CMPSC 390
Bitcoin Transactions

Janyl Jumadinova

Credit: Authors of "Bitcoin and Cryptocurrency Technologies"

January 28, 2021

Where we left off ...

Bitcoin consensus

- Append-only ledger.
- Decentralized consensus.
- Miners to validate transactions.

Where we left off ...

Bitcoin consensus

- Append-only ledger.
- Decentralized consensus.
- Miners to validate transactions.

assuming a currency exists to motivate miners!

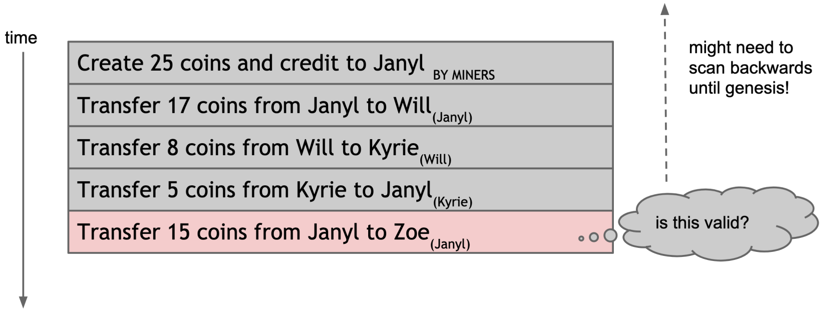
UTXO Model

- Unspent Transaction Output Model.

UTXO Model

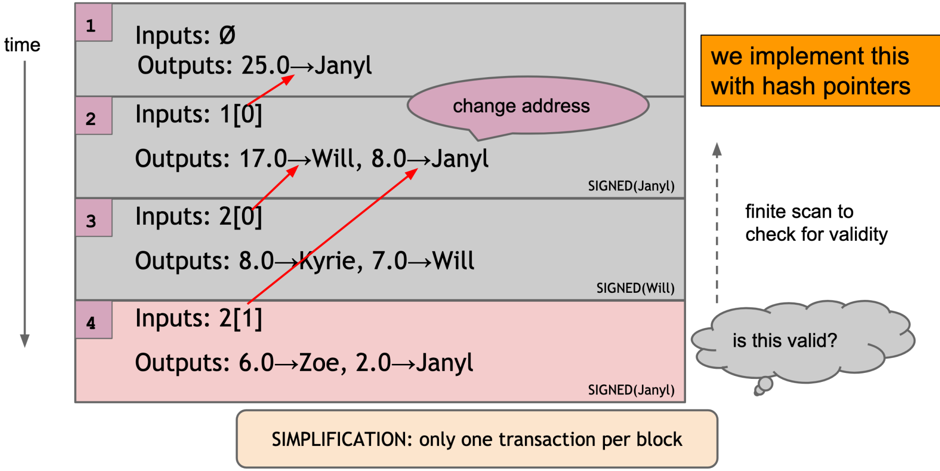
- Unspent Transaction Output Model.
- Transactions map inputs to outputs.
- An account holds a set of)
  - Transactions contain signature of fund's owner.
  - Spending bitcoin is redeeming previous transaction outputs.

An account-based ledger (not Bitcoin)

