CISC 332/326 - Assignment 1 - Report

Bitcoin Core - Conceptual Architecture Analysis

Sunday, February 19th, 2023

Deluca, Seb – 20sad4@queensu.ca

Huang, Robbie – 20rh1@queensu.ca

Kim, Ethan – 20eik@queensu.ca

Leyne, Aidan – 19afl@queensu.ca

Liang, Sean – sean.liang@queensu.ca

Lomonaco, Artie – 19al63@queensu.ca

Abstract:

This project aims to understand how the components found within a conceptual architecture integrate, and function together to complete a desired process. The technology covered within this report is the open-source client-side interface for the Bitcoin protocol: Bitcoin Core – "is an open source Bitcoin fork, styled around the principles of cheap transactions, on-chain scalability, and reliable direct payment networks".

Bitcoin Core facilitates peer-to-peer transactions, staking, and wallets; all of which will be discussed within this report. This report details the analysis and overview of the high-level architecture, as well as use cases and subsystem components. Processes and architectures will be visualized to best understand the flow of the system. Use cases will outline the function of peer-to-peer transactions and inter-wallet features. Lastly, the effects of software concurrency found in high-level architecture will be analyzed thoroughly. Fundamentally this report describes the inner workings of Bitcoin Core, its interactions with users, and the associated processes it follows.

Introduction and overview:

In 2017, Chris Matthieu co-founded Bitcoin Core, an open-source fork of the original Bitcoin protocol, intending to make it easier to build open-source blockchain technologies. Bitcoin Core was designed to be more optimized, more secure, and more flexible than the original Bitcoin interface. Bitcoin Core established itself through its hybrid work-based and stake-based consensus mechanism, which combines the low-energy use of proof-of-stake and the security of proof-of-work.

The use cases cover the implementation and application of Bitcoin Core in a functional setting, including the wallet, sending/receiving transactions, and mining. Wallets can store or spend the respective currency found within, and Bitcoin Core provides a friendly GUI for checking the balance of a virtual wallet. Transactions are made through the blockchain, and Bitcoin Core facilitates these transactions in the most open-source manner compared to other Bitcoin transaction systems. Mining is done through a hybrid model that incorporates proof-of-work and proof-of-stake to securely mint new coins onto the blockchain while reducing energy consumption.

In this report, we will delve into all the key areas critical to the success of Bitcoin Core, detailing subjects and providing a thorough explanation of them. We will also provide diagrams to illustrate the flow of components and a dictionary of terms used throughout the report. By the end of this report, you will have a comprehensive understanding of the key components and design considerations essential to the success of Bitcoin Core.

