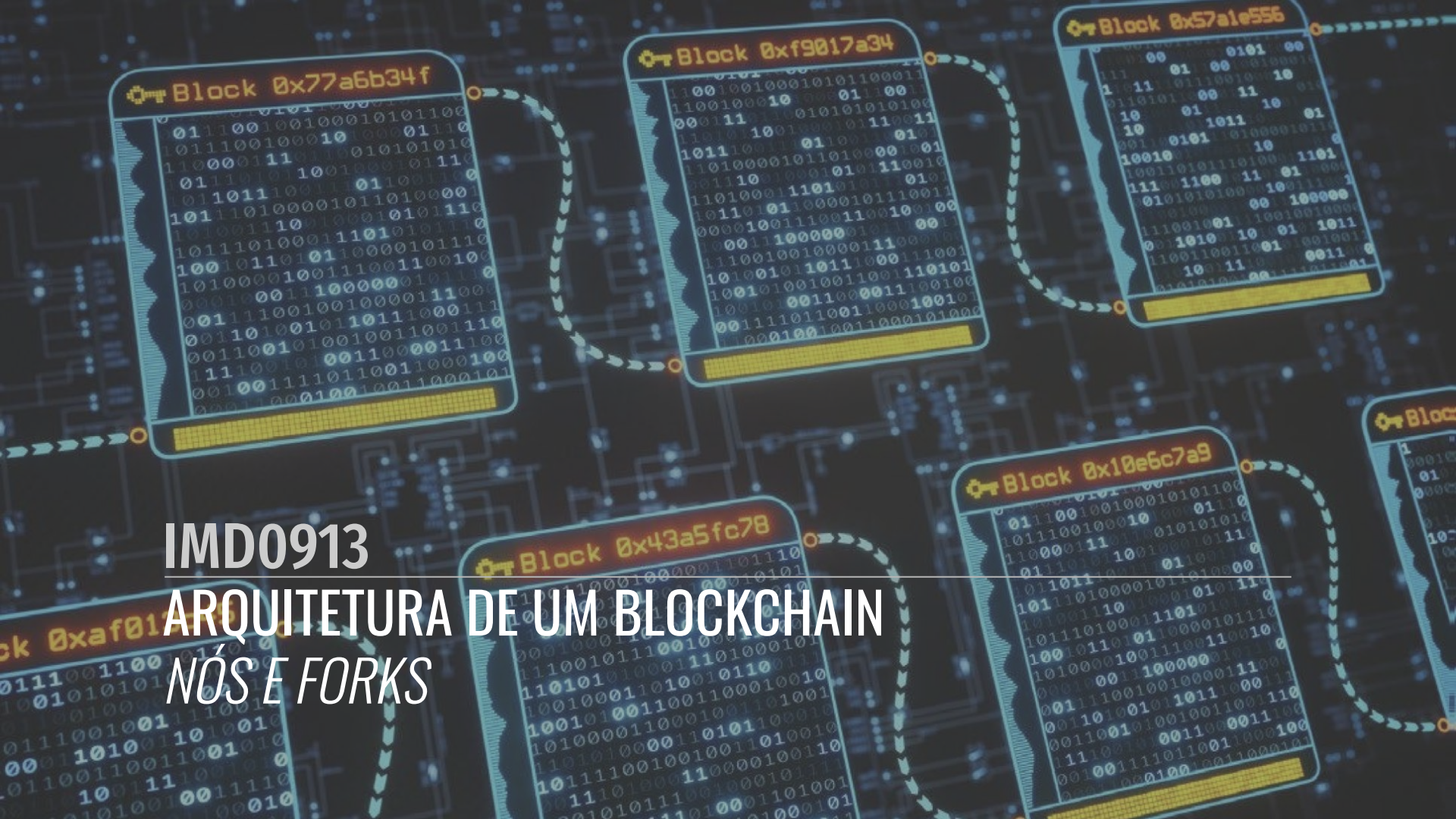
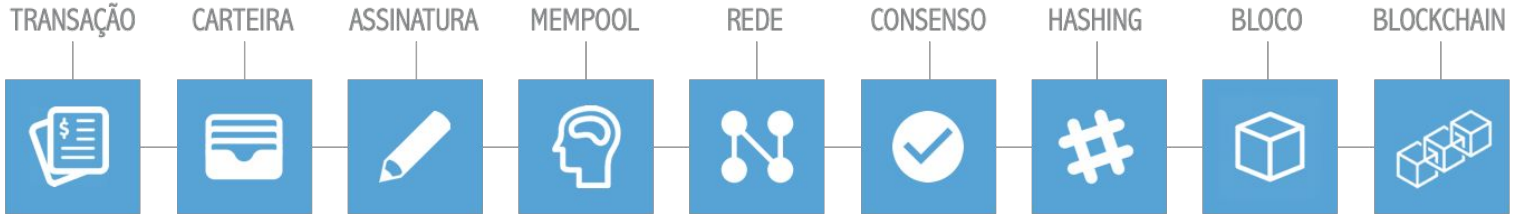