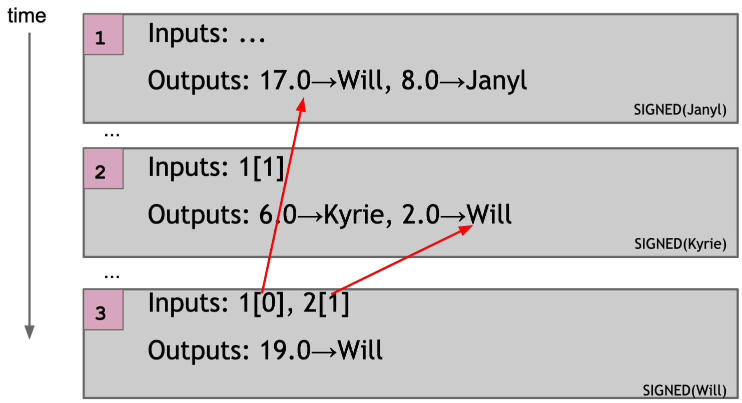

time
Create 25 coins and credit to Janyl BY MINERS
Transfer 17 coins from Janyl to Will (Janyl)
Transfer 8 coins from Will to Kyrie (Will)
Transfer 5 coins from Kyrie to Janyl (Kyrie)
Transfer 15 coins from Janyl to Zoe (Janyl)
might need to scan backwards until genesis!
is this valid?

SIMPLIFICATION: only one transaction per block

Transaction-based ledger (Bitcoin)

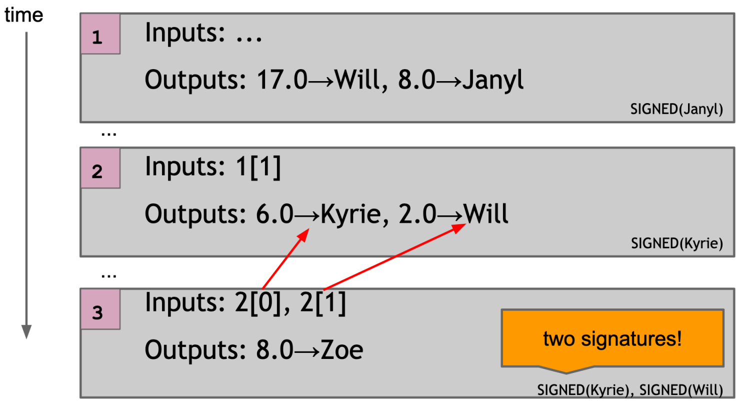

time
1
Inputs: Ø
Outputs: 25.0→Janyl
2
Inputs: 1[0]
Outputs: 17.0→Will, 8.0→Janyl
SIGNED(Janyl)
change address
3
Inputs: 2[0]
Outputs: 8.0→Kyrie, 7.0→Will
SIGNED(Will)
4
Inputs: 2[1]
Outputs: 6.0→Zoe, 2.0→Janyl
SIGNED(Janyl)
we implement this with hash pointers
finite scan to check for validity
is this valid?
SIMPLIFICATION: only one transaction per block

Merging Value



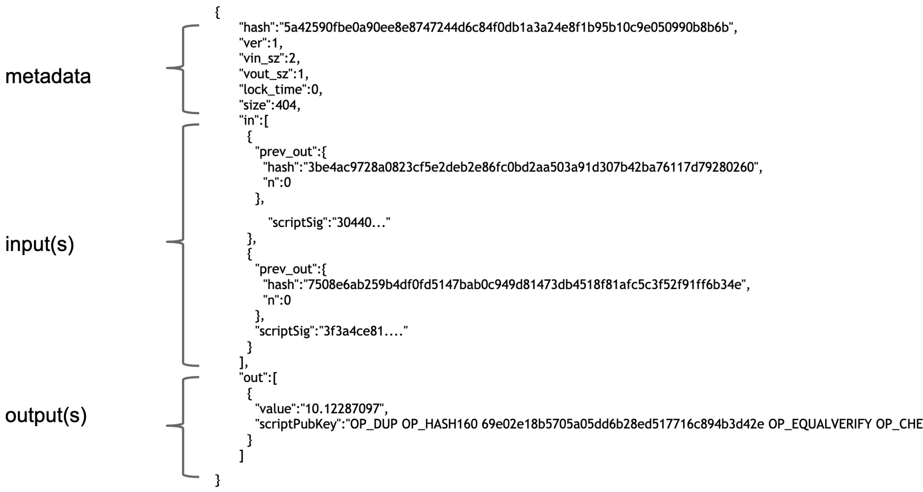
SIMPLIFICATION: only one transaction per block