High-Level Architecture Descriptions

Blockchain

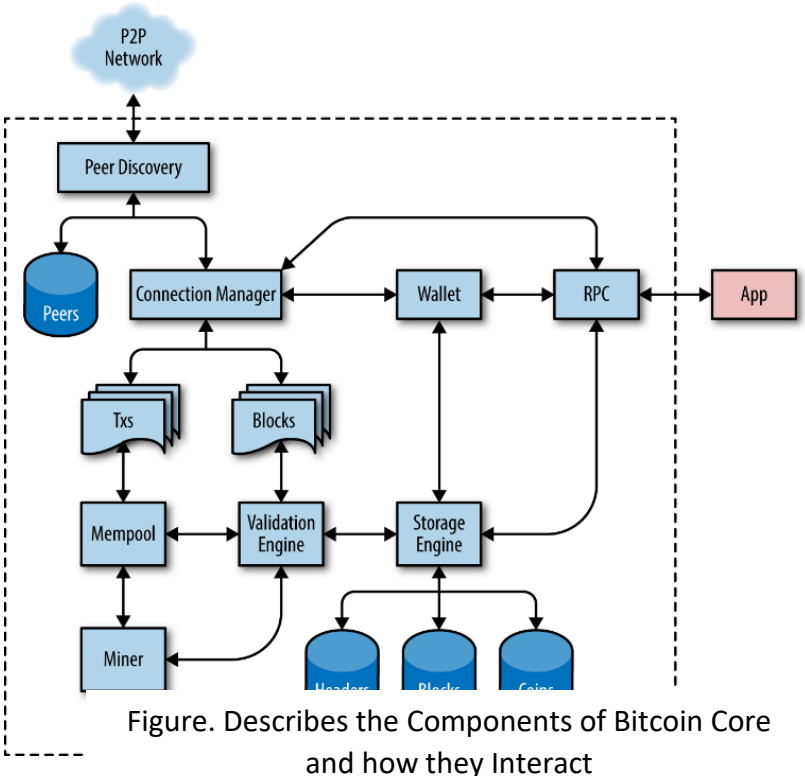


Figure. Describes the Components of Bitcoin Core and how they Interact

A distributed ledger, recording transactions across a network of peers. Transactions are posted publicly and must be verified consensually by other nodes on the network. Blockchain technology is the foundation for cryptocurrencies, as it performs the necessary checks for validity and integrity before adding the transaction to a block and cementing itself in the larger network. This is the record of the individual ownership of all "Bitcoin (BTC)" - the digital currency of the Bitcoin Network. n. Each block contains a set of information - or in this case, transactions - and are dependent on its predecessor. New blocks are therefore appended sequentially to the network's existing chain of blocks - forming the blockchain. Looking to the beginning of the blockchain is the earliest block ever mined, while the newest block is found at the end. Each block has a unique cryptographic hash, and once a block is added to the chain, its information can be considered permanent and unalterable. It is through the peers participating in the network that the chain can be maintained, either by storing the complete chain or simply the most recent block

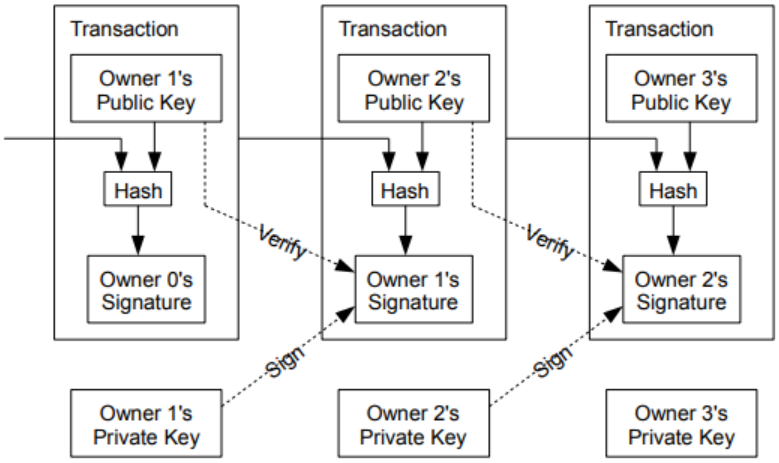
Transactions

When transacting on the network, peers make use of the tokens minted and distributed by the network. Using a combination of their public key and a hash of the previous transaction, coins are sent from one owner to another. When the transaction is received, the new owner will make use of their private key to sign and authorize the contract.

It should be noted that although some linking is inevitable, the use of public keys and changing key-pairs are used to attempt to preserve anonymity within the network. Further security measures are also in place to prevent double spending on the network. Timestamping transactions and keys prevent attackers from double-spending or recalling transactions. Further, the authentication needed within the network helps ensure that an attacker is unable to pre-derive a chain before a transaction is completed.

Contracts

Contracts are used to enforce conditions on the use of funds and financial agreements within the Bitcoin system. In a Bitcoin transaction, funds are locked to a specific public key. The owner of the private key associated with that public key can spend the funds at any time by creating a new transaction that includes a digital signature created with the private key. A contract with a time-lock specification pertaining to fund availability is a first example of a common contract type. Another example of a common type of Bitcoin contract is a multi-signature address. In these contracts, multiple public keys are required to spend the funds; instead of just one. For example, a 2-of-3 multi-signature address would require the digital signatures of any two out of three specified public keys to spend the funds. The example above illustrates the implementation of complex conditions involving multiple parties; a feature unique to smart-contracts.



Wallets

The Wallet is the Bitcoin Core component and subsystem controlling access to a user's tokens, managing key addresses, tracking their balance, as well as creating and signing transactions. Mainly, the wallet refers to the data structure that stores and manages a user's private and public keys digitally in a file, while also maintaining other information related to transactions for the Wallet Program. Wallet Programs create public keys to receive satoshis, the smallest denomination of a Bitcoin and equivalent to one 100 millionths of a Bitcoin. Wallet programs interact with each other and work together to accomplish all needed tasks on the network. For example, one Wallet distributes the public keys to receive satoshis, while another signs transactions and spends them. Overall, the three primary elements of the Wallet are a public key distribution program, a signing program, and a networked program.