IMD0913

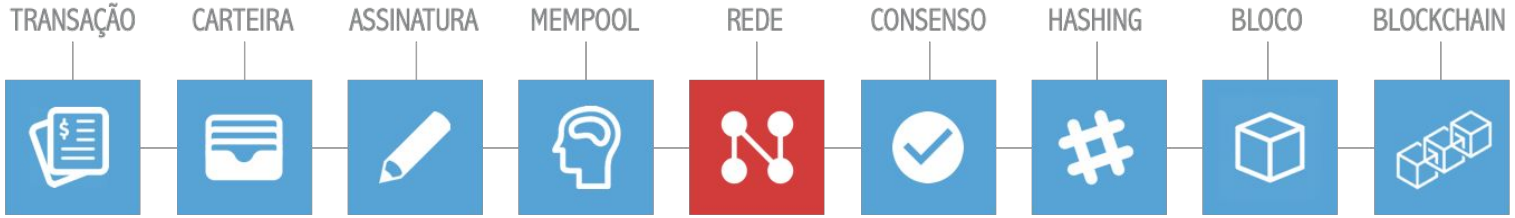
ARQUITETURA DE UM BLOCKCHAIN

NÓS E FORKS

ARQUITETURA DE UM **BLOCKCHAIN**



ARQUITETURA DE UM **BLOCKCHAIN**



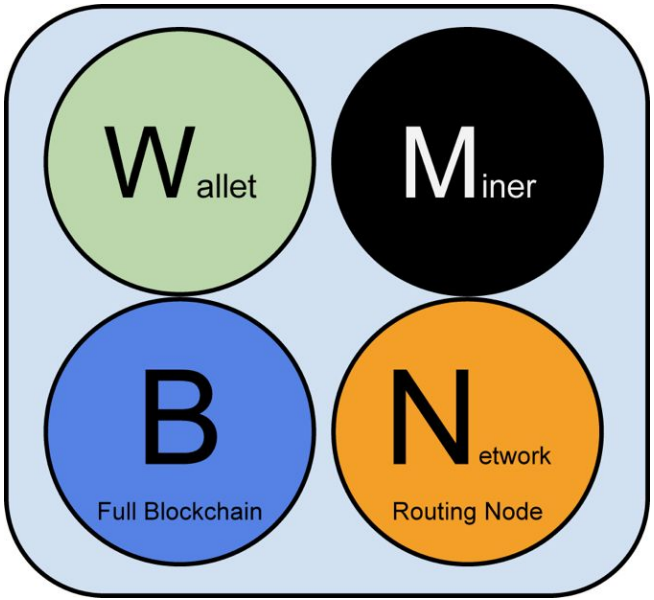
Rede Bitcoin

Um *blockchain* é suportado por uma rede distribuída *peer-to-peer*

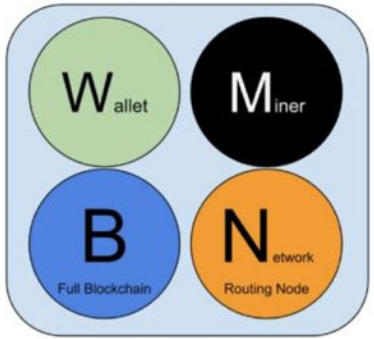
- **Não existe** o papel de servidor
- **Não existe** serviço centralizado
- **Não existe** hierarquia na rede

Rede Bitcoin se refere a coleção de nós executando o protocolo P2P Bitcoin

Rede Bitcoin: Tipos e perfis de nós



Tipos de usuários



Nem todo cliente é minerador

E se eu não tiver um computador potente?

Nem todo cliente tem todo o *blockchain*

E se eu quiser enviar bitcoins do meu celular?

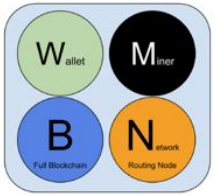
Nem todo cliente está diretamente conectado a rede Bitcoin

E se eu não preciso fazer transações regularmente?

Nem todo cliente tem um carteira

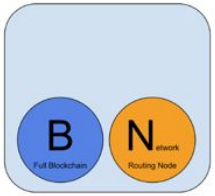
E se eu tiver um cliente de carteira separado?

Rede Bitcoin: Tipos de nós



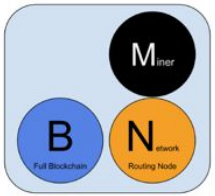
Cliente de referência (Bitcoin Core)

Contém uma Carteira (W), Minerador (M), o blockchain completo (B) e é um nó P2P da rede Bitcoin (N)



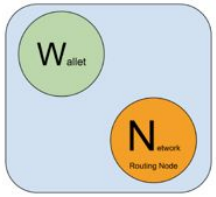
Nó full do blockchain

Contém o blockchain completo (B) e é um nó P2P da rede Bitcoin (N)



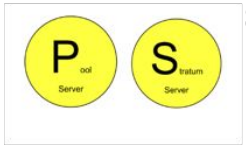
Minerador solo

Minerador (M) que contém o blockchain completo (B) e é um nó P2P da rede Bitcoin (N)



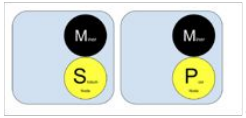
Carteira leve (SPV)

Contém uma carteira (W) e é um nó P2P da rede Bitcoin (N)