Joint Payments



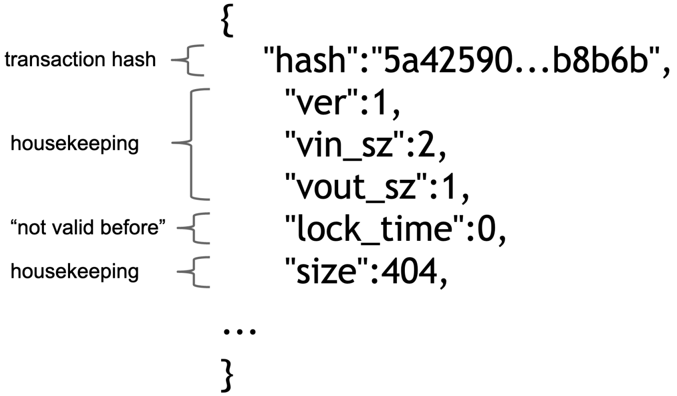
SIMPLIFICATION: only one transaction per block

Bitcoin transaction



```
{
   "hash":"5a42590fbe0a90ee8e8747244d6c84f0db1a3a24e8f1b95b10c9e050990b8b6b",
   "ver":1,
   "vin_sz":2,
   "vout_sz":1,
   "lock_time":0,
   "size":404,
   "in":[
     {
       "prev_out":{
         "hash":"3be4ac9728a0823cf5e2deb2e86fc0bd2aa503a91d307b42ba76117d79280260",
         "n":0
       },
         "scriptSig":"30440..."
     },
     {
       "prev_out":{
         "hash":"7508e6ab259b4df0fd5147bab0c949d81473db4518f81afc5c3f52f91ff6b34e",
         "n":0
       },
       "scriptSig":"3f3a4ce81...."
     }
   ],
   "out":[
     {
       "value":"10.12287097",
       "scriptPubKey":"OP_DUP OP_HASH160 69e02e18b5705a05dd6b28ed517716c894b3d42e OP_EQUALVERIFY OP_CHE
     }
   ]
}
```

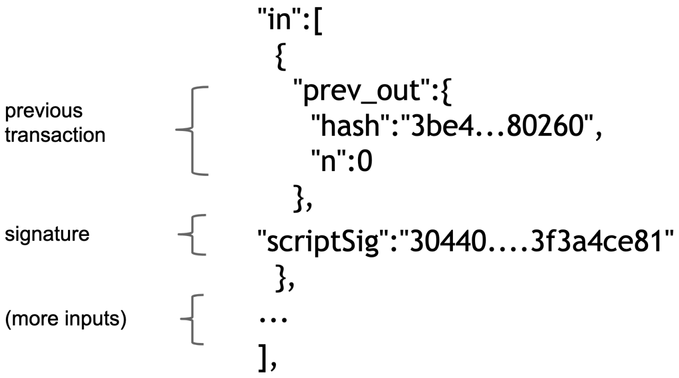
Bitcoin transaction: metadata



```
{
    "hash":"5a42590...b8b6b",
      "ver":1,
      "vin_sz":2,
      "vout_sz":1,
      "lock_time":0,
      "size":404,
...
}
```

transaction hash
housekeeping
“not valid before”
housekeeping

Bitcoin transaction: inputs



```
"in":[
 {
   "prev_out":{
     "hash":"3be4...80260",
     "n":0
   },
"scriptSig":"30440....3f3a4ce81"
 },
...
],
```

Bitcoin transaction: outputs

```
"out":[
      {
        "value":"10.12287097",
        "scriptPubKey":"OP_DUP OP_HASH160 69e...3d42e
   OP_EQUALVERIFY OP_CHECKSIG"
      },
      ...
    ]
```

output value

recipient address??

