## Chapter 2. Basic Concepts

Before we go to the details of Locus Chain's technical aspects, let us introduce some basic concepts and definitions.

### Account

In Locus Chain, an account represents a participating user. A participating user generates a Secret Key (SK) and Public Key (PK) pair for cryptographically sign future information regarding the user. Each account is referenced by its Account Address, which is delivered from the user's Public Key. Both Account Address and Public Key must be publicly disclosed, but Secret Key must be kept secret.

The Account signs his own the transactions with the Secret Key. This sign effectively proves the creator of the transaction because other users or nodes can verify the signature using disclosed Public Key.

An account is initiated by creating its first transaction, TX#0. The first transaction must be a fund-receiving transaction that accepts coins sent by another account. In other words, all accounts (only except for system accounts exist from Locus Chain Genesis) must be delivered from another account.

### Transactions (TXs)

An account may add arbitrary information to Locus Chain's ledger. A transaction is a chunk of information added to the ledger, in a single operation, by an account.

A transaction always contains its creator account's address, and also always signed by the account's Secret Key. No account can create transactions for other accounts.

A transaction may contain hints for other accounts but cannot contain information that modifies other account's status. If an account wants to request some action to other accounts, the account issues a transaction containing the request. After then, the request may be fulfilled when the other account issues an 'accepting' transaction, which refers to the requesting transaction. For example, if account A wants to transfer fund of 10 coins to account B, the A issues transaction like {A.coin=A.coin-10; B.coin=B.coin+10}. In this point, A's coin is reduced, but B's coin is not yet incremented. To increment the B's fund, B itself must issue a transaction like {B.coin=B.coin+10 sent from A}.

### AWTC / Account Chain

Transactions issued by an account have a reference to the last transaction issued just before by the same account. By this, the transactions of an account form an ordered linked list (or a

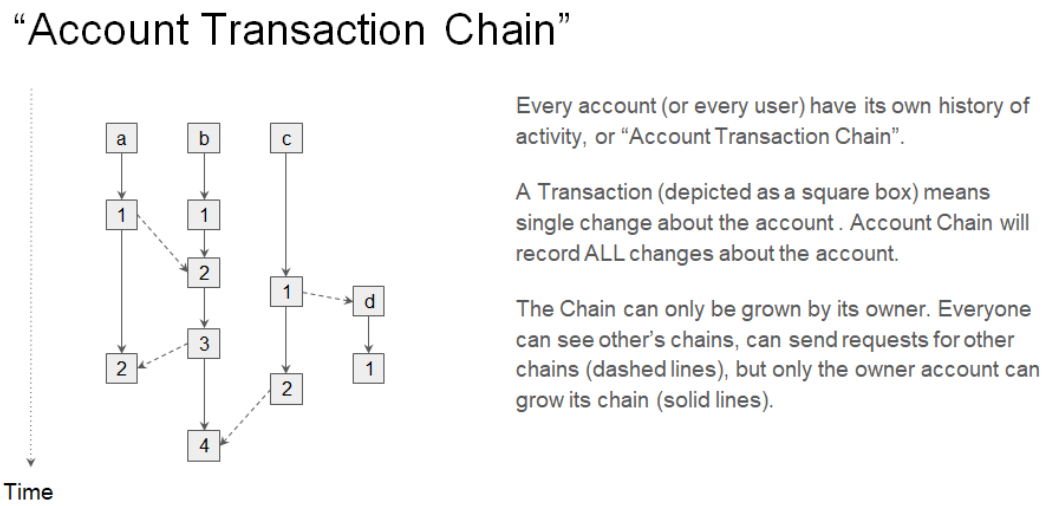
‘chain’). A newly created transaction is always ‘appended’ at the end of the chain. Each account has a chain or its own; therefore Locus Chain’s transactions are ordered in ‘Account-Wise-Transaction-Chain’ structure.

An account assigns each issued transaction with an index, which is sequential numbers increased one by one. The first transaction takes the index number of 0 (TX#0), and the next transactions take number 1 and so on. No transaction of the same account must not have the same number.

Transactions may contain information representing a status change of its owner’s account. The current state of an account is computed by summarizing all transactions from the account's AWTC from index 0 to last. For example, the amounts of current funds the account have can be obtained by adding up all fund changes in all transactions from TX#0 to TX#LAST.

An account may not issue a new transaction that contradicts to its current state. For example, an account should not issue a fund-consuming transaction that consumes more coins than its coin balance.

However, an adversarial account may poison the whole ledger by issuing an invalid transaction. Each node must filter such invalid transactions. This filtering is simple and straightforward in Locus Chain, due to its AWTC structure.




## Ledger

The ledger is a set all meaningful data contained in Locus Chain.

In a narrow sense, the sum of all transactions in Locus Chain composes the ledger.

In a broad sense, the ledger may contain additional data introduced in activities of Locus Chain system, such as a result of Round Consensus.