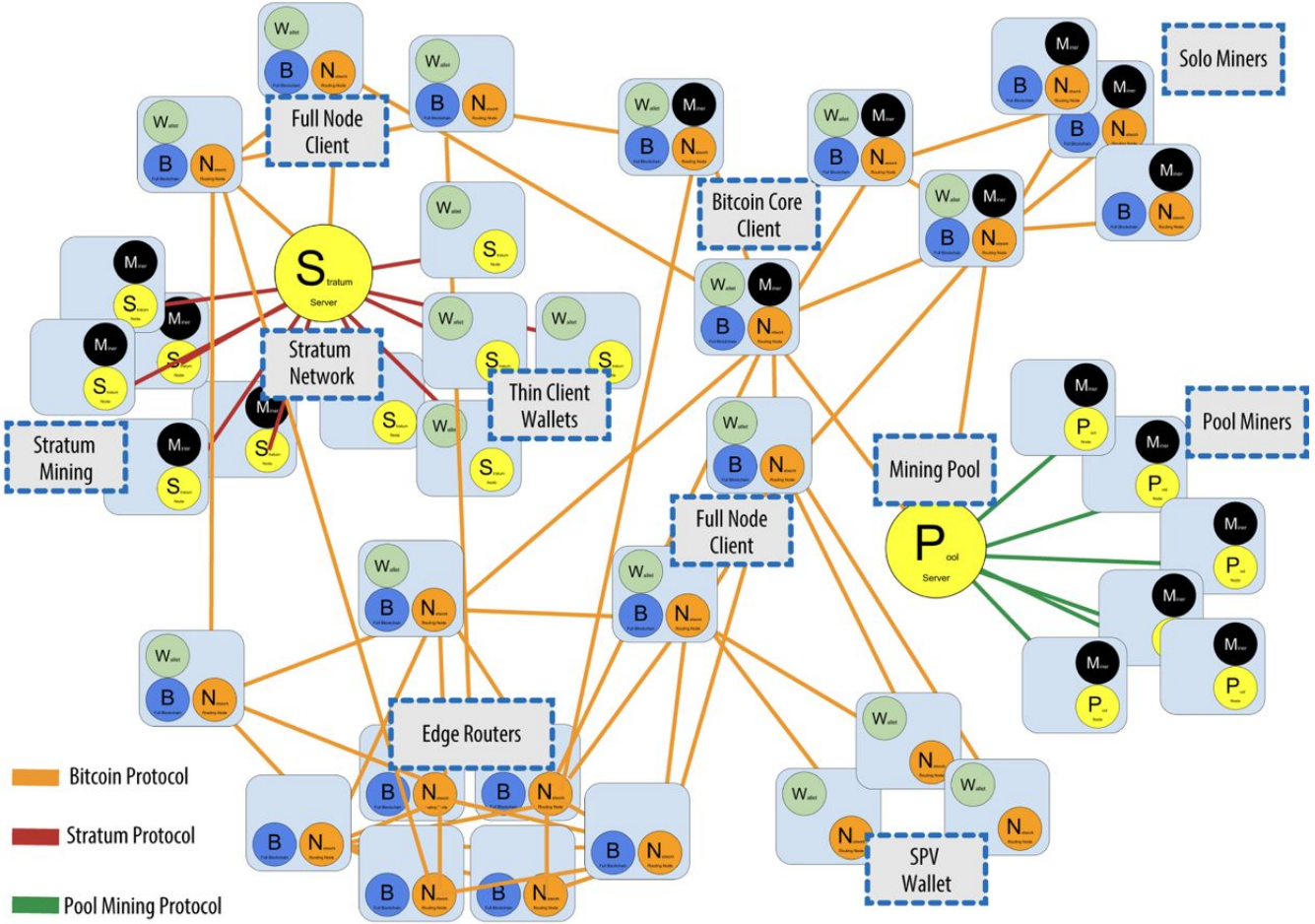
Servidores de protocolos de pool

Gateways conectando a rede P2P bitcoin aos nós que executam outros protocolos como nós de pool ou nós Stratum



Nós de mineração

Contém a função de mineração, sem o blockchain, com o protocolo Stratum (S) ou outro protocolo de nó de pool (P)


Full Node Client
Solo Miners
Bitcoin Core Client
Stratum Server
Stratum Network
Thin Client Wallets
Stratum Mining
Full Node Client
Mining Pool
Pool Server
Pool Miners
Edge Routers
SPV Wallet
Bitcoin Protocol
Stratum Protocol
Pool Mining Protocol

Rede Bitcoin

Quando um novo nó é iniciado, ele precisa descobrir nós bitcoins para se conectar!

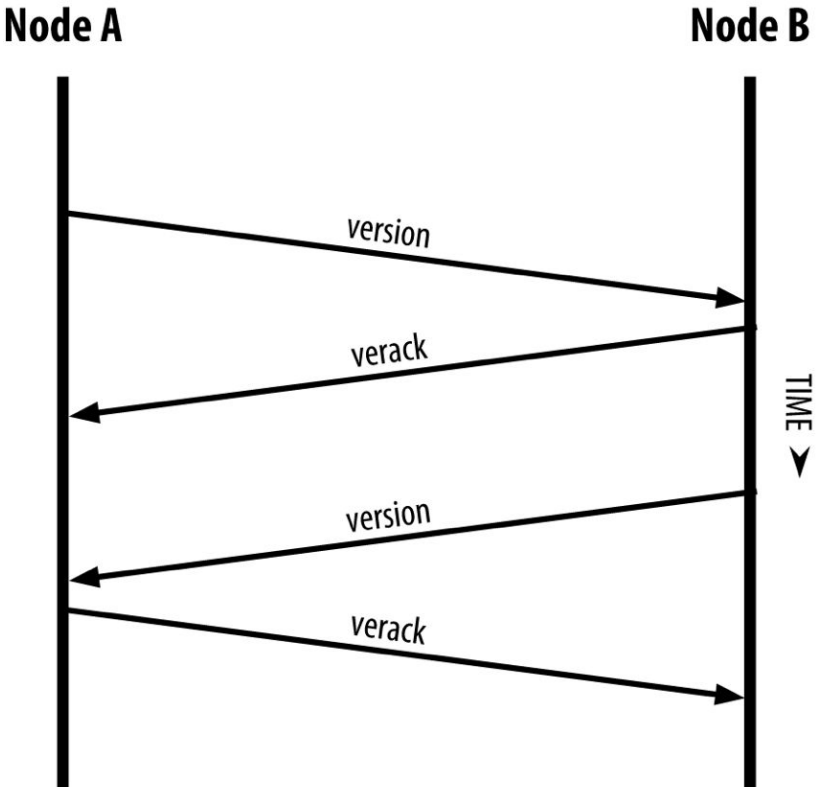
Porta TCP 8333

Opções:

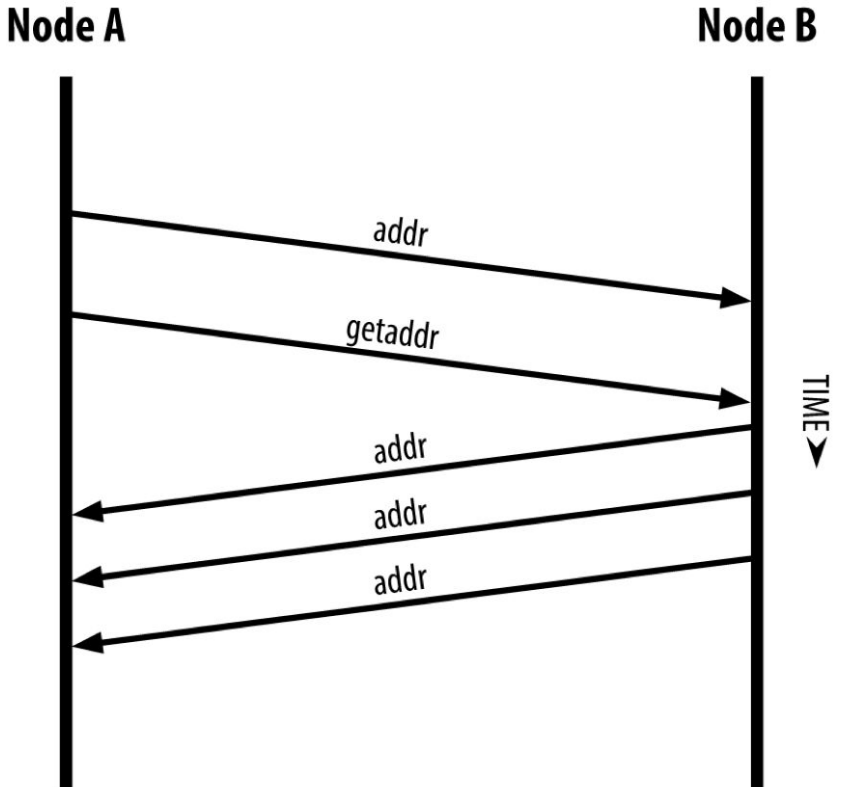
- Alguns servidores conhecidos...
- Ou indicar o endereço de um nó conhecido.

