are becoming more aware of and concerned about the way that their data is collected, stored and shared. These concerns have been heightened by a number of high-profile data breaches where digital platforms have been subject to malicious attacks resulting in the disclosure of users' personal data. Irrespective of the cause, negative publicity and erosion of users' trust arising from a data breach is highly damaging to the development of the sharing economy.

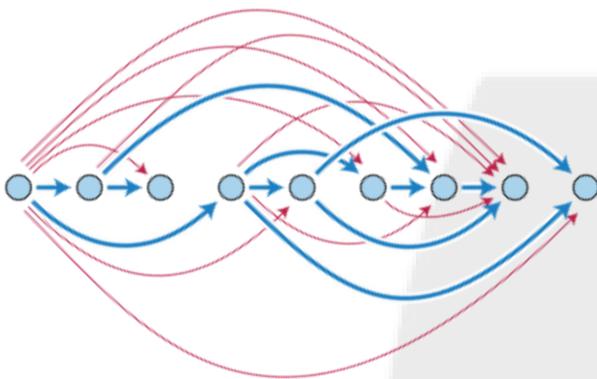
The Blockchain industry is billion dollar business, which is growing steadily every year. Due to technology,Blockchain targets to make daily payment easier. When you pay with Powerpay, you can be certain your payment will be made securely and without high fees. The name Powerpay is derived from the Powerpay credit card that people can use at any ATM in the world, or online business that accepts credit cards.



Why Powerchain Diagram

2.0. DAG Algorithm

DAG is a directed graph data structure that uses a topological ordering. The sequence can only go from earlier to later. DAG is often applied to problems related to data processing, scheduling, finding the best route in navigation, and data compression.



Dag Algorithm Working Method

2.1. Concepts in the DAG Blockchain

2.1.a. The Double-Spending Issue, From a Probabilistic Perspective

The Bitcoin network uses the UTXO (Unspent Transaction Output) model. Users are only allowed to have one transaction placement under their UTXO. There might be more than one miner who solves the hash function at the same time to acquire the right of block validation. This might develop forks temporarily. The validation of a certain transaction is decided by the number of transactions behind it. The rate of transactions coming back into the network is lower with more transactions behind it, which makes the transaction safer.

2.1.b. The Width of the Network

When each transaction is validated, it needs to be linked to an existing and relatively new transaction on the DAG network. If it links to earlier transactions every time, it would make the network too wide to validate the new transactions. Ideally, the DAG network chooses an existing later transaction to link to when a new transaction happens. The goal is to keep the network width within a certain range that can support quick transaction validation. IOTA also proposed its own algorithm controlling the width on the tangle network.

2.1.c Quick Transactions

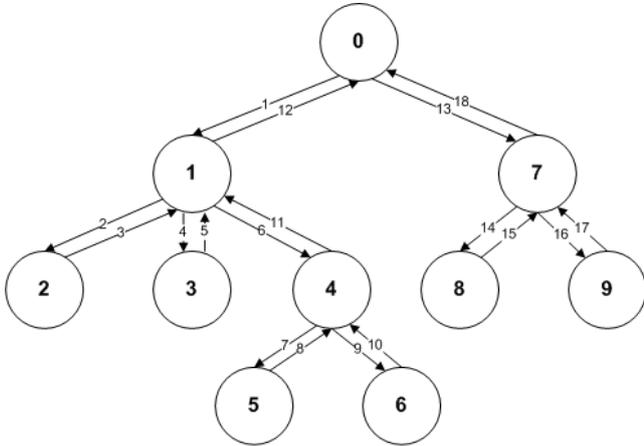
Due to its blockless nature, the transactions run directly into the DAG networks. The whole process is much faster than those of blockchains based on PoW and PoS.

2.1.d No Mining Involved

There are no miners on DAG networks. The validation of transactions goes directly to the transactions themselves. For users, this means transactions go through almost instantly. With the advancement of DAG, we're looking at a future where high functioning and minimum transaction fee chains are possible. That means users can send micro-payments without heavy fees like Bitcoin or Ethereum. DAG will be used for applications that require scalability in the thousands of transactions per second.

2.1. Concepts in the DAG Blockchain

A directed acyclic graph (DAG!) is a directed graph that contains no cycles. A rooted tree is a special kind of DAG and a DAG is a special kind of directed graph. For example, a DAG may be used to represent common subexpressions in an optimising compiler.



0, 1, 2, 1, 3, 1, 4, 5, 4, 6, 4, 1, 0, 7, 8, 7, 9, 7, 0

Example of Common Subexpression.

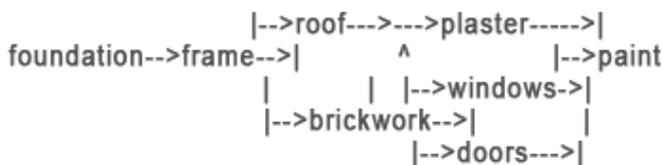
The common subexpression $a*b$ need only be compiled once but its value can be used twice.

A DAG can be used to represent prerequisites in a university course, constraints on operations to be carried out in building construction, in fact an arbitrary partial-order ' $<$ '. An edge is drawn from a to b whenever $a < b$. A partial order ' $<$ ' satisfies:

- (i) transitivity, $a < b$ and $b < c$ implies $a < c$
- (ii) non-reflexive, $\text{not}(a < a)$

These condition prevent cycles because $v_1 < v_2 < \dots < v_n < v_1$ would imply that $v_1 < v_1$. The word 'partial' indicates that not every pair or values are ordered. Examples of partial orders are numerical less-than (also a total order) and 'subset-of'; note that $\{1,2\}$ is a subset of $\{1,2,3\}$ but that $\{1,2\}$ and $\{2,3\}$ are incomparable, i.e. there is no order relationship between them.

Constraints for a small building example are given below.

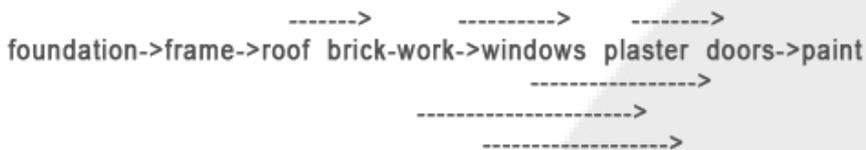


Simplified Construction Constraints.

Note that no order is imposed between 'roof' and 'brick-work', but the plaster cannot be applied until the walls are there for it to stick to and the roof exists to protect it.

2.2.a. Topological Sorting

A topological-sort of a DAG is a (total) linear ordering of the vertices such that v_i appears before v_j whenever there is an edge $\langle v_i, v_j \rangle$ (or whenever $v_i < v_j$).



Example Topological Sort.

