

Stellar

Saravanan Vijayakumaran
sarva@ee.iitb.ac.in

Department of Electrical Engineering
Indian Institute of Technology Bombay

September 7, 2018

Stellar

- A blockchain platform for connecting payment systems
- Use cases
 - Cross-border transactions between any pair of fiat currencies
 - Payments to people without a bank account
 - Transfer and trading of custom assets
 - Micropayments
- History
 - Founded by Jed McCaleb and Joyce Kim; launched in July 2014
 - Stellar Development Foundation incorporated as non-profit
 - Stellar Consensus Protocol went live in Nov 2015
- Lumens
 - Native currency of the Stellar protocol; abbreviated as XLM
 - 100 billion lumens created at launch
 - New lumens added to the network at the rate of 1% per year
 - Distribution
 - 5% for SDF to support development
 - 20% for Bitcoin and Ripple holders (19% + 1%)
 - 25% for partnership program
 - 50% for individuals in small amounts (50 – 300 XLM)

Bitcoin vs Stellar

	Bitcoin	Stellar
Specification	Bitcoin Core client	Stellar Core client
Consensus	SHA256 PoW	Stellar Consensus Protocol
Contract Language	Script	NA
Block/ledger interval	10 minutes	approx 5 seconds
Block/ledger size limit	approx 4 MB	100 tx per ledger close
Difficulty adjustment	After 2016 blocks	NA (no mining)
Currency supply	Fixed to 21 million	Variable (104 billion in Sep 2018) ¹
Currency units	1 BTC = 10^8 satoshi	1 XLM = 10^7 stroops ²

¹<https://dashboard.stellar.org/>

²<https://en.wikipedia.org/wiki/Stroopwafel>

Accounts

- Central data structure in Stellar
- Account fields
 - **Account ID** = 256-bit public key for Ed25519 signatures³
 - **Balance** = Number of lumens held by the account
 - **Sequence number** = Current transaction sequence number
 - **Number of subentries** (trustlines, offers, signer, data entries)
 - **Inflation destination** = Account designated to receive inflation lumens
 - **Flags** = Used by issuer of assets
 - **Home domain** = Domain name to find more information about account
 - **Thresholds** = 4-byte field used for specifying access levels
 - **Signers** = A list of upto 20 public keys with weights for multi-sig
 - **Liabilities** = Lumen buying and selling liabilities
 - **Buying liabilities** = Sum of all lumen buy offers by this account
 - **Selling liabilities** = Sum of all lumen sell offers by this account
- Minimum Account Balance = $(\text{NumSubEntries} + 2) \times 0.5 \text{ XLM}$

³<https://en.wikipedia.org/wiki/EdDSA>

Assets

- Lumens, dollars, euros, rupees, bitcoin, stocks, gold, wheat etc
- Any account can issue an asset
 - Assets are linked their issuer
 - Lumens are the only asset that does not require an issuer
- Issuers of assets are called **anchors**
 - Anchors can be banks, individuals, non-profits, local communities
 - Each anchor has an issuing account
- Holding assets = Holding credit from particular issuer
- One must trust issuer to trade your asset balance for actual asset
- Example
 - I pay Rs 100 to ICICI Bank to get 100 ICICI Bank Rupees credited to my Stellar account
 - Later, I ask ICICI Bank to return my Rs 100 after deduction from my Stellar account
- Asset fields
 - Asset code = 4 or 12 alphanumeric code
 - Account ID = Account of issuer

Trustlines

- Entries in the Stellar ledger which identify and quantify trust
- Trustline fields
 - **accountID** = Account the trustline belongs to
 - **asset**
 - Asset code
 - Account ID of issuer
 - **balance** = Amount of asset owned by account
 - **limit** = Maximum balance the account can hold
 - **flags** = Trustline specific settings
 - **liabilities** = Asset buying and selling liabilities
- Example
 - Account ID = My account ID
 - Asset code = Rupees
 - Account ID of issuer = ICICI Bank
 - Balance = 5,200
 - Limit = 10,000
- Assets can be sent to another account with a trustlines with same issuer

Payment Operation

- A Stellar transaction is made up of operations
- Payment operation inputs
 - **destination** = Account ID of payment recipient
 - **asset** = Asset being sent
 - **amount** = Amount of asset being sent
- Possible errors
 - PAYMENT_UNDERFUNDED = Not enough funds to send
 - PAYMENT_SRC_NO_TRUST = Source account has no trustline with asset issuer
 - PAYMENT_NO_DESTINATION = Destination account does not exist
 - PAYMENT_NO_TRUST = Destination account has no trustline with asset issuer
 - PAYMENT_LINE_FULL = Destination account does not have sufficient limit to receive payment

Full error list: <https://www.stellar.org/developers/guides/concepts/list-of-operations.html>

- Cross-asset payments need an exchange

Decentralized Exchange

- Stellar protocol has built-in support for trading assets
- Accounts can make **offers** to trade one asset for another
 - Account must hold asset it is selling
 - Account must have trustline with issuer of asset it is buying
- Offer fields
 - **sellerID** = Account ID of offer creator
 - **offerID** = 64-bit integer
 - **selling** = Asset the offer wants to sell
 - **buying** = Asset the offer wants to buy
 - **amount** = Amount of asset being offered for sale
 - **price** = 32-bit numerator, 32-bit denominator
 - **flags** = Used to indicate passive offers
- Offers are checked against existing **orderbook** and filled if possible
- Unfilled offers are added to orderbook

Path Payment Operation

- Suppose we want to send Rupees and let the receiver get Nigerian Naira
- Path payment operation inputs
 - **sendAsset** = Asset being sent
 - **sendMax** = Maximum amount of asset to send
 - **destination** = Account ID of payment recipient
 - **destAsset** = Asset recipient should get
 - **destAmount** = Amount of receiving asset recipient should get
 - **path** = Array of upto 5 assets defining hops to take
 - For USD to EUR through XLM and BTC, path = [XLM, BTC]
- Possible errors
 - PATH_PAYMENT_NO_ISSUER = Issuer of one the assets is missing
 - PATH_PAYMENT_TOO_FEW_OFFERS = No path connecting sendAsset and destAsset
 - PATH_PAYMENT_OVER_SENDFMAX = Paths that could send destAmount would exceed sendMax

References

- **Stellar Wikipedia page**
[https://en.wikipedia.org/wiki/Stellar_\(payment_network\)](https://en.wikipedia.org/wiki/Stellar_(payment_network))
- **Stellar network status** <https://dashboard.stellar.org/>
- **Assets** <https://www.stellar.org/developers/guides/concepts/assets.html>
- **List of operations** <https://www.stellar.org/developers/guides/concepts/list-of-operations.html>
- **XDR definitions of Stellar data structures**
<https://github.com/stellar/stellar-core/tree/master/src/xdr>
- **External Data Representation (XDR)**
<http://tools.ietf.org/html/rfc4506.html>