Querying bitcoin blockchain using SQL

Thejeswar Seggam
19111062
6th semester
Biomedical Engineering

1 abstract

Bitcoin is the first major decentralized cryptocurrency with wide acceptance. A core technological innovation of Bitcoin is blockchain, a secure and pseudonymous general ledger that stores every Bitcoin transaction. Blockchain has received enormous attention from both the commercial and academic worlds, and it is generally recognized as the enabling technology of the Internet of Value (IoV), in which securely stored valuable entities are intended to be transferred as easily as information. Current blockchains are designed as special kinds of Online Transaction Processing (OLTP) systems. To incorporate the increasingly important blockchain technology into Information Systems curriculum, one approach is to store blockchain data in a SQL database, thus allowing fast data access and a simpler understanding of the underlying concepts.

2 Introduction

2.1 blockchain

Blockchain is a secure platform, ledger, or database where buyers and sellers could store and exchange value without the need for traditional intermediaries." The results can be drastically reduced transaction cost and friction that disrupts the usual ways of conducting businesses in a wide spectrum of areas. Using higher education as an example, blockchain allows a Web of decentralized transactions, possibly enabling huge changes in keeping student records, optimizing student loan management, improving pedagogy, incubating meta-universities, and ultimately creating a global network of learning institutes.

2.2 Bitcoin

Bitcoin is the first major decentralized cryptocurrency with wide acceptance. It solves the double spending problem, in which a digital currency may be spent two or more times, by storing a publicly accessible general ledger of all Bitcoin transactions in a blockchain. Unlike bank transactions, Bitcoin transactions

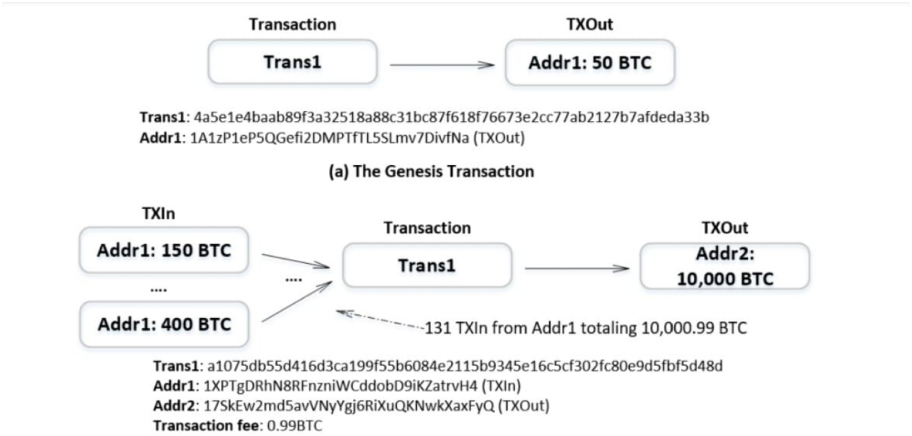
are digitally signed and irreversible, and are stored in a peer-to-peer network of nodes (running Bitcoin Core) using the Bitcoin protocol. Bitcoin Core (Bitcoin.org, 2018) is open sourced and contains code storing and maintaining a copy of the Bitcoin blockchain in a node, together with a reference Bitcoin's client to interact with the blockchain

The current size of the Bitcoin blockchain is more than 200GB and it stores all of the more than 373 million of Bitcoin transactions.

3 bitcoin blockchain and transactions

Bitcoin blockchain stores the entire history of Bitcoin transactions. A transaction stores the transfers of Bitcoins (in the unit of Satoshi, with 1 Bitcoin (BTC) = 100,000,000 Satoshi) from input accounts to output accounts, plus authorization and other information. Bitcoin account addresses are public key hash values that can be authenticated by the corresponding private keys. Users can use a Bitcoin wallet to manage their Bitcoin accounts (public key hashes) and interact with the Bitcoin blockchain. Unlike a bank transaction transferring money from one account to another account, Bitcoin transactions allow multiple inputs and multiple outputs.

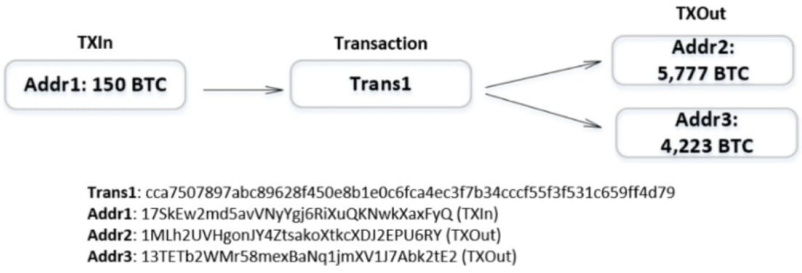
Unlike a bank transaction transferring money from one account to another account, Bitcoin transactions allow multiple inputs and multiple outputs. Figure 1 shows four historically interesting Bitcoin transactions. Figure 1a is the very first Bitcoin transaction as 50 BTC went to the Bitcoin address '1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa', which is assumed to be controlled by Satoshi Nakamoto, the mysterious Bitcoin's inventor(s).



