mas.s62 lecture 5 synchronization

2018-02-21

Tadge Dryja

the Bitcoin network

so far we've talked about:

signatures

mining and blocks

transactions and scripts

... now to put it all together

recap: signatures

public / private keys

private key can sign() a message

can verify(public key, message, sig)

useful for proving identity, ownership. Better than paper signatures!

recap: mining and blocks

change a nonce, hash a bunch of times, get a low output. Proves work

Include the previous data as part of your input, and you make a chain of work -- a blockchain

recap: txs and scripts

Transactions have inputs and outputs

inputs	outputs
txid:index (36B) signature (100B)	script (25B) amount (8B)
txid:index signature	script (pubkey) amount

recap: txs and scripts

inputs point to old outputs and have signatures

outputs have scripts and coin amounts

inputs	outputs
txid:index (36B) signature (100B)	script (25B) amount (8B)
txid:index signature	script (pubkey) amount

tx mining process

users make txs, sign, broadcast

someone takes all the txs, puts them in a block, and does work

those txs are now "confirmed", and the next block can be built

tx mining: header

the block header is the message which must satisfy the proof of work

Headers have a hash of the txs in the block

Really it's the headers that make a chain, not the blocks. Headerchain.

tx mining: header

headers are 80 bytes; similar to pset02 blocks

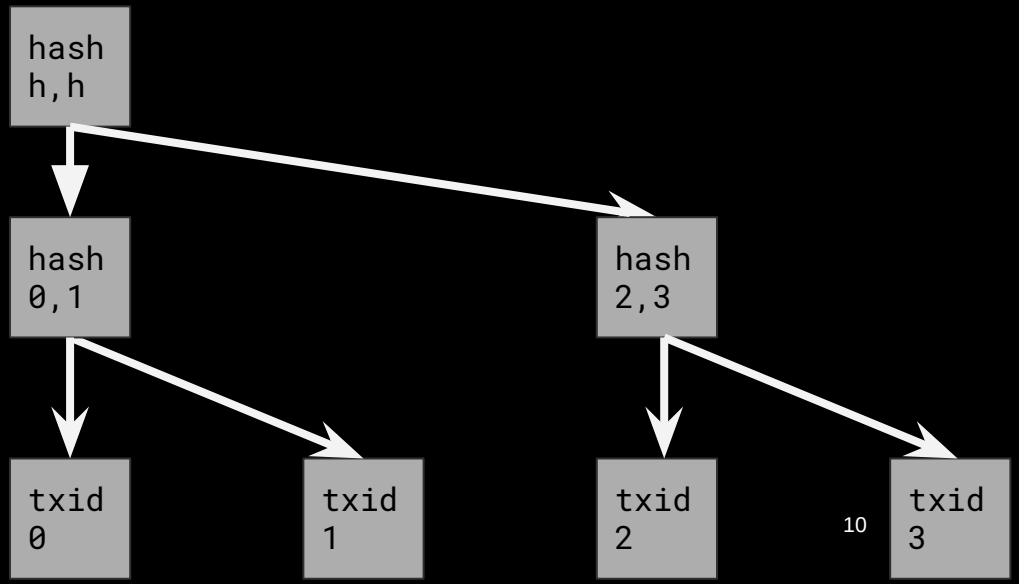
Main components are

prev hash, merkle root, nonce

merkle root recap

Hash in a binary tree

Same level of commitment as h(0,1,2,3)



header fields

version	4B	indicates block version
prev hash	32B	Was used for fork signalling;
merkle root	32B	future use
time	4B	unclear
diff	4B	
nonce	4B	

header fields

version	4B	hash of previous
prev hash	32B	block
merkle root	32B	
time	4B	
diff	4B	
nonce	4B	

header fields

version	4B	hash of all
prev hash	32B	transactions in
merkle root	32B	the block
time	4B	
diff	4B	
nonce	4B	

header fields

version	4B	
prev hash	32B	
merkle root	32B	
time	4B	unix time (seconds since 1970) of claimed block creation
diff	4B	
nonce	4B	(can be before previous block's time!)

header fields

version	4B	PoW target in a weird floating point format
prev hash	32B	
merkle root	32B	
time	4B	
diff	4B	pretty much useless as can be computed anyway
nonce	4B	

header fields

version	4B
prev hash	32B
merkle root	32B
time	4B
diff	4B
nonce	4B

nonce - anything goes here

but there's a problem...