(more outputs)

Bitcoin Script

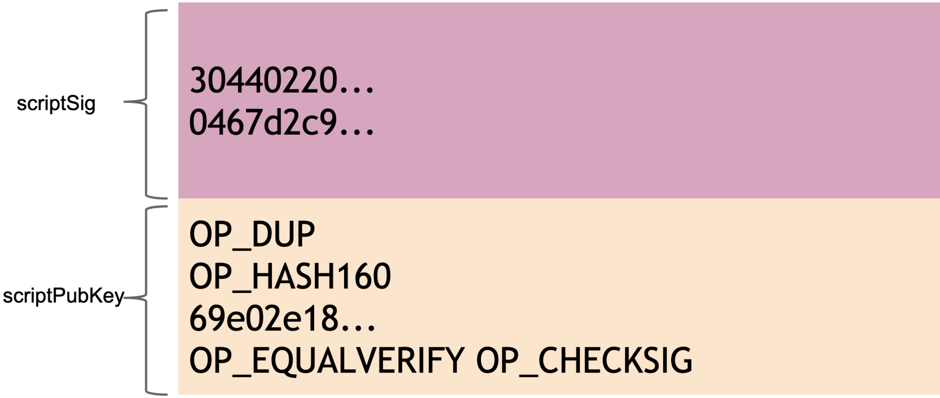
- Output “addresses” are actually scripts.

Bitcoin Script

- Output "addresses" are actually scripts.
- Input "addresses" are also scripts.

Bitcoin Script

- Output "addresses" are actually scripts.
- Input "addresses" are also scripts.



```
30440220...
0467d2c9...
OP_DUP
OP_HASH160
69e02e18...
OP_EQUALVERIFY OP_CHECKSIG
```

Bitcoin Scripting Language ("Script")

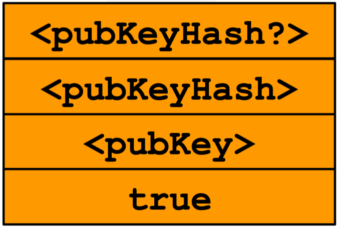
- Built for Bitcoin (inspired by Forth).
- Simple, compact.
- Support for cryptography.
- Stack-based.
- Limits on time/memory.
- No looping.

Bitcoin Script Example

```
<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash?>
OP_EQUALVERIFY OP_CHECKSIG
```

Bitcoin Script Example

```
<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash?>
OP_EQUALVERIFY OP_CHECKSIG
```



Bitcoin Scripting instructions

256 opcodes total

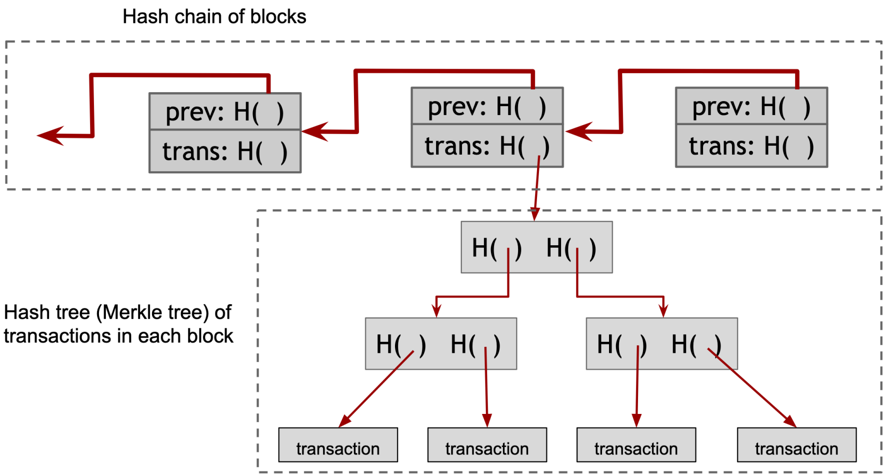
- Arithmetic
- If/then
- Logic/data handling