https://bitnodes.earn.com/

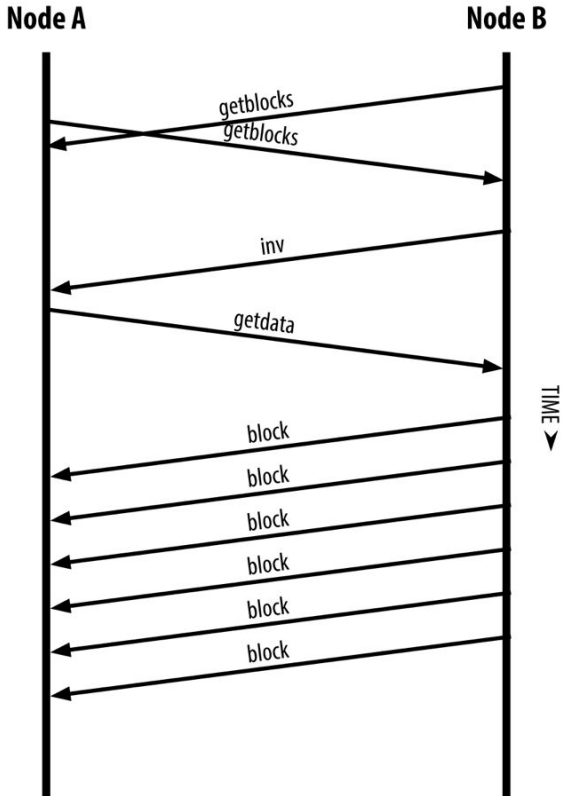
Rede Bitcoin



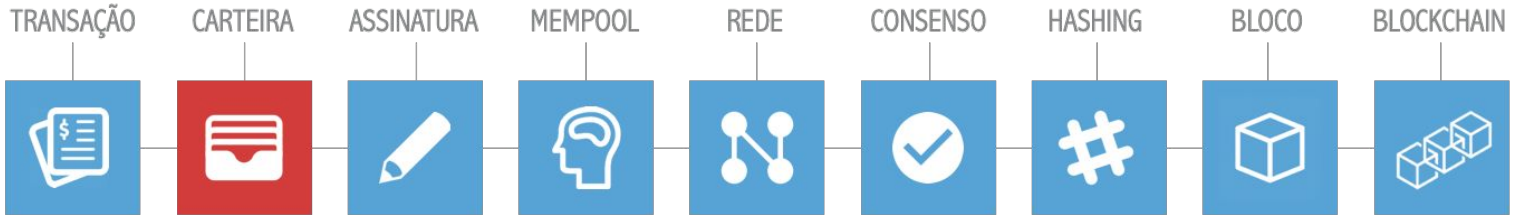
Rede Bitcoin



Rede Bitcoin



ARQUITETURA DE UM **BLOCKCHAIN**



Nós SPV

Nem todo nó armazena o *blockchain* completo

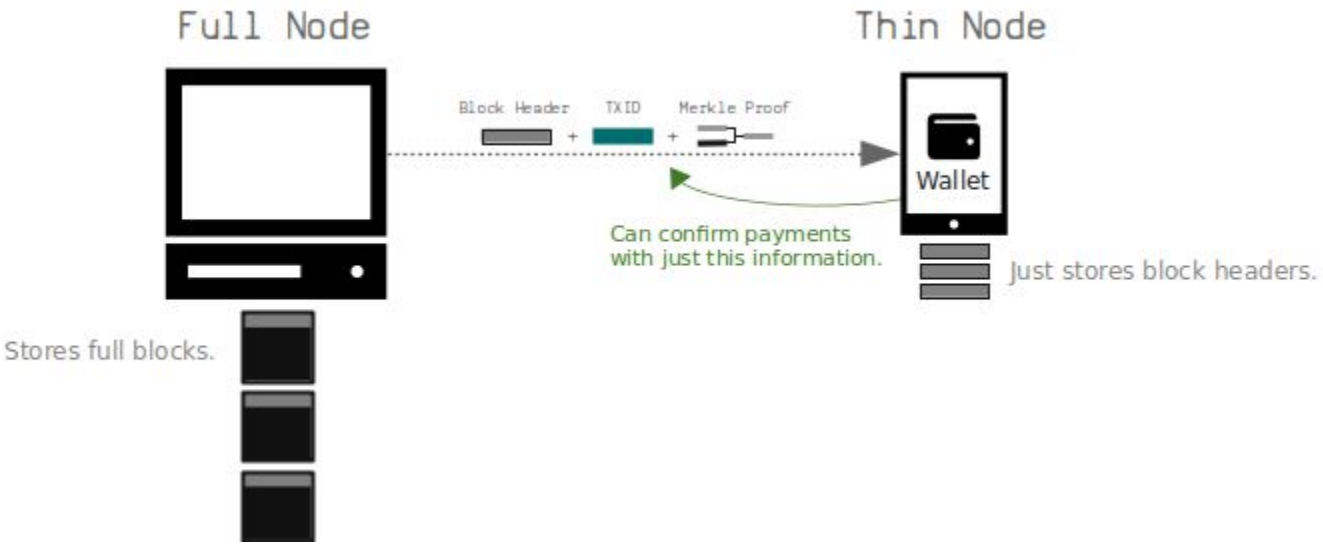
Por exemplo, seu smartphone!