Topological sorting can obviously be useful in the management of construction and manufacturing tasks. It gives an allowable (total) order for carrying out the basic operations one at a time. There may be several different topological sorts for a given DAG, but there must be at least one. Note that there may be reasons to prefer one ordering to another and even to do some tasks simultaneously

2.2.b. Topological Sorting Demonstration

Generate a DAG using the HTML FORM below and see the topological sort that results. $|V|$ is the number of vertices in the DAG. The probability, p_r , determines how dense the DAG is, on average:

There are two obvious strategies for topological sorting. One is to find an initial vertex, print it and remove it and repeat for the reduced DAG. The other is to find a final vertex, remove and save it, repeat and finally print the vertices saved in reverse order. These strategies are equivalent as can be seen by reversing every edge and interchanging 'initial' and 'final'. An initial vertex has no edges arriving at it and a final vertex has no edges leaving from it.

A final vertex can be found by following a path from an initial vertex until it is not possible to extend the path. In fact, a final vertex can be found by following a path from any vertex. If the final edge is $\langle x, z \rangle$, z is a final vertex and can be saved. For every other edge $\langle x, y \rangle$, the process must be repeated from all such y . Vertex x then precedes y & z and so on back up to the start vertex. This is a familiar backtracking process effected by a depth-first traversal (see Tree traversal), but here performed on a graph:

```

// visited[] is an array of Boolean

procedure Depth_First(i :Vertex)// Note similarities
  if not visited[i] then // with Tree traversals.
    visited[i] := true;
    for all edge <i,j> // j must follow i in top'-sort
      Depth_First(j)
    end for;
    save(i) // record or process Vertex i
  end if
end Depth_First;

for all i :Vertex // initialise visited[]; been nowhere!
  visited[i] := false
end for;

for all i :Vertex // try all possible starting points
  Depth_First(i)
end for

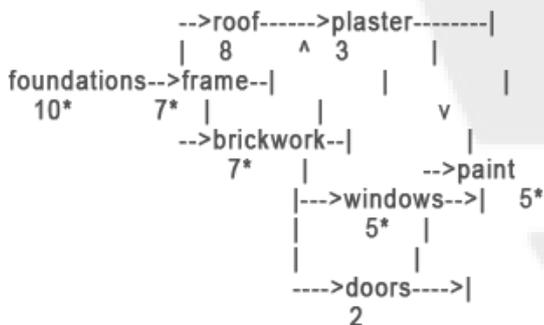
```

Depth-First Traversal of a Graph from a given Vertex.

This algorithm will also traverse an arbitrary graph. It should be compared with the various tree traversal algorithms. The exact coding of the algorithm, in particular the selection of 'each edge', depends on the method of implementing the graph.

2.2.c. Critical Path Analysis

Critical-path analysis is another management problem. The critical-path of a complex task is the most time-consuming sequence of basic operations that must be carried out sequentially even allowing for all possible parallelism. It defines the minimum time that the total task must take even if no expense is spared with the maximum allowed amount of activity going on simultaneously.



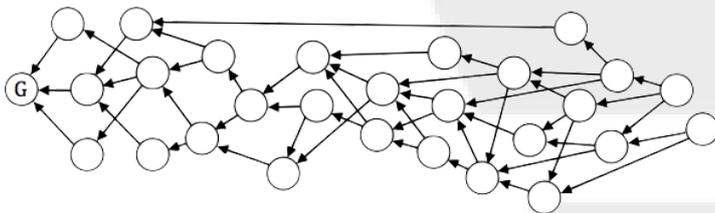
Example of Critical Path Analysis.

The critical path can be found by a modification of the depth-first search.

3.0. Blockchain vs Directed Acrylic Graph

Distributed ledger technology (DLT) enables the maintenance of a global, append only, data structure by a set of mutually untrusted participants in a distributed environment [1]. The most notable features of distributed ledgers are immutability, resistance to censorship, decentralized maintenance, and elimination of the need for a centralized trusted third party. In other words, there is no need for an entity to be in charge of conflict resolution and upkeep of a global truth that is trusted by all stakeholders which do not trust each other. Distributed ledger is suitable for tracking the ownership of digital assets, and hence it's most prominent application is the Bitcoin network [2]. DLT holds promise beyond mere cryptocurrency transfer since an entry in the ledger may be generalized to hold arbitrary data. However, before being applicable on a global scale, DLT needs to solve a number of issues it is currently facing. Blockchains, a specialization of DLTs, are getting a new rival in the field: distributed acyclic graphs (DAG). The most notable difference between the two is that blockchains bundle transactions in cryptographically linked blocks forming a single chain containing the global truth, while DAGs use a graph where a transaction is represented as a node in the graph. This paper compares the two DLT paradigms by focusing on features relevant to their distributed design, and explains how the two tackle some of the known issues distributed ledgers are facing. In particular, we examine the applied data structures for ledger maintenance, consensus mechanisms, transaction confirmation confidence, as well as ledger size and scalability issues. A comparative qualitative analysis is presented using three reference implementations:

Bitcoin [2] and Ethereum [3] serve as reference implementations for blockchain, while Nano (previously known as RaiBlocks) [4] is used to represent DAG. The listed systems are chosen as representative solutions because of a relatively mature implementation with a notable developer community.



Directed Acyclic Graph Working Method

3.1. Ledger Data Structures

This section analyzes data structures that are being used to sustain a distributed ledger. Both DAG and blockchain store transactions in an open ledger. A ledger has its state. Transactions serve as inputs that cause the change to the state, hence DLTs can generally be regarded as transaction-based state machines. However, the two approaches use distinct data structures for maintaining the ledger. While blockchain stores transactions in blocks, DAG stores transactions in nodes. The following subsections explain and compares the two data structures.

3.1.a. Blockchain:

Blockchain consists of ordered units called blocks [7]. Blocks contain headers and transactions, as depicted in Figure 1. Each block header, amongst other metadata, contains a reference to its predecessor in the form of the predecessor's hash. The initial state is hard-coded in the first block called the genesis block. Unlike other blocks, the genesis block has no predecessor. Transactions in Bitcoin and Ethereum are hashed in Merkle Trees [1]. Bitcoin hashes transactions [8] in a single tree, while Ethereum uses three different structures to store transactions, receipts and state [9]. These structures are reviewed further in Section V-A in order to explain how to decrease ledger size.

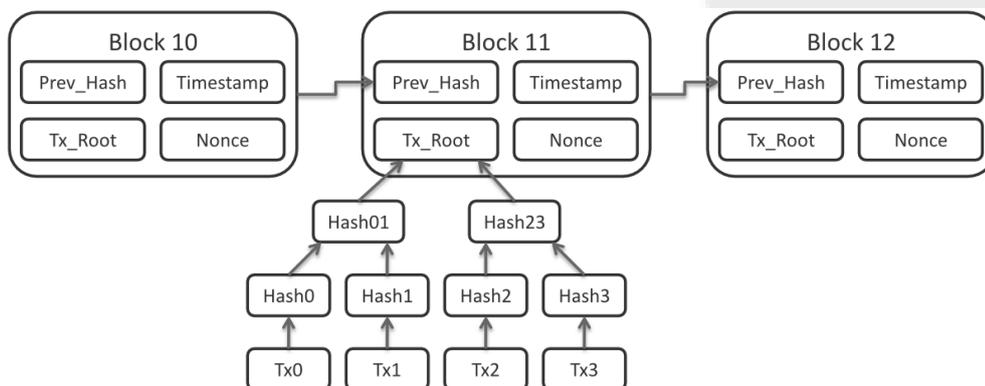


Fig.1. Blockchain as a Data Structure

3.1.b. Directed Acyclic Graph:

In contrast to blocks, a DAG structure stores transactions in nodes, where each node holds a single transaction. In Nano, every account is linked to its own account-chain in a structure called the block-lattice equivalent to the account's transaction/balance history. The structure of the block-lattice is displayed in Figure 2. Each account is granted an account chain. An account chain can be considered as a dedicated blockchain, just for a single account. Nodes are appended to an account-chain, each node representing a single transaction on the account chain. Similar to the genesis block in blockchain, a DAG holds a genesis transaction. The genesis transaction defines the initial state.