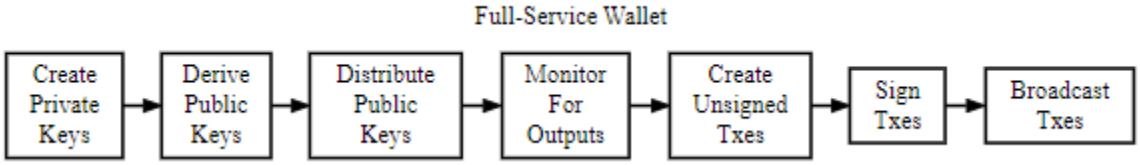


Figure. Example of a Full-Service Wallet

The Bitcoin Wallet probes the peer-to-peer network information from the blockchain and enables the broadcasting of new transactions. Once transactions are signed, it is the responsibility of the Wallet that they are broadcasted to the network. In the above figure, the Broadcast TXs box connects to the connection manager – the component responsible for broadcasting the transactions to other peers in the network.

The Bitcoin Wallet also interacts with RPCs, another main function of the application. The RPC interface allows other programs to control Bitcoin Core. For example, the Wallet allows spending funds, affecting consensus and verification, reading private data, and otherwise performing any operations that can cause loss of money, data, or privacy. The Wallet state is callable using RPC, remaining consistent with itself and with the chain state at the time of the call.

Payment Processing

Payment processing represents the larger service allowing users to send and receive payments in exchange for products or services. In conventional banking systems, the payment processing is outsourced, and connected to third-party APIs. Amongst other factors, exchange rates between various currencies must be considered when linking two parties. It is the responsibility of this application to collect and process the required data to provide fair value. If the values differ substantially, the system enters into a "safe mode" so a human can evaluate the situation.

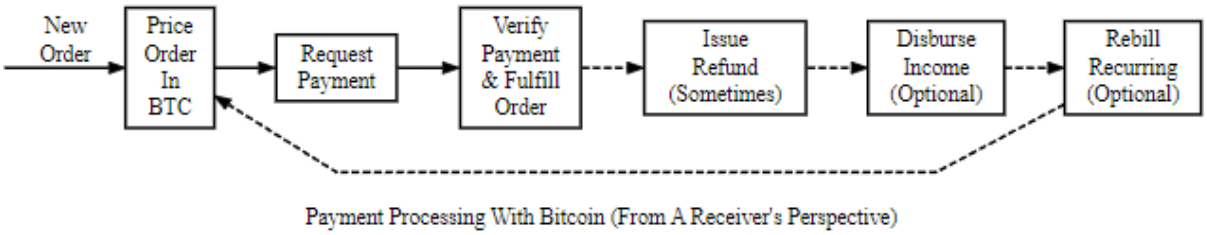


Figure. Illustrates payment processing through receivers perspective.

Operating modes

There are two primary configurations (modes) for a client to validate the blockchain:

Full Nodes - the default and most secure mode of operation for Bitcoin Core. Systems running this configuration download the entire blockchain and validate all blocks from the genesis block) up-to to the most recent. It also broadcasts new transactions and blocks to the network.

SPV Clients - a more lightweight client only downloading and validating the block headers rather than the entire blockchain. They are faster and require less storage space than full nodes but are also less secure.

Peer-to-Peer Networks

The peer-to-peer network subsystem of Bitcoin Core is a critical component of the Bitcoin network. It enables the communication between nodes, facilitating the sharing of information about transactions, blocks, and the state of the blockchain. The network is composed of nodes - that is, every computer running the Bitcoin software. The peer-to-peer architecture allows for the removal of central servers for a protocol facilitating communication directly to other peers. The network maintains the security and integrity of the blockchain through a consensus of the nodes, agreeing if new blocks are valid before appending them to the blockchain. In turn, enabling the Bitcoin Network to operate as a decentralized system, without any singular entity possessing control over the blockchain, making the network more resilient to attacks and providing better privacy for users.