header fields

version	4B
prev hash	32B
merkle root	32B
time	4B
diff	4B
nonce	4B

nonce - anything goes here

but there's a problem...

too small!

header fields

version	4B	2^{32} possible
prev hash	32B	nonces
merkle root	32B	
time	4B	But current
diff	4B	blocks need 2^{70}
nonce	4B	work!

header fields

version	4B	adjust time
prev hash	32B	
merkle root	32B	modify merkle
time	4B	root
diff	4B	
nonce	4B	

tx order in block

tx0 is the coinbase tx:

generates new coins, and takes fees from all other txs in block

all other txs can be in any order, but can only spend outputs from previous txs

tx order in block

if txB spends an output of txA, then txA must come first in block ordering

this ensures linear verification of transactions can proceed

intermission

256 second break

prove work by moving body mass against force of gravity

work = f*d = m*g*h

sync process

I just downloaded bitcoin!

What's been going on for the last 9 years?

sync process

Download binary / compile code

Verify GPG signatures somehow...

Hardcoded DNS seeds to find peers

connect, ask for headers

download & verify 500K headers

sync process

Get the header chain first - quick

takes under a minute with good connections

verify all the work before any signatures

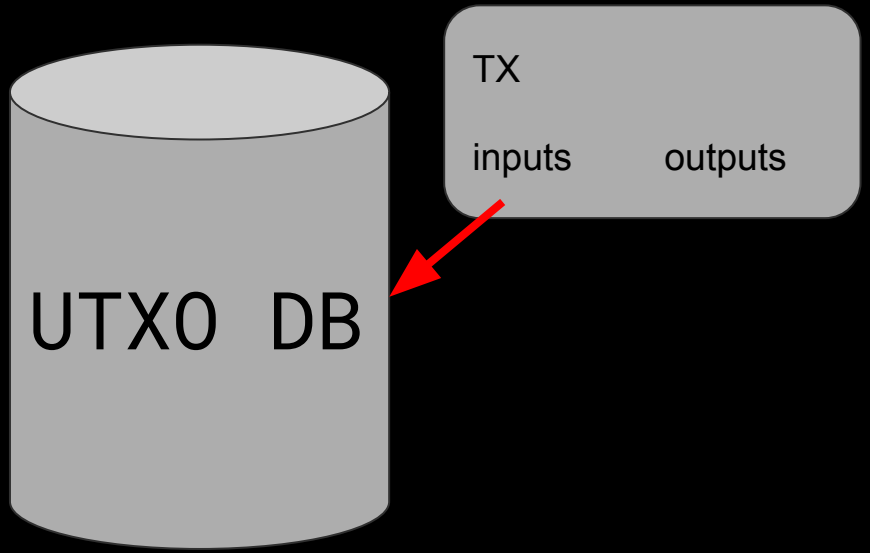
sync IBD

After headers, Initial Block Download (IBD)

Request blocks from peers, match tx list to merkle root in header, process each tx in order