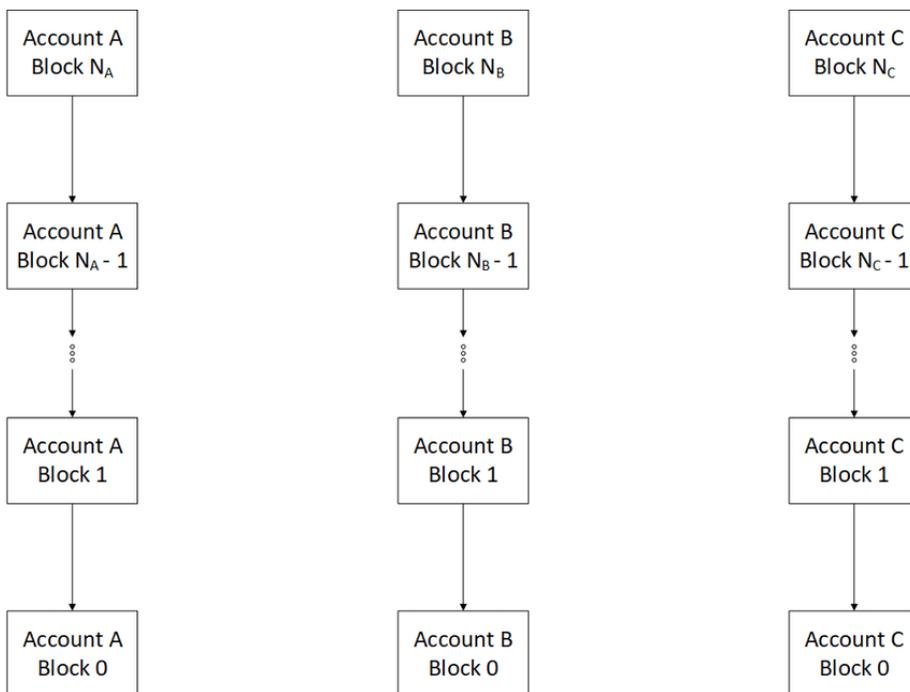


Fig.2. Nano's DAG, the block-lattice.

In Nano, instead of having a single transaction that transfers value, two transactions are needed to fully execute a transfer of value. A sender generates a send transaction, while a receiver generates a matching receive transaction, as depicted in Figure 3. When a send transaction is issued, funds are deducted from the balance of the sender's account, and are pending in the network awaiting for the recipient to generate the corresponding receive transaction. While in this state, transactions are deemed unsettled. When the receive transaction is generated, the transaction is settled. The downside of this approach is that a node has to be online in order to receive a transaction.

3.2. Consensus

In a public and permissionless environment where each node can read from the ledger and append to the ledger, blocks or nodes can be malicious and can not be implicitly.

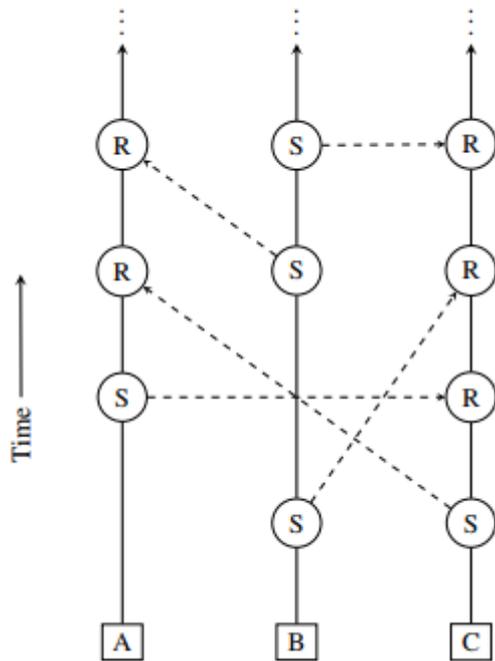


Fig.3. Transaction handling in the block lattice. S represents a send transaction, R represents a receive transactions.

Trusted [7], [10]. Bitcoin, Ethereum and Nano are all public and permissionless solutions, and hereafter we discuss consensus mechanisms for such environments. A. Blockchain For an entry to be appended to the ledger, consensus about the entry needs to be reached in the network, that is, an agreement must be reached regarding the validity of a new entry that is to be appended to the ledger by all nodes. The assumption is that a supermajority of nodes are honest and reliable. Algorithms for achieving consensus with arbitrary faults generally require some form of voting among a known set of participants. One method, often referred to as the Nakamoto consensus, elects a leader by some form of a lottery [11]. The leader then proposes an entry that can be added to the ledger containing a list of previously committed entries. The entries are checked for validity by all other nodes and their consistent ordering is verified.

Both Bitcoin [2] and Ethereum [9] are based on a lottery function called the Proof of Work (PoW) (Ethereum has announced to support Proof of Stake (PoS) in near future [12]). The elected leader broadcasts the new entry to the rest of the participants who implicitly vote to accept the entry by adding it to their local copy of the ledger, and may propose subsequent transaction entries that build on the ledger [11].

1) Proof of Work: In Proof of Work, the first participant to successfully solve a cryptographic puzzle wins the leader election lottery. For example, Bitcoin uses partial hash inversion as the cryptographic puzzle function. Partial hash inversion requires that the hash of a block of transactions together with a nonce (a free variable in the function) matches a certain pattern. The pattern starts with at least a predefined number of 0 bits [2]. The function is difficult to solve intentionally since to manipulate the ledger, an attacker would need to have the supermajority of the computing power in the network, which makes an attack expensive to perform.

Nodes that generate blocks in a Proof of Work driven systems are called miners and the process is called mining. For the use of their resources, miners are granted tokens in the network, as an economic incentive to mine (Ether in Ethereum, Bitcoin in Bitcoin). If there are no miners, no blocks can be mined and there is no transaction throughput.