Mining

The mining subsystem of Bitcoin Core is responsible for managing the process of hashing and minting. Within Bitcoin and other cryptocurrencies, minting enables the creation of new Bitcoin tokens by adding blocks to the blockchain. Mining, through the process of hashing, involves solving complex mathematical problems using specialized software and powerful hardware. The first miner to solve the hashing problem and add a new block to the blockchain is rewarded with a certain amount of Bitcoins, known as the block reward. However, mining is not always a one-peer operation; pooled-mining also exists. In the case of single-peer-mining, also known as solo mining, they are seeking to generate new blocks entirely on their own. While boasting the highest reward, this is both time and computationally expensive for the peer. In contrast, pooled mining allows multiple miners to share, or "pool", their resources together to generate blocks. Understandably, this method is often quicker; however, the rewards are distributed between the contributors based on the amount of work contributed by each miner.

Use Cases

Checking Wallet Balance

The user's balance is calculated by scanning the Bitcoin network for all UTXOs and finding the sum to calculate the balance that belongs to the user. UTXO is an arbitrary value for multiple Satoshi. Similar to a physical coin, once a UTXO is created, it is not divisible. During a transaction, if the UTXO is larger than the desired value of a transaction, it must be consumed entirely and will give two outputs: the first one being the amount paid to the other user, and the second being the 'change' from the transaction or remaining amount after a transaction. This 'change' value is returned to the sender's wallet.

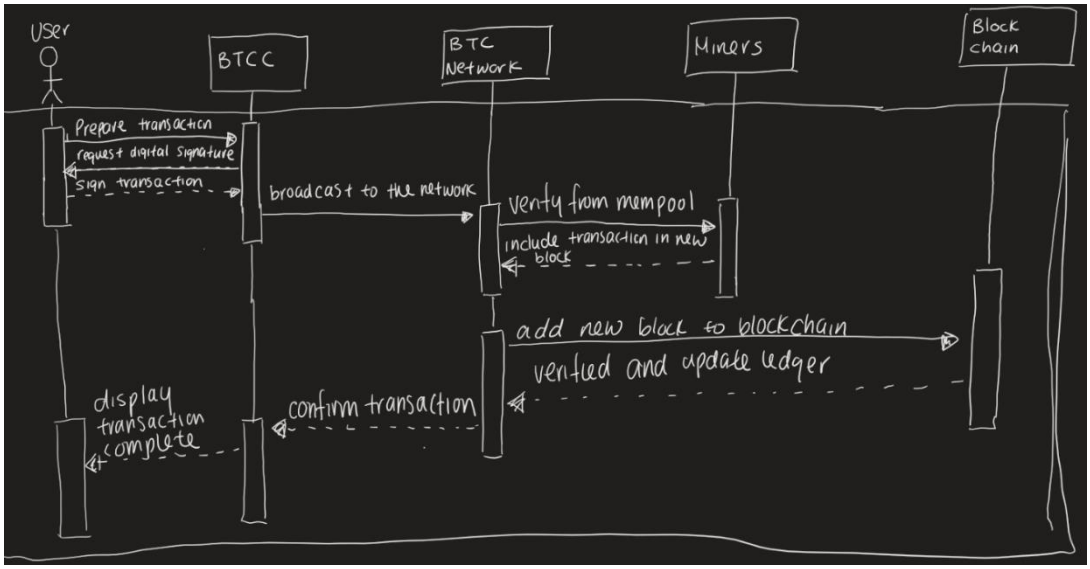
Sending Transactions

Making transactions on the Bitcoin network is an essential use case. The transactions are the core building block of the Bitcoin system, as they are the base for all operations supporting the network.

To send a transaction, the user will first create a new transaction within the Bitcoin Core software GUI. The user must specify the recipient address, the amount of bitcoin sent, and transaction fees available to the miners, rewarding their resources to process the transaction. Bitcoin Core will then create a digital signature for the user with their private key, verifying the user, and allowing the transaction to take place.

Once the transaction has been signed, Bitcoin Core broadcasts the transaction onto the Bitcoin network. The transaction is then moved into the mempool, where the miners will then process the transaction by adding it to a new block and have it be validated through mining.

Once validated, the block is added to the blockchain, allowing the funds to move to the recipient's wallet. The transaction is then confirmed, and the ledger is updated to reflect the new transaction. The user can monitor the status of their transaction using the Bitcoin Core interface.



As we can see, similarly to traditional financial institutions, the sending and receiving of transactions is paramount to the success and continuation of the Bitcoin network. Throughout the process, Bitcoin Core along with other Bitcoin protocols, maintains the integrity of the blockchain, allowing for a secure and safe transaction.