Simple Payment Verification (SPV) é um método de verificar se determinada transação está incluída em um bloco sem precisar baixar o bloco completo

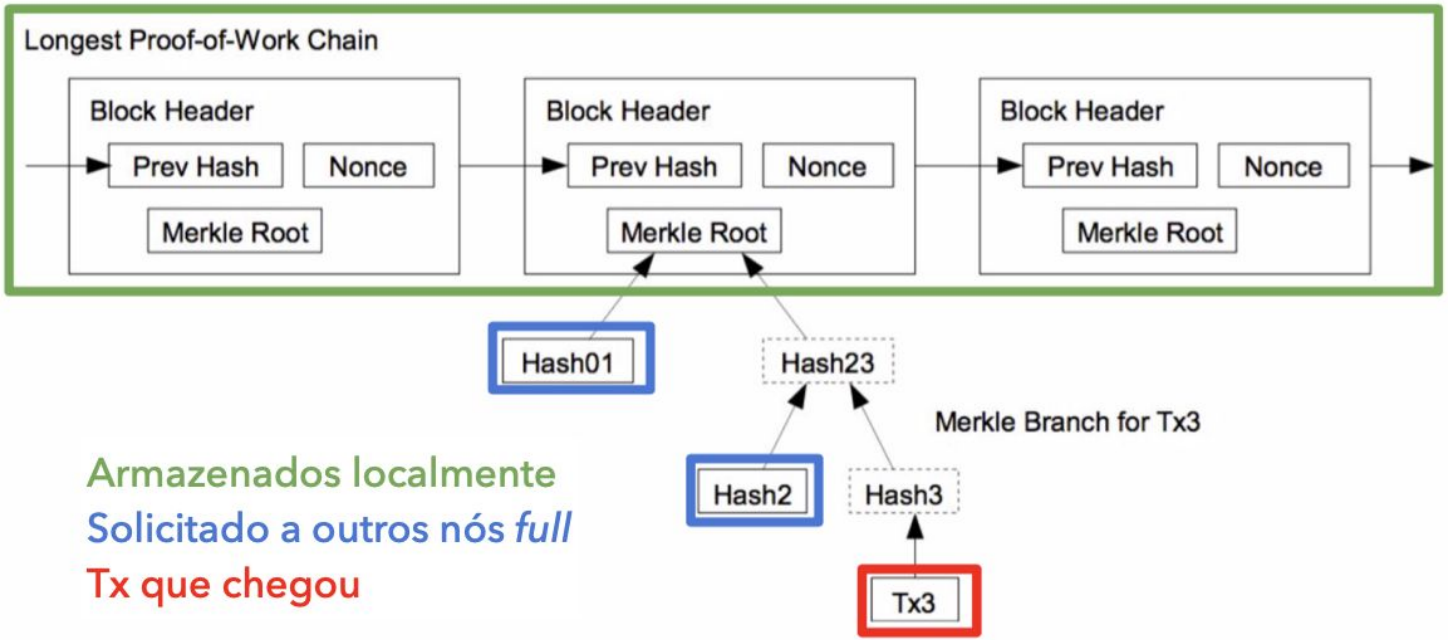
Baixa somente os cabeçalhos dos blocos (1000x menor)

Clientes **leves** (*lightweight*)

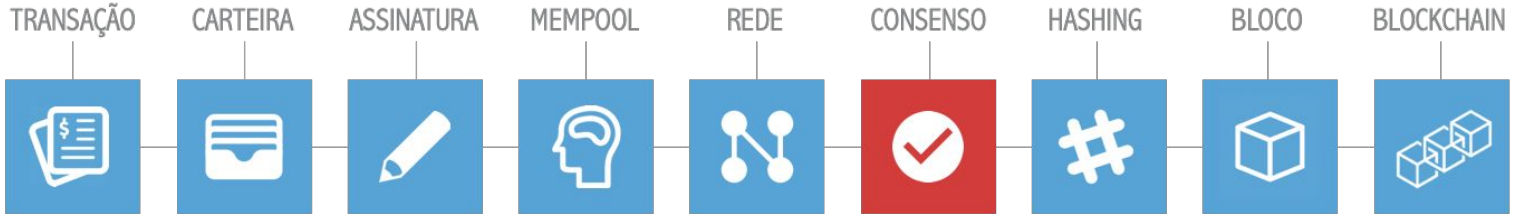
Nós SPV


Full Node
Thin Node
Block Header
TXID
Merkle Proof
Wallet
Can confirm payments
with just this information.
Just stores block headers.
Stores full blocks.

Nós SPV


Longest Proof-of-Work Chain
Block Header
Prev Hash
Nonce
Merkle Root
Block Header
Prev Hash
Nonce
Merkle Root
Block Header
Prev Hash
Nonce
Merkle Root
Hash01
Hash23
Merkle Branch for Tx3
Hash2
Hash3
Tx3
Armazenados localmente
Solicitado a outros nós full
Tx que chegou

ARQUITETURA DE UM **BLOCKCHAIN**



Processos que ocorrem em um nó

1. Verificação independente de cada transação, por todos os nós *full*, baseado em alguns critérios
2. Agregação independente de transações em um novo bloco, por nós mineradores, com a inclusão do PoW
3. Verificação independente de novos blocos por todos os nós e inclusão no *blockchain*
4. Seleção independente, por todos os nós, do *blockchain* mais longo e válido

1. Verificação independente de cada transação

Checklist de critérios:

Sintaxe e estrutura de dados corretos;

Lista de *inputs* e *outputs* não vazios;

Rejeita se soma das entradas for menor que a soma das saídas;

Os *unlocking scripts* de cada entrada deve validar os *locking scripts* das saídas correspondentes;

...