2) Proof of Stake: While miners in a PoW driven system commit their computational resources to be elected for block generation, in a PoS driven system users stake their tokens to be able to create blocks. In Ethereum, PoS is implemented in the form of a smart contract named Casper [12]. Validators deposit their stake in the smart contract, which in turn picks the validator allowed to create a block. The more tokens a validator stakes, it has a higher chance to create the next block. If an incorrect block is submitted (e.g., it contains double spending transactions), the validator's stake is burned, thus penalizing the validator. PoS has its advantages over PoW. Firstly, it consumes far less electricity than PoW. For example, based on a recent analysis, Bitcoin mining consumes more electricity in a year than a selected set of 159 countries [13]). Secondly, attacks on the network are easily penalized relative to PoW. After an attack on a PoW driven network, the attacker still owns the hardware used for the attack. In PoS, burning stake has the same economic effect as dismantling an attacker's mining equipment.

B. Directed Acyclic Graph In Nano, there is no need for a leader election since users are obligated to order their own transactions. PoW is still being used, however not for the sake of leader election (since there is none). In the context of Nano, PoW is used as a spam protection measure to prevent over-generation of transactions by a malicious user, similar to Hashcash [14]. However, a different method for conflict resolution has been introduced, a system of representatives. When an account is created, it must choose a representative that can be changed over time. Representatives vote in order to resolve conflicts. Their votes are weighted: a representative's weight is calculated as the sum of all balances for accounts that chose this representative. In the case of a conflict, the winning transaction is the one that gained the most votes with regard to the voters weight [4]. For a transaction with no issues, no voting overhead is required.

3.3. Confidence Of Transaction Confirmation

Blockchain As stated in Section 2, PoW uses a stochastic process which makes it impossible to know which node will be elected as a leader. Furthermore, even though an entry has been added to the ledger, there is no guarantee that it will remain a valid entry. This seems to be counter intuitive to the inherent feature of distributed ledger, immutability. However it is indeed expected that a ledger may find itself in a temporary state where there are two different histories stored within the ledger. This phenomena is called a soft fork [15] in blockchain. The issue is eventually resolved by abandoning one version of history over another.

Figure 4 depicts forks in a blockchain. A soft fork can occur when two different blocks are created at roughly at the same time. Due to network delays, some nodes will receive one block over the other, resulting in a state where two blocks claim the same predecessor. For the time being, nodes continue to build the chain on top of their received blocks, effectively creating two chains possibly containing conflicting transactions.

The problem resolves itself when a block is mined that makes one chain longer than the other. The longer chain is adopted, while the shorter one is discarded or orphaned, along with all transactions within it. Orphaned transactions need to be included in a new block. Since a soft fork can occur at any time, if a block has been appended to the chain, there is no guarantee that it will not get orphaned. As the chain increases in length over the referent block, the probability of the block being discarded decreases. Depending on the implementation, there is a suggested number of blocks that need to be appended above the referent one before it is safe to say that it will remain in the chain with great certainty. The number of appended blocks that guarantee block inclusion with high probability are six for Bitcoin [12] and five to eleven for Ethereum [16]. It is worth mentioning that Ethereum is soon to introduce Casper FFG [17], a proof of stake based finality system that is supposed to introduce non-reversible checkpoints, guaranteeing block inclusion.

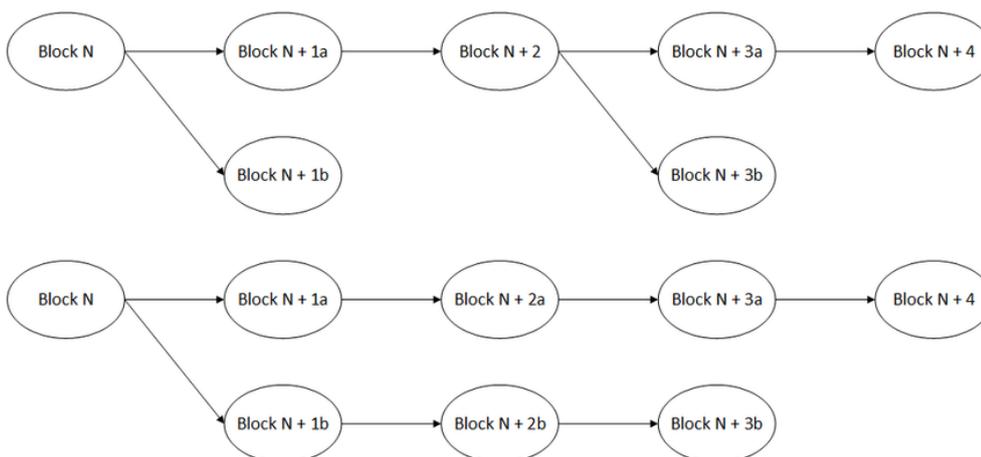


Fig.4. Diagram demonstrating temporary Blockchain forks. The top chain depicts a typical fork, while the bottom chain depicts an atypical fork.

Directed Acyclic Graph In Nano, nodes can create transactions at their own discretion at any point in time. However, inconsistencies similar to those in blockchain are still possible. For example, two transactions may claim the same predecessor causing a fork (forks in Nano are only possible as a result of a malicious attack or bad programming) or a transaction may not have been properly broadcasted, causing the network to ignore all subsequent transactions on top of the missing block. When an inconsistency occurs, representatives are called to vote following the procedures explained in Section III-B. It is important to note that even though a transaction may be deemed settled, it is only confirmed when it receives a majority vote for the send and receive transactions. Beside voting on conflicts, representatives vote automatically on blocks they have not seen before. A representative that sees a new transaction forwards the transaction with its votesignature attached if the transaction is valid.

This means that the network automatically broadcasts consensus information, while the transaction is making its way through the network. A feature that is supposed to be implemented in the future is block-cementing which will prevent transactions from being rolled back after a certain period of time, guaranteeing thus transaction finality [4].

3.4. Ledger Size

As every ledger contains all information since its genesis, its size is constantly increasing. With further penetration of the technology, the size will increase even faster. Bitcoin is estimated to be 145,95 GB in size on 02.01.2018 [18], Ethereum 39.62 GB on 02.01.2018 [19]. Nano's ledger size is 3.42GB with around 6,700,078 blocks on 25.02.2018 [20] [21]. In this section we investigate how reference implementations tackle the issue of increasing ledger size.

Blockchain Bitcoin clients offer a pruning mode, allowing users to delete raw block data after the entire ledger has been downloaded and validated, keeping only a small subset of the data. The data is kept in order to be able to relay recent blocks to peers and handle soft forks. The advantage of the method is that disk space is saved. The downside is that other nodes are no longer able to download the entire history of a pruned node [22]. Similar to Bitcoin's method of ledger pruning, Ethereum offers a pruning mechanism. Ethereum keeps track of the deltas in the global state maintained by a Merkle state tree. A delta in a global state is the difference between two states of the ledger. Changes made to the state are kept in the ledger in the case of a soft fork, when a state needs to be rolled back, and then updated correctly by the miners on the orphaned branch. However, if one is not interested in past states, the deltas can be discarded without harming the chain integrity. A fast sync algorithm has been implemented to tackle this issue. Instead of processing the entire blockchain one link at a time and replaying all transactions that ever happened in history, fast syncing downloads the transaction receipts along the blocks, and pulls an entire recent state [23]. After downloading a state which is recent enough (headofthechain-1024blocks, also called the pivot point), the process is paused for state sync where the Merkle state tree is downloaded from the pivot point. For every account found in the tree, its contract code and internal storage state tree is retrieved. From the pivot point onward, all blocks are downloaded and the node continues its usual operation. The result of the mechanism is a database pruned of the state deltas.

