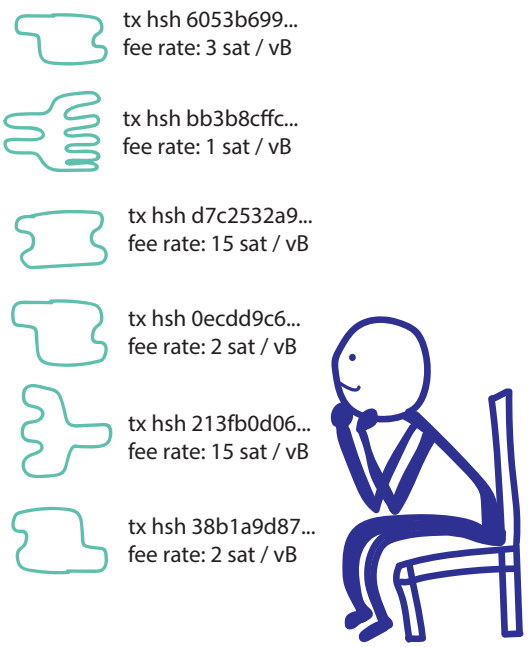
@amizi

what's a MEMPOOL?

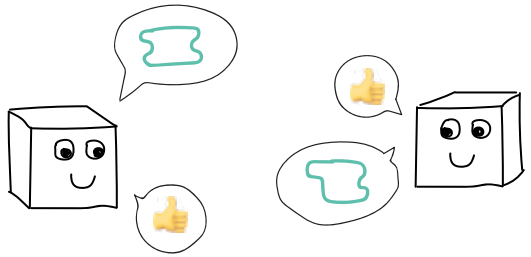
bitcoin core fundamentals

a mempool is where transactions wait to be confirmed into a block

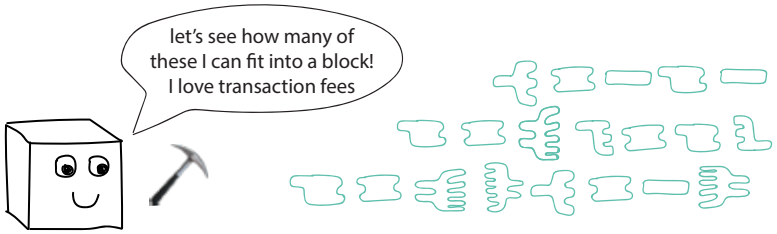


the main purpose of a mempool is to...

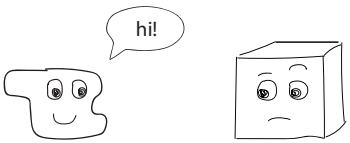
(a) relay unconfirmed transactions



(b) provide miners transactions to mine



when a node first receives a transaction from a peer,



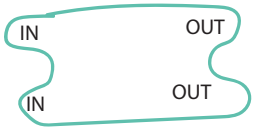
it has to verify the transaction is legit. nobody wants faulty or deceptive transactions.

Accept To Memory Pool (ATMP) involves checking things like...

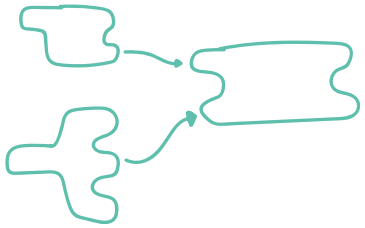
- do I already have this transaction?
- is there a conflict with a different transaction in the mempool?
- does the ₿ in cover the ₿ out?
- do the signatures prove the previous outputs can be spent?
- are there enough fees?

what's in a mempool?