2. Agregação independente de transações em um novo bloco

Após validar uma transação, um nó a inclui no *mempool*

Nós mineradores começam a construir um bloco candidato e iniciam a busca da solução do PoW

Se outro bloco chegar, elimina as transações incluídas, remove do *mempool*, e começa a trabalhar em outro bloco candidato

Incluir a transação *coinbase* para o endereço do próprio minerador

Recompensa atual + Tx *fees*

2. Agregação independente de transações em um novo bloco


```
{
    "version" : 2,
    "merkleroot" : "c91c008c26e50763e9f548bb8b2fc323735f73577effbc55502c51eb4cc7cf2e",
    "tx" : [
        "d5ada064c6417ca25c4308bd158c34b77e1c0eca2a73cda16c737e7424afba2f",
        "b268b45c59b39d759614757718b9918caf0ba9d97c56f3b91956ff877c503fbe",

        ... 417 outras transações ...

       ],
    "time" : 1388185914,
    "nonce" : 924591752,
    "bits" : "1903a30c",
    "previousblockhash" : "0000000000000002a7bbd25a417c0374cc55261021e8a9ca74442b01284f0569"
}
```

2. Agregação independente de transações em um novo bloco


```
{
    "txid" : "d5ada064c6417ca25c4308bd158c34b77e1c0eca2a73cda16c737e7424afba2f",
    "version" : 1,
    "locktime" : 0,
    "vin" : [
        {
            "coinbase" : "03443b0403858402062f503253482f",
            "sequence" : 4294967295
        }
    ],
    "vout" : [
        {
            "value" : 25.09094928,
            "n" : 0,
            "scriptPubKey" : {
                "asm" : “02aa970c592640d19de03ff6f329d6fd2eecb023263b9ba5d1b81c29b523da8b21 OP_CHECKSIG",
                "hex" : "2102aa970c592640d19de03ff6f329d6fd2eecb023263b9ba5d1b81c29b523da8b21ac",
                "reqSigs" : 1,
                "type" : "pubkey",
                "addresses" : [
                    "1MxTkeEP2PmHSMze5tUZ1hAV3YTKu2Gh1N"
                ]
            }
        }
    ]
}
```

3. Verificação independente de novos blocos

Checklist de critérios:

A estrutura de dados do bloco é sintaticamente válida

O *hash* do cabeçalho do bloco é menor que o alvo (PoW)

O *timestamp* do bloco é maior que a média dos *timestamps* do últimos 11 blocos e menor que 2h no futuro

Tamanho do bloco é aceitável dentro dos limites

A primeira transação é a *coinbase*

Todas as transações dentro do bloco são válidas conforme critérios vistos em (1)

4. Seleção independente do *blockchain* mais longo e válido

Nós mantém três conjuntos de blocos:

os conectados ao *blockchain* principal

aqueles que formam *branches* do *blockchain* principal (*blockchains* secundárias)

blocos que não tem um pai conhecido pelo nó (orfão)

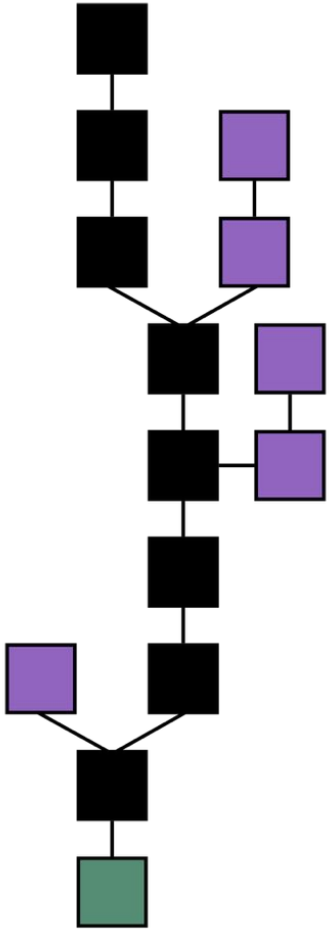
Forks do *blockchain*

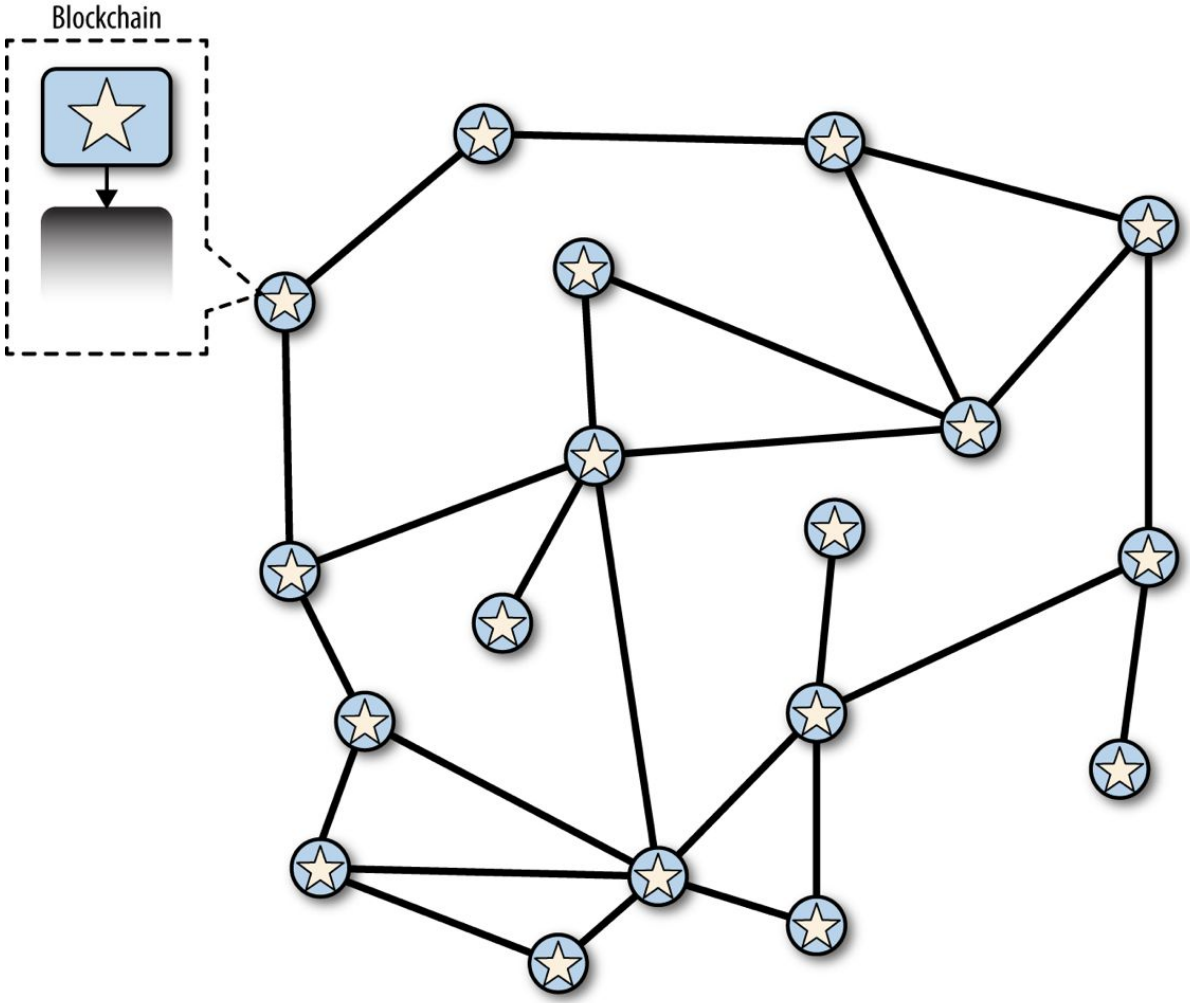
Como é uma estrutura de dados descentralizada, diferentes cópias do *blockchain* podem não ser consistentes