Directed Acyclic Graph Nano distinguishes between three types of nodes: historical which keep record of all transactions, current which keep only the head of account-chains, and light that do not hold any ledger data and only observe or create new transactions (in the current implementation, all nodes are historical nodes). In order to reduce the ledger's size, Nano plans to implement pruning. Since the accounts keep record of account balances instead of unspent transaction inputs, all other historical data can be discarded to decrease ledger size. This feature is yet to be implemented in 2018 [24].

3.5. Scalability

One of the most relevant issues hindering global scale DLT adoption is its scalability. At 05.01.2018. There were around 186,951 pending transactions in the Bitcoin network [18] and around 22,473 pending in the Ethereum network [25]. This section explains how the two technologies handle incoming transactions in terms of scalability. A. Blockchain In order for a transaction to be included in a block (included being different from confirmed, see Section IV-A), a block must be created. A block is created every time when a PoW puzzle is solved, a thus transaction rate is limited by the periodicity at which blocks are created and also by the block size. When increasing the number of nodes in the system, the frequency of block creation does not increase significantly due to the fact that the PoW puzzle difficulty is dynamic so that the block generation time converges to a fixed value. In Bitcoin, a block is mined roughly every 10 minutes with a maximum block size of 1 MB, thereby limiting the Bitcoin transaction rate to between 3 and 7 transactions per second, depending on the size of individual transactions on the blockchain [26] [27].

]. In Ethereum, a block is mined roughly every 15 seconds [28] with a dynamic block size not measured in bytes but rather in gas. Gas is the unit used to measure the fees required for a particular computation [9]. In the context of Ethereum block size, a measure called gas limit defines the maximum amount of gas all transactions in the whole block combined are allowed to consume. In contrast to Bitcoin, this value is dynamic and will adapt to network conditions. This enables Ethereum's transaction rate to be roughly between 7 to 15 transactions per second [29]. The transition to PoS should decrease Ethereum's block generation time to 4 seconds or lower [30]. However, this is still a rather limited block generation rate.

Since Bitcoin and Ethereum are used for payments, it is interesting to compare them with already existing payment solutions, such as Visa which is able to process 56,000 transactions per second [31]. Note also that Ethereum has a significant benefit compared to Bitcoin since it supports smart contracts [9], which expands its potential to become a platform rather than only a cryptocurrency. A potential approach to improve scalability is to increase the block size (be it in megabytes or in gas limit). Increasing the block size also increases the maximum amount of transactions that fit into a block, effectively increasing transaction rate. However, the block size increase would eventually lead to centralization due to the fact that consumer hardware would become unable to process blocks leading to the network relying on supercomputers [29]. One of such efforts is Segwit2x [32] in Bitcoin which, among other features, tries to increase the block size to 2MB. Another approach is to create channels, scaling the transaction capacity. One such implementation is the Raiden Network [33] on top of Ethereum or the Lightning Network [34] on top of Bitcoin.

The solution revolves around creating an off chain channel to which a prepaid amount is locked in for the lifetime of the channel. The involved parties are able to run micro transactions at high volume and speed, avoiding the transaction cap of the network. Any party may choose to leave the channel, after which the final account balances are recorded on chain and the channel is closed. Another attempt to increase scaling in Ethereum is Plasma [35]. The framework creates a nested blockchain structure by the use of smart contracts with a root chain being the Ethereum main chain. Constraints and consensus mechanisms are defined by a smart contract and based on PoS. Only Merkle roots created in the sidechains are periodically broadcasted to the main network during non-faulty states allowing scalable transactions. For faulty states, stakeholders need to display proof of fraud and the Byzantine node gets penalized. An example network being written for the Plasma framework is OmiseGO [36]. A more complex approach to further improve scalability is sharding. Sharding splits the network in K partitions, no longer forcing all nodes in the network to process all incoming transactions. Every shard $k \in K$, in its simplest form, has its own transaction history and the effects of a transition in shard k would effect only the state of k . In a more complex scenario, cross shard communication is available, meaning that for $k, m \in K, k \neq m$ a transaction from k can trigger an event in m [29].

The downside of this approach is that developers would need to be aware that they are programming in a cross shard environment. The Ethereum foundation is attempting to make cross shard communication transparent for developers [29], which will in turn further increase the complexity of the protocol.

B. Directed Acyclic Graph Opposed to blockchain technology where dedicated validators must exist in order to generate and order blocks, a user in Nano must sort his/her own transactions. This approach vastly differs from the way transactions are executed on blockchain systems. Namely, instead of having validating nodes charged with transaction ordering, transaction ordering is done asynchronously by the account owner being in charge of transaction ordering. This approach greatly influences scalability. The consequence of this design decision is that there is no inherent cap in the transaction throughput in the protocol itself. However, peak throughput on a test reached on the main network was 306 Transactions Per Second (TPS) with an average of 105.75 TPS [37]. The limit is currently determined by the quality of consumer grade hardware and network conditions.

3.6. Conclusion

When comparing DAG and blockchain based ledgers, one can conclude that DAG based ledgers store transactions as edges in an directed acyclic graph while blockchains bundle transactions in blocks and append blocks one after another. Blockchain technology determines the global truth by choosing a single branch that holds all the transactions. Global truth and transaction ordering in public and permissionless blockchains is generally done by some sort of leader election, either using PoW or PoS. Leaders are elected stochastically and the global truth is found in the longest chain, while a shorter one is abandoned. Nano's DAG abandons leader election and delegates transaction ordering to users and their representatives to resolve conflicts. Due to the fact that a branch in a blockchain may become orphaned, just the fact that a transaction is included in a block doesn't mean that it will remain in the ledger version containing the global truth.

