Bitcoin and Bitpay API

Universal Network Coin (UNC) Team
What we hope you learn

- The fundamental features of the underlying structure of Bitcoin
- How you can take advantage of systems such as Bitcoin to support services such as Bitpay, which can process payments
  - Use a nice interface to Bitcoin instead of creating your own closed-off currency!
- Offers some peace of mind in using these APIs if you know how the underlying service works
- How digital currencies can deviate from Bitcoin (and still be successful implementations)
  - Litecoin
  - Ripple
What is Bitcoin? (High-level)

Bitcoin is a P2P digital currency system

Key features (general):
- Anonymity
- Decentralized
- Secure
- Supports buying and selling
- Block chain (record of transactions)
- Mining/Currency creation
First, a quick [e-]handout

- Explanation of terms you will need (Bitcoin vocabulary)
  - We will go over them in gory detail
- Assumption of basic hash function properties
  - Namely, easy detection of modification
Fundamental pieces

Will serve as our “roadmap” through Bitcoin

- The P2P structure that Bitcoin relies on, to include:
  - Protocol
  - Network structure
- Users
  - Miners and traders
- Transactions
- Transaction blocks and the verification process
- The block chain
The P2P network

- Implements a gossip protocol for communication between peers on the network
- This protocol relies on an overlay network design
Gossip Protocol Example

- Think of a 3rd grade classroom
  - There is a break for all students every 30 minutes, during which students pair off and chat
  - At the beginning of the day, Alice tells Bob that Cathy wore 2 different colored socks
  - At each 30 minute interval (assuming random pairing), a new individual will receive the news
    - There is the case that an individual has already heard the news, in which case there is no additional effect
      - This can be beneficial, though. Take for instance a situation in which Joe did not hear the rumor well - it will be reinforced at his next break
Application of the example

- We can therefore think of nodes in Bitcoin’s P2P network as 3rd graders spreading “rumors” of transactions.
- In this way, information propagates very quickly.
- Another (darker) way to view this is as the spread of a disease:
Overlay network

- A network where constituent components are built on top of one another
- Follows closely with the nature of the gossip protocol, taking advantage of its hierarchical nature
- Nodes are connected by “links”, which can be thought of as paths in a tree graph
- Underlying layers of the network facilitate these connections or links
- A good way to think of this is telephone networks operating as a layer built over the internet -> VOIP
Depiction-graphs on graphs
Users

- Consist of miners and traders
  - can really be thought of as one user who can do 2 functions
- Traders make *transactions*
- Miners both verify individual transactions as well as entire *blocks*
  - New blocks are added to the *block chain*
- All users are nodes on the aforementioned P2P network

*These terms will become clear as we progress*
The fundamental component that everything else builds on
Where we start our Bitcoin journey
Basically synonymous with “trade” or “purchase”
Illustrate by example:
○ User A wishes to trade 5 BTC to User B
○ User A “initiates” this trade by broadcasting the intent to his/her peers
○ User A also specifies a transaction fee (mining reward), which is awarded to the miner who verifies the trade
What does this information look like/how is it presented or stored?
How is this transaction ultimately “confirmed”?
For each transaction, a transaction block is made containing:
- The public keys of user A and B
- The amount of coins to trade (5 BTC in our running example)
- Previous transaction record(s) >= amount of BTC specified by user A
  - “I’m user A and I just got 10 BTC from user D, so I have enough currency for this new transaction. Here is evidence...”
  - In this case, user A must specify the change resulting from the transaction
  - “I’ve just shown that I have more than enough, so I will also mention that there is some change here (10-5=5 left over)”
- In actuality, all 10 will be “spent”, but a virtual user will give 5 back to user A. In this way, there is no extra xact. for change
- A transaction fee (reward for miner, as previously mentioned)
The transaction block is broadcasted to all miners through the P2P network. Multiple miners accumulate and start to work on verifying transaction(s) with the following goals:

- verify that the transaction is signed by A
- verify that A has enough BTC for this transaction
- basically, make sure the info in the transaction block “adds up”