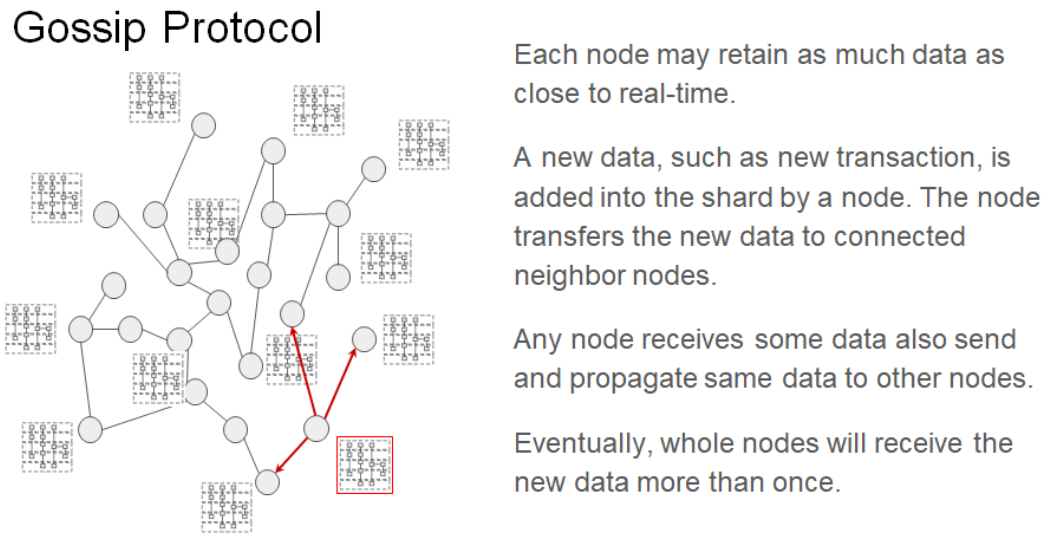
Locus Chain is parallelly managed by multiple shards. Therefore the ledger is updated partially and in parallel by shards.

When a new network node joins Locus Chain network, the first thing the node do is updating ledger data by obtaining the information from the node's shard.

### Network, Nodes, and Shards

Physically, Locus Chain's Network is formed by many computers communicating over the Internet. We call each computers participating the Locus Chain Network as a Locus Chain Node (or just a node). Each node can communicate to any other nodes participating Locus Chain. The communication between nodes is performed in Gossip manner over Peer-to-Peer connected nodes.

If the size of the total network goes too large, the efficiency of communication and computing resources drops. To prevent the problem, Locus Chain splits the network to several 'shards,' which are nodes grouped to a proper size. The ledger is also divided into several groups and assigned to shards. Nodes in shards still could be able to access all ledger data but encouraged to process the shard's local ledger preferentially.




### Round and Committee

When an account issues a transaction, the transaction data is 'flooded' over the nodes in the

same shard, in gossip manner. All nodes in the same chunk receive the transaction when a certain time passes, thanks to the nature of P2P communication over the Internet.

A shard executes some consensus algorithm at a regular time interval to determine and confirm the transactions propagated.

The interval between two consensuses is called a Round.

For each round, a shard elects a certain number of Committee Nodes. The committee nodes, selected in a random and fair fashion, became a Round Committee for the round's consensus. Then the round committee executes an algorithm to agree on the shard's ledger state for the round.

### Coin and Gas

Transactions may contain arbitrary information. However, to express changes of representative numerical value for each account (such as funds,) Locus Chain uses a numerical value named Coin. In essence, Coin in Locus Chain corresponds to some external, real-world value.

Each account is created with some Coins transferred from another account at TX#0. The amounts of Coin changes when the account issues transactions that exchange coins with other accounts.

While Coin resembles some external value, Gas represents a value of Locus Chain's internal activities. For example, issuing a transaction may consume ('burns') the account's Gas. Alternatively, participating in a Round Committee may be rewarded with Gas.

### Proof of Stake of Nodes

Logically, the activity of Locus Chain is established by the activities of all participating accounts. However, in physically, the Locus Chain system is supported by computing and communication power of all participating nodes. A node proves its authority and privileges by amounts of funds(Coins) assigned to the node. An account either can be the primary account of a node, or can assign('delegate') its coins to a node. A node uses the sum of assigned coins ('delegated stakes') as a Proof-of-Stake of the node.

### Mining, Generation, and Epoch

The integrity of a Blockchain is maintained by the honest and fair activity of its participants. Locus Chain encourages honestness and fairness to nodes by fairly rewarding them with incentives including Coin and Gas. Locus Chain includes a popular concept of 'mining' in its design. Currently, Locus Chain compensates the participants at every 24 hours. At the beginning of each 24 hours, (or 'epoch'), the effort of participants is measured and rewarded as Coins and

Gases.

In the next part, we will look into details of Locus Chain’s Consensus, which is the very heart of its design.