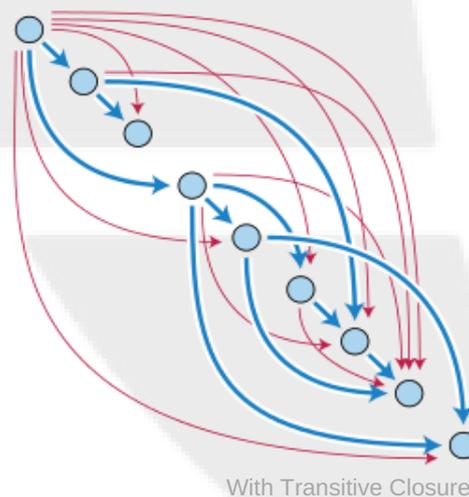
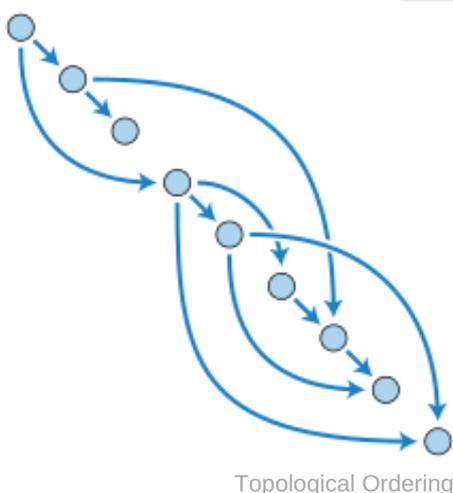
For that reason, it is recommended to wait for some number of blocks to be appended above the referent one before concluding that it is confirmed. In Nano's DAG, a transaction is confirmed when there is a majority of votes cast in favor of a transaction by the representatives. Increased ledger size is a significant problem for all DLTs and this issue is tackled by ledger pruning. The entire history is federated to historical nodes while other nodes only maintain a subset of historical data. Generally, a tradeoff between disk space usage and historical data accessibility is being made. A scalable DLT can be defined as a system where every node does not need to process every transaction, and thus existing DAG or blockchain implementations do not guarantee scalability per se. This paper describes how existing blockchain and DAG implementations try to achieve scalability: Blockchain solutions propose the following approaches: increased block size, support of off-chain channels, hierarchical chains and ultimately sharding. DAGs can improve scalability by coupling network usage and transaction verification, meaning that a user must handle his/hers own transactions in order to use the network. Even though theoretically uncapped protocols for achieving global consensus exist (e.g. Nano's consensus protocol is theoretically uncapped while the Bitcoin network creates a block every 10 minutes), one must take into account real world limitations, e.g., network conditions and processing power.

4.0. POWERCHAIN Coin Technology

4.1. Contents of DAG

DAG will have 4 things:

- 1. Nodes: A place to store the data.
- 2. Directed Edges: Arrows that point in one direction (the thing that makes this data structure different)
- 3. Some great ancestral node with no parents.
- 4. Leaves: Nodes with no children



4.2. Blockless Security

When DAG cryptocurrencies began to develop, most notably DagCoin/Byteball and IOTA. These DAG-based cryptocurrencies broke the blockchain mold, improving system performance and security. Byteball achieves consensus by relying on a “main-chain” comprised of honest, reputable and user-trusted “witnesses”, while IOTA achieves consensus via the cumulative PoW of stacked transactions. Nano achieves consensus via a balance-weighted vote. Nano requires no additional overhead for typical transactions. In the event of conflicting transactions, nodes must vote for the transaction to keep on conflicting transactions. This consensus system provides quicker, more deterministic transactions while still maintaining a strong, decentralized system.

4.3. High Speed Transactions

Powerchain Coin Provides Super fast transactions worldwide and a proper block size with a real coin made on a new DAG based Algorithm which will solve the scalability issues that other coins have.

When the Hash is converted to a DAG based Algorithm 300.000 Tps per second will be reached and the target of team is 1000.000 Tps per second by the end of 2018.

4.4. Secret Miners

Using DAG, Powerchain Network is able to assign the same exact duties to its every member; all the users on the network are both issues and transaction validators at the same time. Powerchain names this issue “Secret mining”

To have a transaction verified by Powerchain, one has to approve two previous transactions (and ensure they’re not conflicting). Also, one needs to attach a tiny amount of proof of work as low difficulty computations are needed to prevent spam on the network.

This removes completely the need to pay fees to miners and thus opens up the possibility to execute microtransactions which could be worth as little as a few cents.

5.0. Token Metrics

A total of 10 billion PCX Coins will ever be created. 10 billion Pcx tokens will be created as Erc20 and then swap to original DAG based Pcx coin after mainnet launch.

Total number of tokens 10,000,000,000 Tokens		Token Symbol PCX	
Total number of tokens for sale 6,500,000,000 Tokens			
Round	Private Sale	Pre-Sale	Public Sale
Bonus	TBA	30%	0%
Lock up	1 Year	No	No
Token Price	1 ETH 100.000 PCX	1 ETH 78.000 PCX	1 ETH 60.000 PCX
Funding(ETH)	10.000	19.200	40.000
Token Distribution	1 Billion PCX	1.5 Billion PCX	4 Billion PCX

PCX Token Metrics

5.1. Private Sale:

10% of PCX Coins (1,000,000,000) will be available for purchase during the private sale for Powerchains Long-term Strategic Business Partners and coin will be locked for 1 year.

5.2. Pre Sale:

15% of PCX Coins (1,500,000,000) will be available for purchase during the presale for all investor interests with %30 bonus.

5.3. Public Sale:

40% of PCX Coins (4,000,000,000) will be available for purchase during the public sale to all investor interests.

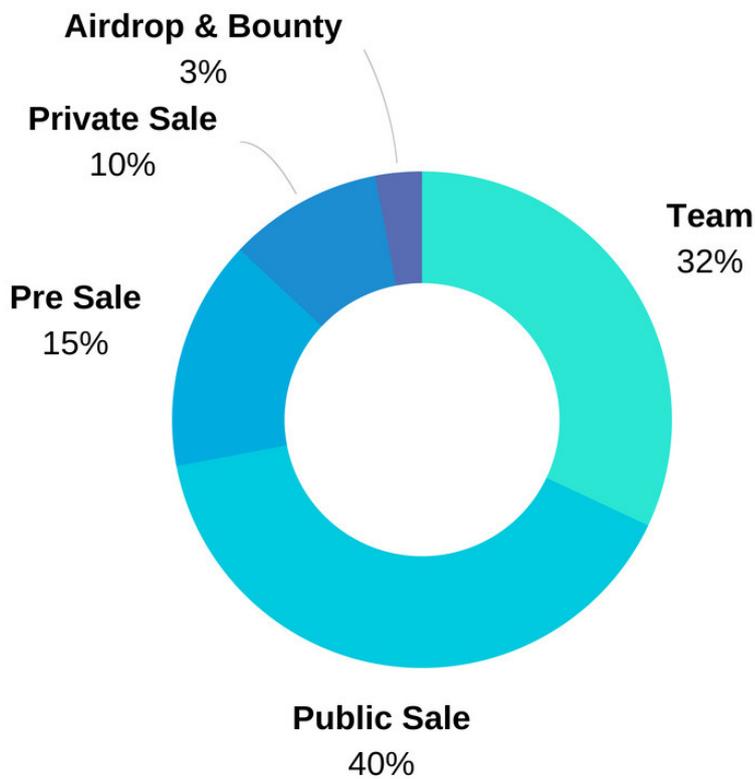
5.4. Team:

32% of PCX Coins (3,200,000,000) will be allocated to the Powerchain Network core team and advisors, to align the team with project delivery.

5.5. Bounty&Airdrop:

3% of PCX Coins (300,000,000) will be distributed for Bounty program and Airdrop participaters.

- %40 of total revenue will be used for marketing issue
- The tokens allocated to the Powerchain Network team and advisors will be locked for two years and unsold tokens will be burned.