sync IBD

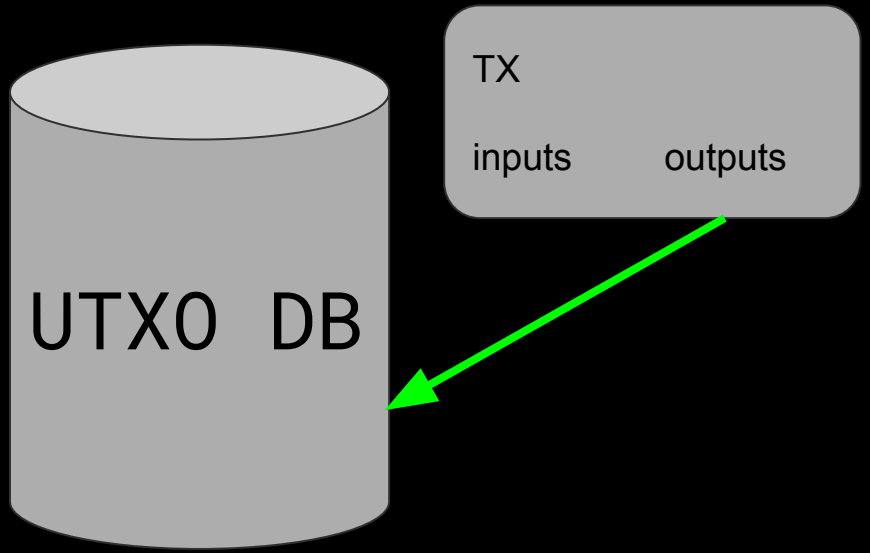
Delete all input txos



sync IBD

Delete all input txos

Add output txos



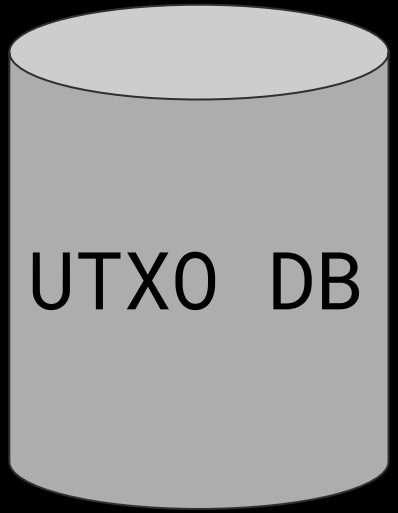
sync IBD

Do this ~300M times

Downloads 170GB

End result:

55M txos, ~3.2GB



pruning

By default, store all 500K blocks

Can serve to others who need to IBD

But can "prune" / delete blocks after IBD with no loss of security

Downside?

pruning

By default, store all 500M blocks

Can serve to others who need to IBD

But can "prune" / delete blocks after IBD with no loss of security

Downside? Not everyone can prune

blockchain data

What does it store?

banlist.dat
chainstate
peers.dat
wallet.dat
bitcoin.conf
blocks
database
debug.log
mempool.dat

blockchain data

What does it store?

```
banlist.dat  1.8K    bad nodes
chainstate
peers.dat    4.0M    good nodes
wallet.dat      1.4M    my precious
bitcoin.conf 144     config file
blocks
database
debug.log    11M     log file, rotates
mempool.dat  20M        more like diskpool
```

blockchain data

What does it store?

```
banlist.dat  1.8K
chainstate     3.0G    utxo set
peers.dat    4.0M
wallet.dat     1.4M
bitcoin.conf 144
blocks         183G    all the
blocks
database     80K     ? nothing?
debug.log    11M
mempool.dat  20M
```

blockchain as database

186GB, but a really crummy database

remember tx `9e95c3c3c96f57527cdc649550bf8e92892f7651f718d846033798aee333b0c3`

from back in 2014?

blockchain as database

186GB, but a really crummy database

remember tx `9e95c3c3c96f57527cdc649550bf8e92892f7651f718d846033798aee333b0c3`

from back in 2014?

No. It's somewhere in the blocks folder but I don't know where.

It's not in chainstate

blockchain as database

how about output

02b1500a0f3b059819dd923f1c78bacc0a3de303fc51836ce7f46a3206b29ba7:0

it's an op_return output, can you tell me what the data is?

blockchain as database

how about output

02b1500a0f3b059819dd923f1c78bacc0a3de303fc51836ce7f46a3206b29ba7:0

it's an op_return output, can you tell me what the data is?

Nope! op_return outputs don't get stored in the chainstate.

blockchain as database

Hey I have a pubkey with hash

1d493f9536c692d096536ba9d1c081feabd7ccf3

how many coins do I have? How many outputs?

blockchain as database

Hey I have a pubkey with hash
1d493f9536c692d096536ba9d1c081feabd7ccf3

how many coins do I have? How many outputs?

No idea! Gotta search through all of chainstate. Doesn't index based on PkScript, only txid:index

blockchain as database

how many coins does output

7434e09a302eaa4e2e0826aea08c2cca282a8bfc606cb680aa1f3f331a7e4f69:1

have?

blockchain as database

how many coins does output

7434e09a302eaa4e2e0826aea08c2cca282a8bfc606cb680aa1f3f331a7e4f69:1

have?

Lots! 239.99913132. It's in the utxo set because it hasn't been spent yet.

Can quickly find based on txid:index

blockchains are bad databases

Only keeps track of utxos, which is hard enough

Can add further indexes, but they take lots of space. Most common is "address index" so people can ask if they have any money.

blockchains are bad databases

DB queries not given to network peers

Network peers are scary, ban them if they act funny

Provide headers, blocks, txs, other nodes IPs

bad DB but good consensus

Everyone's got the same utxo set

Even though they all really want more utxos. Or to break the system. It seems to work.

pset02 update

Bunch of blocks mined

Pls reduce server queries; an 18. address is doing 5+ TCP connections per second. Also GCE? 35.

Could use blocks here to start a coin...

MAS.S62 Cryptocurrency Engineering and Design

Spring 2018