Bitcoin Scripting instructions

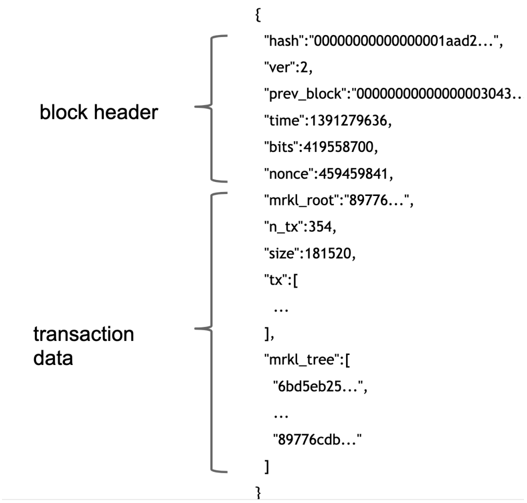
256 opcodes total

- Arithmetic
- If/then
- Logic/data handling
- Hashes. Signature verification. Multi-signature verification

Bitcoin Blocks

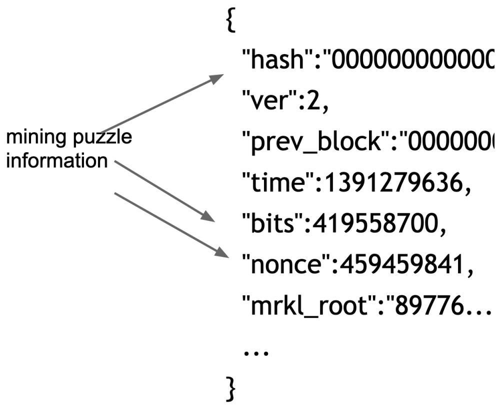


Bitcoin Blocks



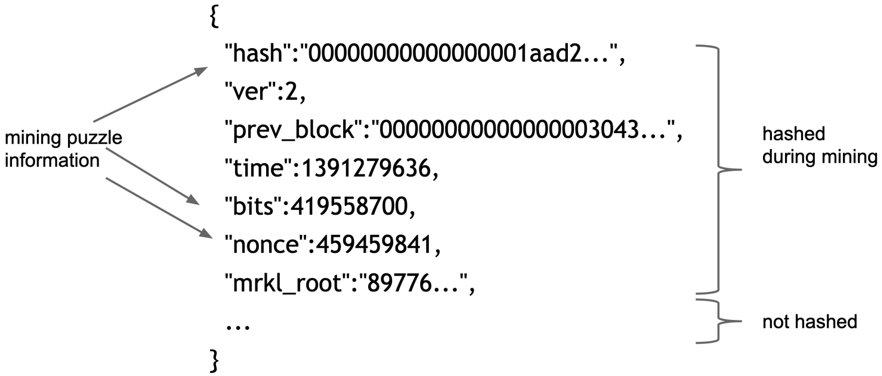
```
{
  "hash":"0000000000000001aad2...",
  "ver":2,
  "prev_block":"00000000000000003043..
  "time":1391279636,
  "bits":419558700,
  "nonce":459459841,
  "mrkl_root":"89776...",
  "n_tx":354,
  "size":181520,
  "tx":[
    ...
  ],
  "mrkl_tree":[
    "6bd5eb25...",
    ...
    "89776cdb..."
  ]
}
```

Bitcoin Blocks



```
{
  "hash":"00000000000C
  "ver":2,
  "prev_block":"000000C
  "time":1391279636,
  "bits":419558700,
  "nonce":459459841,
  "mrkl_root":"89776...
  ...
}
```

Bitcoin Blocks



```
{
  "hash":"0000000000000001aad2...",
  "ver":2,
  "prev_block":"0000000000000003043...",
  "time":1391279636,
  "bits":419558700,
  "nonce":459459841,
  "mrkl_root":"89776...",
  ...
}
```