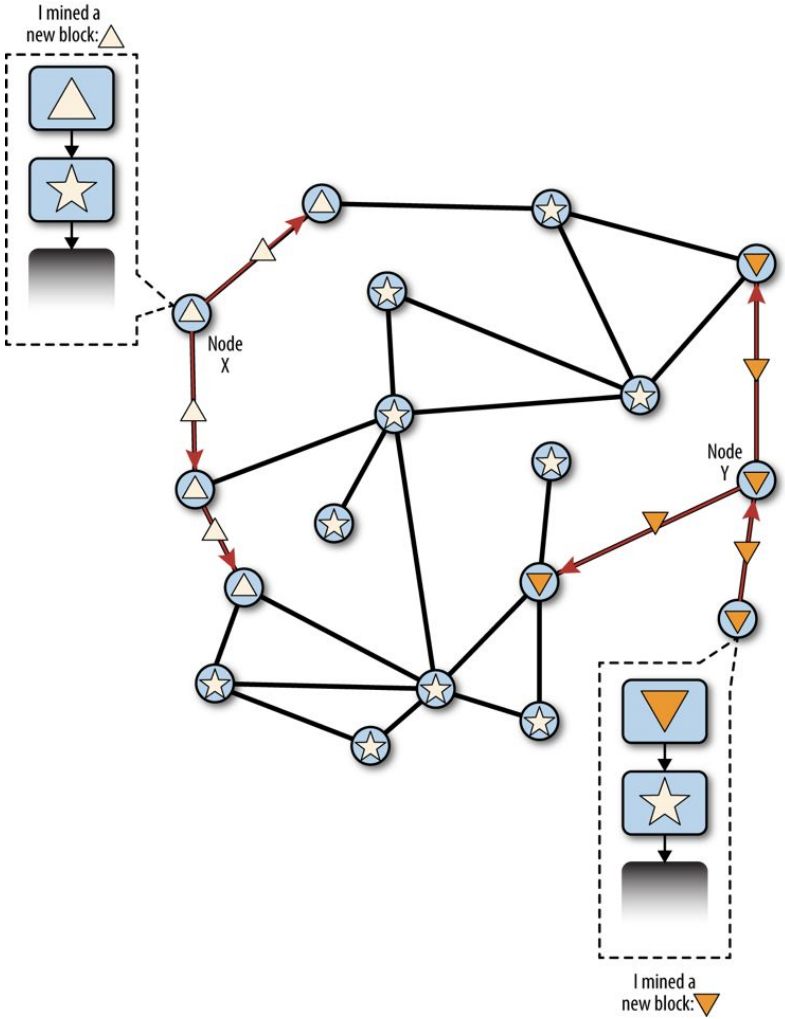
Forks ocorrem como inconsistências temporárias entre versões diferentes do *blockchain*, que serão resolvidas eventualmente através da reconvergência

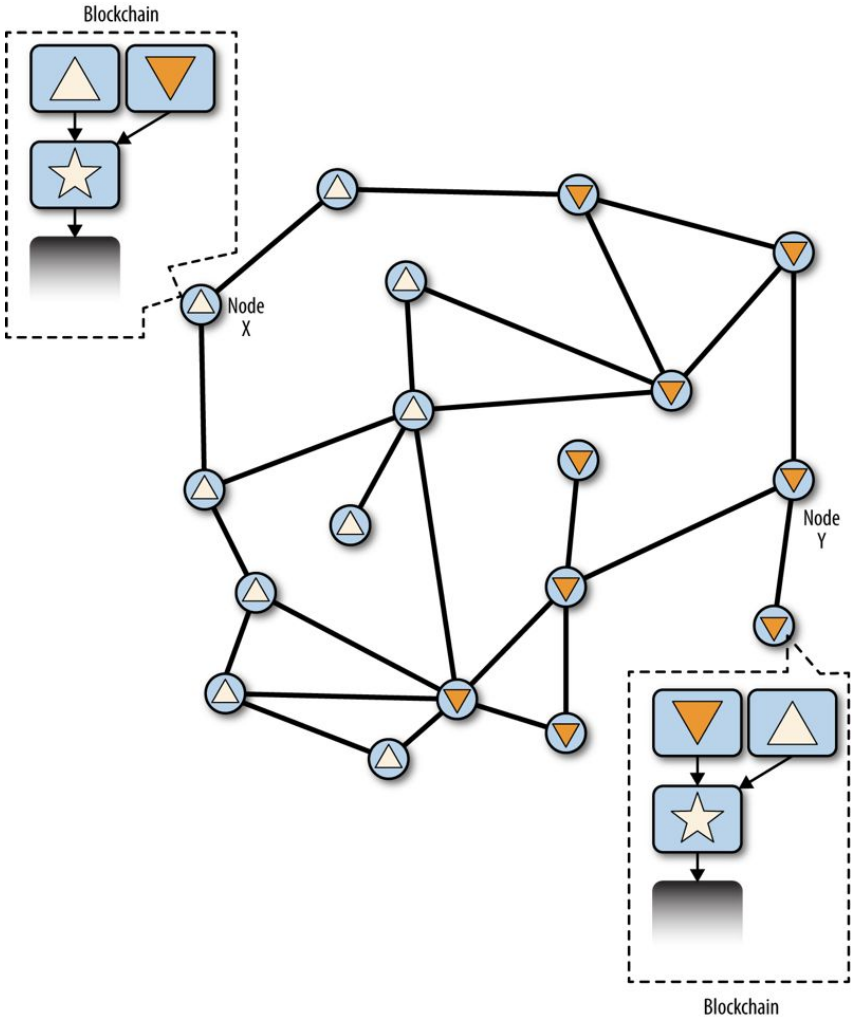
Isso é diferente dos *forks* induzidos! Veremos isso em breve!