this is a transaction with
2 inputs and 2 outputs

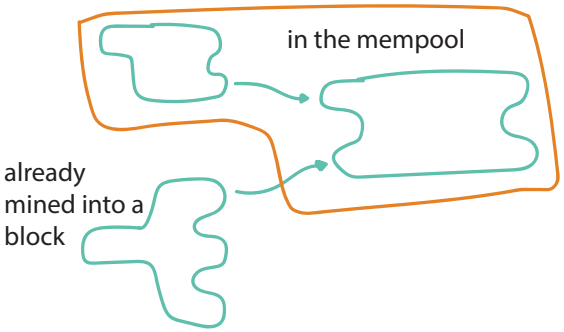


the inputs spend the outputs
of a previous transaction

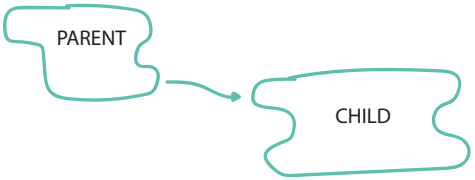


let's say one of those previous
transactions is also in the mempool

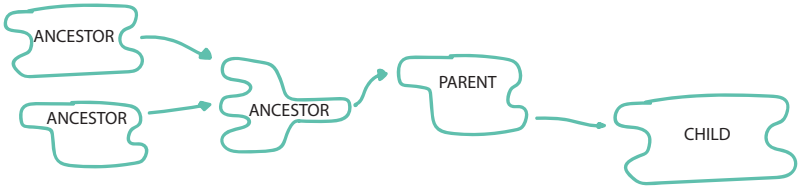
we call this a mempool package



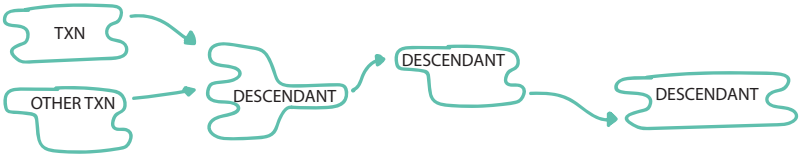
sometimes we refer to them as child & parent



in longer chains, we see ancestors



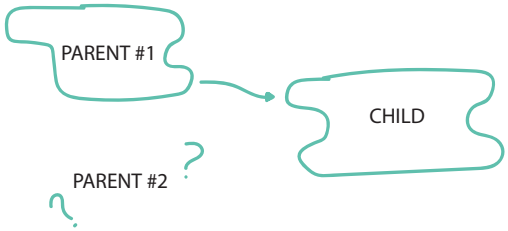
and if you frame it differently, we see descendants



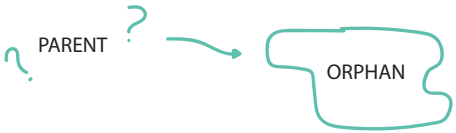
in bitcoin core, mempool packages
are limited to 25 transactions

what's in a mempool?

sometimes we receive a child but are missing a parent



we call these transactions orphans



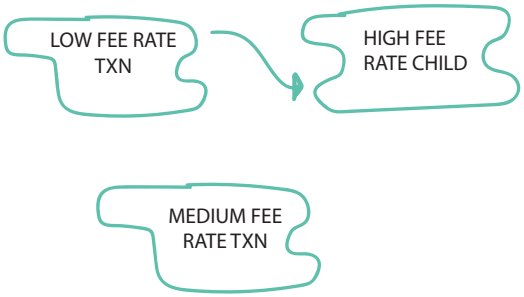
without all of its parents, we cannot validate a transaction and accept it into our mempool yet

so we keep them in an orphan pool

and request the parent transactions from our peers

when miners are deciding what transactions to select, they look at the fee-rate of the entire package

end users can utilize this to bump the fee rate of a mempool transaction (to get it mined faster)



this is called child pays for parent

what's MY mempool?

configure your mempool via command line!

-maxmempool=<n>

limit the memory usage of mempool transactions to a max of <n> megabytes (default: 300)

-maxorphantx=<n>

remember a max of <n> transactions that are missing parents (default: 100)

-mempoolexpiry=<n>

don't keep transactions that are more than <n> hours old (default: 2 weeks)

-blocksonly

disables your mempool (default: false)

-persistmempool

save your mempool to disk when you shutdown your node & load when you restart (default: save)

-minrelaytxfee=<amt>

set a minimum fee rate of transactions you want. don't accept anything less than <amt> BTC/kB, and request your peers not to send them to you (default: 1 satoshi/byte)

& learn more via your logs!

if you have a node running, you can enable additional mempool logging by running

```
bitcoin-cli logging "[\"mempool\"]"
```

and check that its enabled by running

```
bitcoin-cli logging
```

you should see a list of log categories, with

```
"mempool": true
```

this will add output to your `stdout.log` that inform you when certain events happen, such as...

- transaction was accepted into the mempool
- transaction removed from the mempool & for what reason (eg. expiry time, evicted due to low fee rate)
- orphan transaction ignored, stored, or removed

AND LOTS MORE! what will you find?