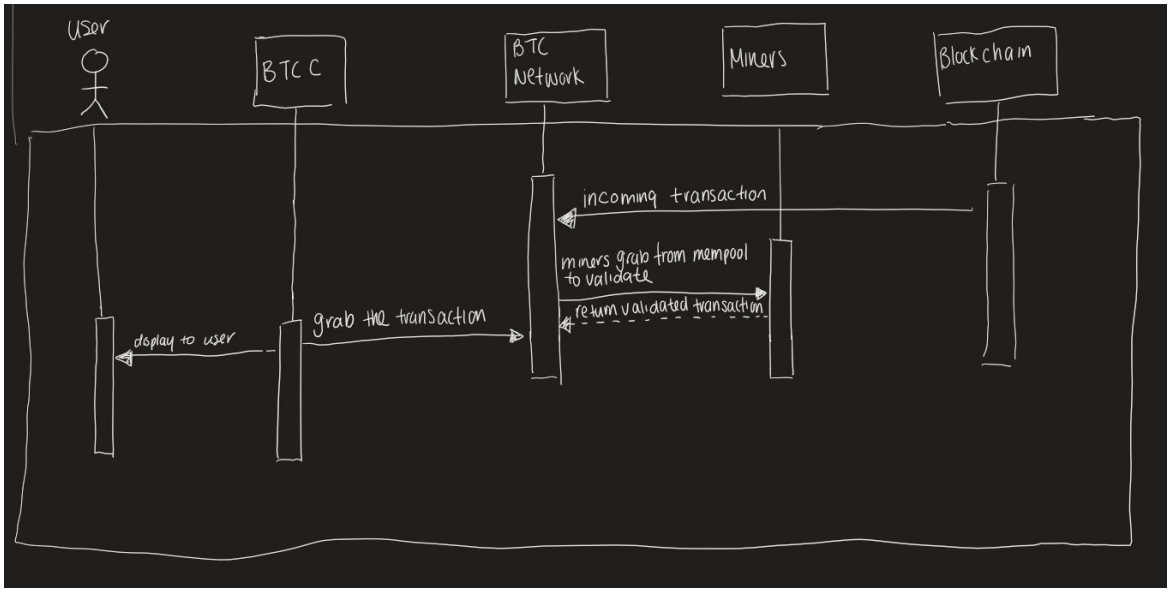
Receiving Transactions

The complementary portion of the transaction process consists of the user receiving the funds from the sender through the network. This process deviates slightly from what was presented above. The user sending the Bitcoin transaction will ask for the wallet address. Once acquired, the user specifies the amount of Bitcoin sent, as well as pay the transaction fees required to the miners. The Bitcoin is then be broadcasted on the bitcoin network where the miners will grab it from the mempool and have it validated through mining. The receiver's wallet will then detect the transaction that is being sent to their wallet; it will grab it and add the transaction to the wallet. Afterwards the transaction is confirmed and updated on the ledger.

Mining

The mining process within the Bitcoin network is essential to its operation. Bitcoin Core facilitates this process in several ways.

For example, Bitcoin Core operates through consensus rules – these rules that dictate how blocks are created and validated. Rules are enforced by validating each mined block, ensuring that it follows the rules outlined in the Bitcoin protocol, ensuring that the network remains secure and trustworthy. Miners are not exempt from this; they must also follow the consensus rules as they are interacting with the blocks within the Bitcoin network and creating new ones.