PCX Token Distribution

5.6. Coinburn Program:

Powerchain network plans to release a coinburn program by Q2-2019. Team will buy back and burn a calculated amount of Pcx coins until reaching the target %30 of Total supply. The amount will be calculated monthly and coin burn program will go on until total supply decrease to 7 billion.

6.0. PowerStack Wallet

Powerchain intends to develop the Powerstack Wallet, which will handle all Powerchain- related transaction activities such as creating the user's wallet on the Ethereum blockchain (or future Powerchain public chain).

In the Powerstack Wallet API/SDK, all communications are securely encrypted via 256-bit encryption. The Wallet's private key will be only accessible by the wallet owner. Transactions will only be authorised by the Powerchain wallet owner. Powerchain does not keep any personal data within its system.

PowerStack IOS. App.



The complexity of transaction fees, encryption necessitating private and private key management and alphanumeric addresses may create significant barriers to mass adoption. To address these issues, Powerchain will plan to develop simpler authentication methods such as biometric authentication. Powerchain will also utilise different solutions to minimise transaction fees while keeping transactions fully transparent.

Powerstack, the highly anticipated mobile wallet for Powerchain and its digital asset \$PCX. Powerstack is launched on Beta on Android. Powerstack V1.0 release for Android and IOS will be rolled out with additional features added in the near future. Powerstack app is a crypto asset management tool integrating peerto-peer transactions, enabling integration with third-party apps and interfaces through MicroApp, and access to market information. With Powerstack, you can easily create and import digital wallets, make instant face-to-face transactions via QR codes, and track real-time digital market information to acquire market changes timely.

A Unique Social Application:

Our application is a social application. You will be able to add friends, family and others to your own friend list. You can easily send and receive coins from people in your friend list. You will be able to chat with your contacts, through our encrypted chat function. 100% secure wallet to wallet chat function (peer to peer).

PowerPAY IOS. App.



7.0. PowerPay Credit Card

7.1. Pay With Crypto Instantly Anywhere

Problems with debit and credit cards are a part of life. In some countries, the local ATM will not accept your card. There is even the risk that your card will be lost in the ATM machine. Once again, you will pay high fees to use your card abroad. Additionally, when you pay for your accommodation with your credit card, you will have to hand over your card to the person behind the desk. They can easily get all your information and use your card at a later moment for their own personal use or leak your information to third parties.

We turn bright user-centric ideas into reality, bridging decentralized economy and daily life. Imagine if crypto was not just an investment asset, but a real-life spendable currency. Meet PowerPay, the first fast and convenient, ready to use crypto debit card, which can be used for everyday purchases anytime, anywhere. Powerchain will come with its very own physical card that can be used at ATMs. Transactions will only work with your own pincode, so even if you lose your card, no worries, no one else will be able to use your Powerpay card.



PowerCARD Fig.

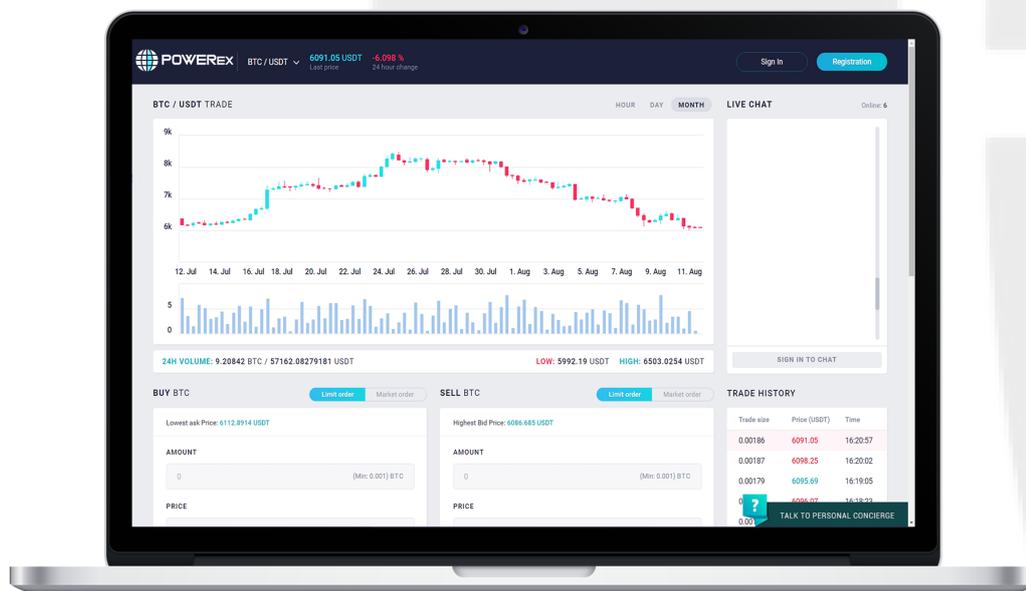
7.2. Powerstack - Powerpay Integration

Powerchain Network is studying on integrating Powerstack wallet to Powerpay that allows owner to send PCX coins from wallet to Powerpay credit card. After this integration Powerpay card will gain double-function issue as valid worldwide Credit and Debit card.

8.0. POWERCHAIN and PowerExchange

8.1. A Decentralized Exchange

The Digital Asset Trading Platforms has been a key component in validating the value of blockchain and its future potential, and as a mean to increase the awareness of blockchain in total. With these trading platforms in it's early adoption, following similar business models and structures found in Stock exchanges, we are now starting to see new trends that are embracing a whole new relationship between blockchain companies, their tokens and the digital asset trading platforms. Although blockchain in its nature represents decentralization, exchanges in other hand have been centralized benefitting a few from token holders' trading. However it is drastically changing. Blockchain and cryptocurrencies have driven the current financial system to an extent where there is no looking back. In this developing era of digital currency, exchanges have become effortless, transparent and decentralized. The decentralization of the exchanges are now moving towards embracing the ideas of sharing economy.



PowerExchange Mac. Fig

If exchanges were centralized in the past, we are moving fast towards business models that embraces decentralization where the users have the opportunity to share the exchanges' revenues and receive additional discounts by participating in the exchange.

8.2. Sharing Exchange

Powerexchange also promotes giving back to the community members by ensuring that 70% of the exchange's revenue goes back to holders of PCX tokens.

Powerchain aims to be on a leading edge with the development of Powerexchange and we look forward to provide more details about Powerexchange the next coming weeks.