To achieve these goals, miners

- use A’s public key to check the signature
- check if the associated transaction record(s) (10 BTC from D to A in our example) is valid
  - Achieved by checking the entire block chain
  - hash functions aid in detecting (nefarious) modifications
Once a miner provides a proof-of-work for their efforts in verifying all the transactions in their given block,
- (Useful to think of each miner’s block as a ledger in this case)
- Ledger is filled as incoming broadcasts are received
- Once an entire ledger has been confirmed, it is broadcasted to the network and nodes begin to work off of a block chain that ends with that particular confirmation
  - (The longest ledger is the one that is accepted)
  - Once nodes start using a block chain that incorporates that specific block, all transactions therein have become “real” (they are now concrete and in effect)
The block chain

- The block chain contains all blocks of transactions
  - Literally ALL blocks
    - starting with the “genesis block” - the very first block of the block chain (from the dawn of [Bitcoin] time)
- Each miner has a copy of the block chain
- Note that copies of the block chain could be different at some point for some miners
  - This is forking, and is perfectly acceptable
  - The network will simply accept the longest block chain in circulation
Block chain Diagram

Simplified Bitcoin Block Chain
Summary

- Bitcoin is a competitive system for obtaining currency
- It is a “distributed consensus system” implemented by a gossip protocol which operates on an overlay network
- Let’s go over all of the pieces now that we have seen the entire picture
  - Users trade on the network, creating pending transactions
    - miners verify and accumulate these transactions in blocks
  - Blocks are verified by the network and added to the global block chain
  - In this way, everyone agrees on the same state of the system
    - reliance is heavily based on hash functions ensuring integrity
Block chain
  ○ A “shared public ledger”
  ○ Each user has their own copy
    ■ Very large! (currently ~22gb); mail order disc copy
    ■ [https://blockchain.info/charts/blocks-size](https://blockchain.info/charts/blocks-size)
  ○ The entire Bitcoin network is reliant on this component
    ■ Contains all constituent blocks since the inception of Bitcoin
  ○ Enforces integrity
    ■ No duplicate/falsified transactions - a global reference to check transactions against
Take advantage of Bitcoin

Now that we have a basic understanding of Bitcoin,

- How can one make use of Bitcoin and the Bitcoin network in a given project?
- We will focus on a fairly popular Bitcoin integration service: Bitpay
What is Bitpay?

Bitpay provides Bitcoin payment service
Main Features of Bitpay

- Accept payments from a mobile wallet (smartphone app)
- Web API allows for
  - Email billing
  - Shopping carts
    - Bitpay will “host” your checkout (a magic box to throw all your checkout details into)
    - Manage your shopping cart - payment is taken care of by Bitpay
  - Embedded invoice
  - JSON payment gateway API
- Libraries available in
  - Python, Java, PHP, Nodejs, Perl, Ruby…
Apart from these general applications/features, there are 3 main features of Bitpay that are well-suited for developers:

- Bitcore
- Insight
- Copay
Insight

- Open-source block chain explorer
  - Accessed through RESTful interface
  - [https://insight.bitpay.com/](https://insight.bitpay.com/)
  - Similar to Blockchain: [https://blockchain.info/](https://blockchain.info/)
Bitcore

- Open-source Javascript library
- Allows interfacing with Bitcoin network in almost any imaginable way
- Demo: creating a wallet
  - [http://bitcore.io/#started](http://bitcore.io/#started)
Copay

- Modeled after co-signing a check in real life
- Multisignature provides extra security
- Decentralization eliminates third party risk
- Deterministic wallets ease back up needs
- Open-source ensures transparency
  - [https://github.com/bitpay/copay/](https://github.com/bitpay/copay/)
Limitations of Bitcoin

- Degree of acceptance
- Volatility
- Mining difficulty
- Space requirement to store the block chain
- Waiting time before a transaction is verified?
Leave it to the audience

- Case studies:
  - https://ripple.com/
  - www.litecoin.com