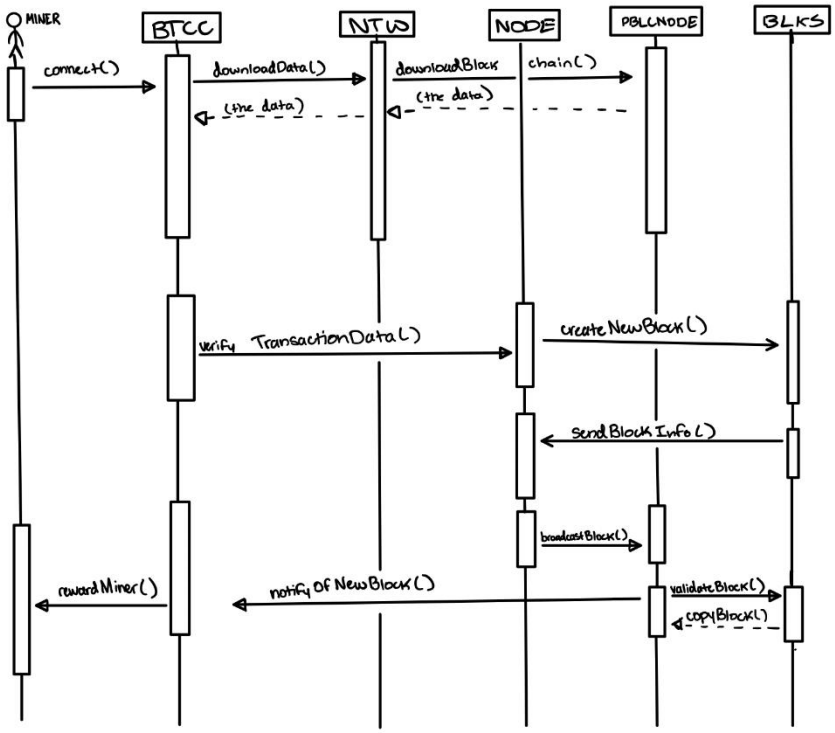


Interaction with other nodes maintains a large role in the mining component of Bitcoin Core. Through the peer-to-peer network, miners collaborate to pool resources and share rewards. Miners also share information with other miners entering the networks as to the current block, new transactions, and the state of the blockchain. It is through the miners that new blocks are effectively distributed to other nodes on the network.

All transactions on the network must be verified; it is through mining that this computationally expensive task is achieved. Ensuring the sender of a transaction has enough funds to complete a transaction, that transaction adheres to the consensus rules, and that the transaction is properly posted are all core elements of the network's functionality accomplished through mining. Finally, Bitcoin Core allows miners to interact with the blocks through the c Creation of new blocks, as well as posting transactions to the blocks, adding a nonce value, and calculating a block header. The block header is then sent to the Bitcoin Core network to be validated and then added to the blockchain.

Overall, the mining operations outlined above, and with all credit to the mining nodes, enabled through the Bitcoin Core application, form the backbone and support of the Bitcoin network.

Sequence Diagram



Process of Sequence Diagram

- The miner launches Bitcoin Core (BTTC) and initiates the mining process using the function connect().
- Bitcoin Core connects to the Bitcoin network (NTW) and downloads the latest block and transaction data using the function downloadData()
- The Bitcoin Network retrieves the Blockchain from the Public Nodes (PBLNODE) and returns it back to Bitcoin Core through the Network.
- Bitcoin Core verifies the transaction information received and sends it to the miner's node (NODE) using the function verifyTransactionData().
- The miner's node receives new transactions from Bitcoin Core, and then creates a new Block (BLKS) with the function createNewBlock().
- The miner's node broadcasts the new block to the public nodes using the function broadcastBlock().
- Other nodes on the network validate the new block, using the function validateBlock(), and add it to their local copy of the blockchain using the function copyBlock().
- A block is validated if it adheres to the consensus rules, as mentioned above.
- Bitcoin Core is notified of the new Block's successful validation using the function notifyOfNewBlock().
- The miner receives a reward in Bitcoin for successfully mining the block from Bitcoin Core using the function rewardMiner().

Limitations and Lessons Learned

Some limitations found when exploring the Bitcoin Core user experience are:

- The Bitcoin Core platform lacks support for commercial wallets. This inherently reduces the functionality of the platform and the network as businesses are not able to participate in the transactions using the token
- The user experience has a steep learning curve making it difficult for first-time users and other new adopters to understand and grasp the platform leading to possible stagflation in the number of adopters
- Scalability - the current design of the Bitcoin Core protocol limits the maximum number of transactions that can be processed in a given frame. This has resulted in long wait times and high fees during periods of high network activity, creating a significant barrier to adoption and use.
- Usability – Bitcoin Core can be difficult for non-technical users to understand and begin to use. Once aging, hindering widespread adoption and use.