Bitcoin blockchain is known to be pseudonymous as all transactions are publicly accessible but the ownerships of accounts are anonymous within the blockchain. Many Websites provide Web pages and APIs to access Bitcoin's data in various formats such as HTML, JSON or XML. For example, one can copy and paste Bitcoin addresses, transaction hash addresses, or block addresses from this paper into the popular site, Blockchain.info.

The first Bitcoin transaction, called the genesis transaction here, is included in the first block (known as the Genesis Block or Block 0) as shown in Figure 1a. It is known as a Coinbase transaction to reward 50 BTC to the Bitcoin miner who had successfully created the block. Since the reward is created out of nowhere by Bitcoin, there is no input in a Coinbase transaction. Bitcoin mining involves finding a small enough block hash of the 80 Bytes header of the new block. The required smallness, or difficulty level, of the block hash is adjusted every 2,016 blocks to ensure that every block is mined in about 10 minutes.The 80 Byte block header contains the hash of the Merkle tree which is constructed from the hashes (addresses) of all transactions, ensuring that transactions cannot be changed. The block header also contains the previous block hash and thus the block is chained together. Changing a block will change its block hash, and any subsequent block hashes will needed to be recomputed. This ensures that the blockchain is nearly impossible to tamper with.

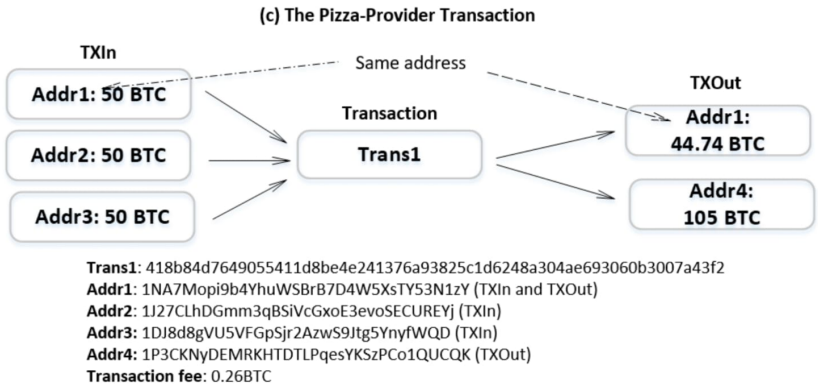
Unlike a bank that keeps the balance of every account, Bitcoin blockchain keeps track of every transaction, including those transaction outputs (TXOut) that have not yet been spent, which are known as unspent transaction outputs (UTXO). UTXO can be used for future transaction inputs (TXIn).



above figure shows another famous transaction, a1075db55d416d3ca199f55b6084e2115b9345e 16c5cf302fc80e9d5fbf5d48d, the first documented purchase of a good with Bitcoin in which 10,000 BTC was used to buy two Domino's pizzas on May 17, 2010. This pizza transaction has one TXOut (presumably going to an account owned by the pizza provider). Note that the buyer gathered together 131 UTXO from previous transactions as TXIn to pool together 10,000.99 BTC. This paid the 10,000 BTC to the TXOut, and the transaction fee of 0.99BTC, which was

collected by the block miner together with the 50 BTC mining reward. After the transaction was confirmed, these 131 UTXO were recorded as spent and can no longer be used as inputs to other transactions, thus solving the double spending problem. The current approximate 60,947,620 UTXO indicated in Table 1 is how the current Bitcoins are 'stored.'

Figure below shows how the 10,000 BTC were used by the 'pizza person' to provide for two TXOut in a transaction called the pizza-provider transaction here. Again, after this transaction, the previously unspent TXO to the pizza person with 10,000 BTC was recorded as spent.



In general, transactions can have multiple inputs and multiple outputs. Figure 1d shows the oldest Bitcoin transaction with three TXIn and two TXOut such that one TXOut address is also a TXIn address in the transaction (418b84d7649055411d8be4e241376a93825c1d 6248a304ae693060b3007a43f2). The sender gathered three UTXO in his accounts, each with 50 BTC. One TXOut received 105 BTC, and the change of 44.74 BTC, after 0.26 BTC transaction fee, was sent back to the address 1NA7Mopi9b4YhuWSBrB7D4W5XsTY53N1zY, which is one of the input addresses owned by the sender. We refer to this transaction as the 3i2ochange transaction