9.0. Project Roadmap

2018 Q2:

- Project Startup & Team Formation
- Private sale to strategic long term partners

2018 Q3:

- First round-Presale
- Second round-Crowdsale
- Airdrop and Bounty program
- Public launch on Exchanges
- Start of 4 months 4 Top exchange listing program
- Whitepaper Release

2018 Q4:

- PowerPAY payment solution with mobile payment App-Android beta with V1.0 and IOS App. release
- Powerstack Wallet V1.0 release

2019 Q1:

- Decentralised PowerEXchange public launch
- Powercard V1.0 release
- Powerstack Powerpay integration
- Testnet release & open source coderefining & release of BISC, MVM, PoIE design

2019 Q2:

- Full-function union debugging & updateStress testing and security scanning Mainnet Launch
- Project development

9.5. References

- [1] B. Y. A. Narayanan, J. Clark, and I. F. Y. O. U. Have, "Bitcoin's Academic Pedigree," *Communications of the Acm*, vol. 60, no. 12, pp. 36–45, 2017. [Online]. Available: [https://dl.acm.org/citation.cfm?id=3167461.3132259\(&coll=portal&dl=ACM&CFID=1011196797&CFTOKEN=79862492](https://dl.acm.org/citation.cfm?id=3167461.3132259(&coll=portal&dl=ACM&CFID=1011196797&CFTOKEN=79862492)
- [2] P. Franco, "The Blockchain," *Understanding Bitcoin*, pp. 95–122, 2014. [Online]. Available: <http://dx.doi.org/10.1002/9781119019138.ch7>
- [3] V. Buterin, "A next-generation smart contract and decentralized application platform," *Ethereum*, no. January, pp. 1–36, 2014. [Online]. Available: <http://buyxpr.com/build/pdfs/EthereumWhitePaper.pdf>
- [4] C. Lemahieu, "RaiBlocks : A Feeless Distributed Cryptocurrency Network," pp. 1–8, 2008.
- [5] S. Popov, "The Tangle," 2017. [Online]. Available: <https://iota.org/IOTA{ }Whitepaper.pdf>
- [6] A. Churyumov, "Byteball : A Decentralized System for Storage and Transfer of Value," pp. 1–49. [Online]. Available: <https://byteball.org/Byteball.pdf>
- [7] Daniel Drescher, *Blockchain basics - a non-technical introduction in 25 steps*, 2017. [Online]. Available: <https://console.bluemix.net/docs/services/blockchain/ibmblockchain{ }overview.html>
- [8] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," *Www.Bitcoin.Org*, p. 9, 2008. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>
- [9] G. Wood, "Ethereum: a secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, pp. 1–32, 2014.
- [10] BitFury Group and J. Garzik, "Public versus Private Blockchains. Part 1: Permissionless Blockchains," pp. 1–23, 2015. [Online]. Available: <http://bitfury.com/content/5-white-papers-research/ public-vs-private-pt1-1.pdf>
- [11] "Introduction — Sawtooth v0.8.13 documentation." [Online]. Available: <https://sawtooth.hyperledger.org/docs/core/releases/latest/introduction.html{#}proof-of-elapsed-time-poet>
- [12] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, "Untangling Blockchain: A Data Processing View of Blockchain Systems," pp. 1–20, 2017. [Online]. Available: <http://arxiv.org/abs/1708.05665>
- [13] "Bitcoin mining's electricity bigger than annual usage of 159 countries - Business Insider." [Online]. Available: <http://uk.businessinsider.com/ bitcoin-mining-electricity-usage-2017-11>
- [14] A. Back, "Hashcash - A Denial of Service Counter-Measure," *Http://Www.Hashcash.Org/Papers/Hashcash.Pdf*, no. August, pp. 1–10, 2002.
- [15] J. J. Sikorski, J. Haughton, and M. Kraft, "Blockchain technology in the chemical industry: Machine-to-machine electricity market," *Applied Energy*, vol. 195, pp. 234–246, 2017. [Online]. Available: <http://dx.doi.org/10.1016/j.apenergy.2017.03.039>
- [16] C. Natoli and V. Gramoli, "The Blockchain Anomaly," *Proceedings 2016 IEEE 15th International Symposium on Network Computing and Applications, NCA 2016*, pp. 310–317, 2016.
- [17] V. Buterin and V. Griffith, "Casper the Friendly Finality Gadget," pp. 1–10, 2017. [Online]. Available: <http://arxiv.org/abs/1710.09437>
- [18] "Blockchain Size - Blockchain." [Online]. Available: <https://blockchain.info/charts/>
- [19] "Etherscan ChainData Size Growth - Fast Sync." [Online]. Available: <https://etherscan.io/chart2/chaindatasizefast>
- [20] "RaiBlocks - Do one thing, and do it well - In block-lattice we trust!" [Online]. Available: <https://raiblocks.net/page/summary.php>
- [21] "Nanode • Blocks." [Online]. Available: <https://www.nanode.co/blocks>
- [22] "Prune Release Notes Bitcoin." [Online]. Available: <https://github.com/bitcoin/bitcoin/blob/v0.11.0/doc/release-notes.md{#}block-file-pruning>
- [23] "Geth Sync –fast." [Online]. Available: <https://github.com/ethereum/go-ethereum/pull/1889>
- [24] "RaiBlocks Road Map 2017 V2.0 - General Discussion RaiBlocks Forum." [Online]. Available: <https://forum.raiblocks.net/t/ raiblocks-road-map-2017-v2-0/1392>
- [25] "Ethereum Blockchain Explorer and Search." [Online]. Available: <https://etherscan.io/>
- [26] A. Hari and T. V. Lakshman, "The Internet Blockchain," *Proceedings of the 15th ACM Workshop on Hot Topics in Networks - HotNets '16*, pp. 204–210, 2016. [Online]. Available: <http://dl.acm.org/citation.cfm?doid=3005745.3005771>
- [27] C. et al, "On Scaling Decentralized Blockchains," *International Financial Cryptography Association*, vol. 1, pp. 1–31, 2016. [Online]. Available: <http://www.springerlink.com/index/10.1007/978-3-642-03549-4>
- [28] "How Will Ethereum Scale? - CoinDesk." [Online]. Available: <https://www.coindesk.com/information/will-ethereum-scale/>
- [29] "Sharding FAQ." [Online]. Available: <https://github.com/ethereum/wiki/ wiki/Sharding-FAQ>
- [30] "Can Proof of Stake (PoS) improve the number of Transactions per Second? - Ethereum Stack Exchange." [Online]. Available: <https://ethereum.stackexchange.com/questions/5708/ can-proof-of-stake-pos-improve-the-number-of-transactions-per-second>
- [31] Visa, "Visa Inc. at a Glance," no. August, p. 1, 2015. [Online]. Available: <https://usa.visa.com/dam/VCOM/download/corporate/media/ visa-fact-sheet-Jun2015.pdf>
- [32] "SegWit2x Working Group — segwit2x.github.io." [Online]. Available: <https://segwit2x.github.io/>
- [33] "Raiden Network - Fast, cheap, scalable token transfers for Ethereum." [Online]. Available: <https://raiden.network/101.html>
- [34] S. O.-c. I. Payments, "The Bitcoin Lightning Network:," pp. 1–59, 2016.
- [35] J. Poon and V. Buterin, "Plasma : Scalable Autonomous Smart Contracts Scalable Multi-Party Computation," pp. 1–47, 2017. [Online]. Available: <https://plasma.io/plasma.pdf>
- [36] J. Poon, "OmiseGO Decentralized Exchange and Payments Platform," pp. 1–16, 2017.
- [37] "Stress Testing The RaiBlocks Network: Part II – Brian Pugh – Medium." [Online]. Available: <https://medium.com/@bnp117/ stress-testing-the-raiblocks-network-part-ii-def83653b21f>