Throughout this project, multiple lessons and learning points have also transpired:

- The interaction between the number of components of the Bitcoin Core software and as they pertain to their use in the larger Bitcoin network and protocol.
- Accessibility of information relating to the subsystem interactions. Only a small number of diagrams and documentation was present, while other information had to be obtain from other sources and related-back in order to obtain a larger understanding.
- Documentation provided mainly focused on the interaction between the client and the Bitcoin protocol instead of interactions within the client. In hindsight, this is something that should have been considered, as most users are not looking for documentation on the software itself.
- Sequence diagrams were difficult to generate, requiring multiple revisions and a constant flow of verification from whatever sources possible.

Conclusion

The architecture of Bitcoin Core is a fine mesh of components that deliver a streamlined Bitcoin interface. The open-source and innovative hybrid PoS/PoW consensus mechanism provides a quality unparalleled by competing technologies. In summary, this report has delivered a detailed look into the high-level architecture, components, and functionality that comprise the foundation of Bitcoin Core. Chris Matthieu and the founding developers of Bitcoin Core have laid a path for a more equitable, open-source, block-chain future.

Glossary

Abbreviations

API - Application Programming Interface

BTCC - Bitcoin Core

NTW - Bitcoin Network

P2P - peer-to-peer

PoS - Proof-of-Stake

PoW - Proof-of-Work

RPC - Remote Procedure Call Node

SPV - Simplified Payment Verification

TXs - Transactions

UTXO - Unused Transaction Output

Diagram Abbreviations

NODE - the Miner's personal node on the Blockchain

PBLNODE - nodes on the Blockchain that do not belong to the Miner

BLKS - Blocks on the Blockchain

Other Terms

Bitcoin - the pioneering cryptocurrency, introduced by Satoshi Nakomoto and establishing the idea and proof of a decentralized banking system

Bitcoin Core - the open-source client software developed by the Bitcoin Organization, enabling users to interact with the Bitcoin blockchain.

Blocks - a unit of the larger block chain, stemming from the prior, comprising a number of public verified transactions, officially posted and locked cryptographically.

Blockchain - a distributed ledger, recording transactions of blocks across a network, most commonly of peers.

Contract - a programmable way of binding two or more parties to an agreed transaction

Genesis Block - the first block posted to the blockchain

Hashing - the process of

Mining - the action of validating transactions posted to a block by other users of the blockchain, resulting in the minting of a reward for the user, or group of users, for their resources.

Minting - the issuing of a new coin, or portion of a coin, by the network usually to reward work performed.

Network - the collection of multiple nodes opting in and out, using the same protocol, and working together, forming the backbone of the blockchain.

Node - the term used to describe a member of a network. A node is a computer as a single user can have multiple computers. Nodes are able to create transactions when using the platform but are also responsible for verifying other transactions.

Peer-to-peer - the interaction style where two nodes communicate between each other without the need to route through a central service. Such connections are between 2 or more nodes at once.

Private Key - unique string of alphanumeric characters, generated and given to a node/user when it enters the network, used for multiple purposes and to be kept secret from others

Public Key - unique string of alphanumeric characters, hashed from a private key, allowing a user/node to interact with other users/nodes on the network

Proof-of-Stake -

Proof-of-Work - the consensus system used in the Bitcoin blockchain where a node, or multiple nodes, perform hashing computations in order to verify transactions on the blockchain.

Timestamp - the process of attaching the time occurrence of an event to a transaction or other object; basis of the Bitcoin protocol transaction and security verification.

Transaction - the exchange of an amount of tokens, or partial tokens, between parties (peers) on the network and posted publicly for verification

Simplified Payment Verification - an evolved process of payment verification, differentiating itself by querying other nodes for recent transactions instead of requiring a node to maintain the full blockchain

Wallet - virtual location where a node or user's tokens are stored; the virtual or crypto-wallet maintains the idea of a bank account that can be found in the traditional banking system.

References

Bitcoin Core integration/staging tree. (2023, February 20). GitHub. https://github.com/bitcoin/bitcoin/blob/master/doc/JSON-RPC-interface.md

Developer Guides — Bitcoin. (n.d.). Developer.bitcoin.org. https://developer.bitcoin.org/devguide/index.html

Transactions - Mastering Bitcoin [Book]. (n.d.). www.oreilly.com. Retrieved February 20, 2023, from https://www.oreilly.com/library/view/mastering-bitcoin/9781491902639/ch05.html#:~:text=The%20wallet%20calculates%20the%20user

A Blockchain Glossary for Beginners. (n.d.). ConsenSys. https://consensys.net/knowledge-base/a-blockchain-glossary-for-beginners/