4 Accessing bitcoin blockchain data with SQL

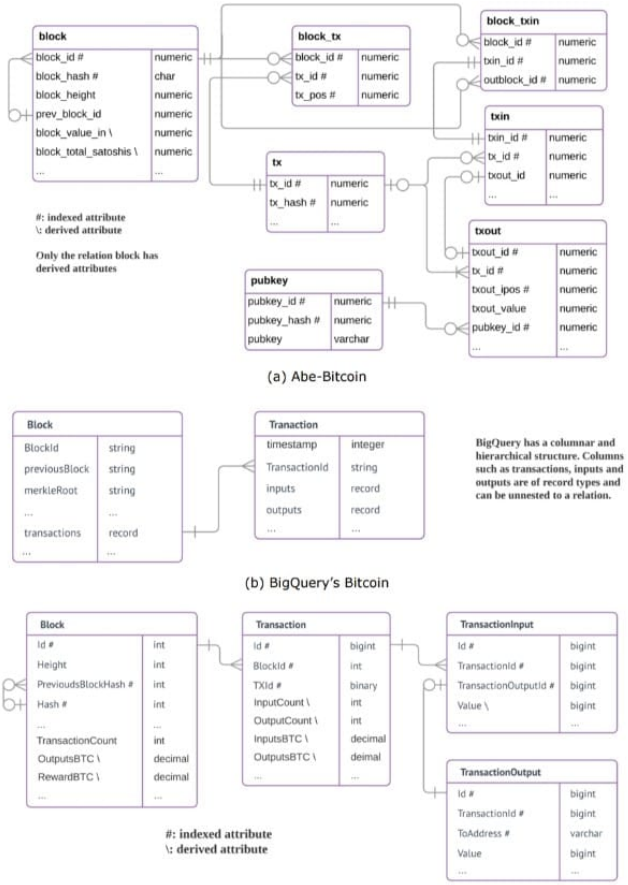
4.1 local bicoin SQL database

Storing the Bitcoin blockchain in a local SQL database allows full control and customization to satisfy diverse needs. Bitcoin blockchain is an append-only database in which the only changeoccurs about every ten minutes when a new block is created. Blocks are stored by Bitcoin Core in data files that do not change (except for the most recent evolving one) and can be parsed to populate a SQL database. There are available open source Bitcoin SQL database options,

such as Abe (2018) and Bitcoin Database Generator (2018). Abe is generally used because it captures more blockchain data, is more popular, and can also be used to store a number of other cryptocurrencies.

To install Abe, it is necessary to install Bitcoin Core to obtain a local copy of the blockchain first. Depending on the connection bandwidth and computer configuration, it may take a few hours to a few weeks to fully synchronize with the Bitcoin network.

Partial ER diagrams of the two methods are shown in the figure below.



The Table here shows some of their basic parameters. Abe has 17 tables and 4 views.

	Abe	BigQuery	bcsql
# tables	17	2	13
# views	4	0	0
# stored derived columns	5	0	15

Table 2 Some Parameters of the Three DB

It is designed to be flexible enough to handle multiple cryptocurrencies. Many tables do not have derived columns that are computed and stored for efficiency. For example, the table txin(txin-id, tx-id, txin-pos, txout-id, txin-scriptsig, txin-sequence) stores information about transaction input. The field txin-id serves as a surrogate primary key, and tx-id and txout-id are foreign keys referencing the transaction containing the txid, and the txout used for the TXIn respectively. The other three columns are basic raw data. Users accessing a TXIn usually needs more than raw basic data and Abe uses a view txin-detail, which has 21 columns to provide contextual and summary data for the TXIn.

To provide some ideas of how queries can be constructed, consider the following three problems, each related to an example transaction:

4.2 Genesis transaction:

Find the (Genesis) block hash from the transaction hash.

```
select b.block_hash
from block b join block_tx bt on (b.block_id = bt.block_id)
join tx t on (bt.tx_id = t.tx_id)
where t.tx_hash = '4a5e1e4baab89f3a32518a88c31bc87f618f76673e2cc77ab2127b7afdeda33b';
```

4.3 Pizza transaction:

find the addresses and amounts of the TXIn from the pizza transaction hash.

```
select i.txin_pos, o.pubkey_hash, o.txout_value
from tx t join txin i on (t.tx_id = i.tx_id)
join txout_detail o on (o.txout_id = i.txout_id)
where t.tx_hash = 'a1075db55d416d3ca199f55b6084e2115b9345e16c5cf302fc80e9d5fbf5d48d'
order by 1 asc;
```

4.4 Pizza-provider transaction:

find the pizza-provider transaction hash, its output addresses, and amounts that used the UTXO of the pizza transaction.

```
    select t2.tx_hash as tx_hash,
p.pubkey_hash as address,
o2.txout_pos as position,
o2.txout_value as amount
from tx t1, txout o1, txin i, tx t2, txout o2, pubkey p
where t1.tx_id = o1.tx_id
and i.txout_id = o1.txout_id
and o2.tx_id = i.tx_id
and i.tx_id = t2.tx_id
and o2.pubkey_id = p.pubkey_id
and t1.tx_hash = 'a1075db55d416d3ca199f55b6084e2115b9345e16c5cf302fc80e9d5fbf5d48d
```