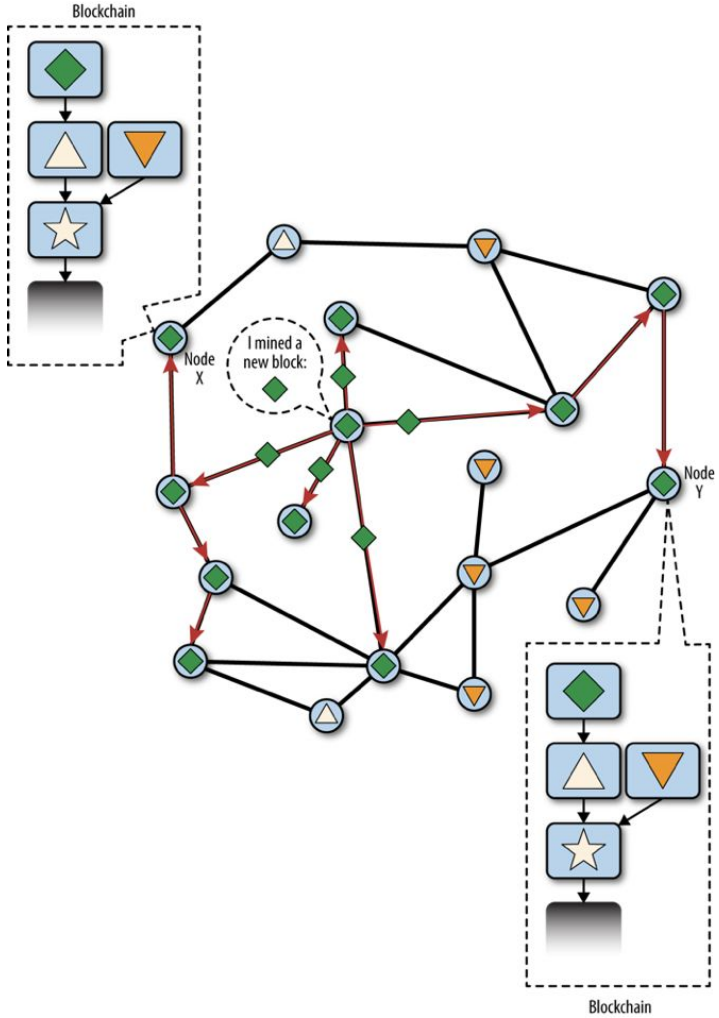
É corrigido após a reorganização da cadeia (***reorg***)

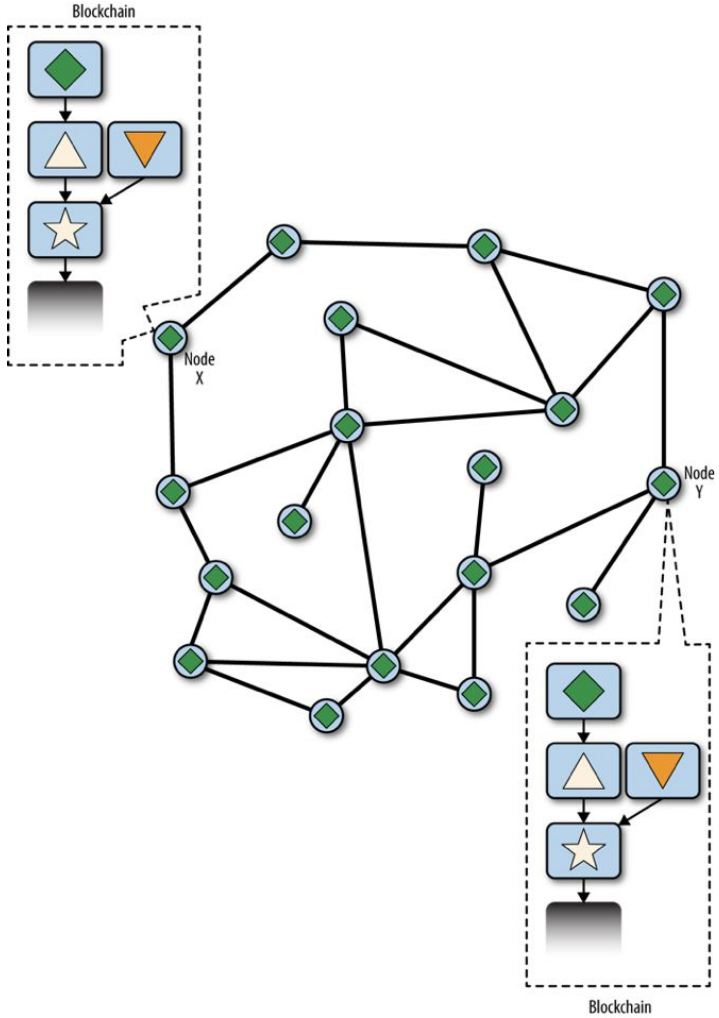



Blockchain

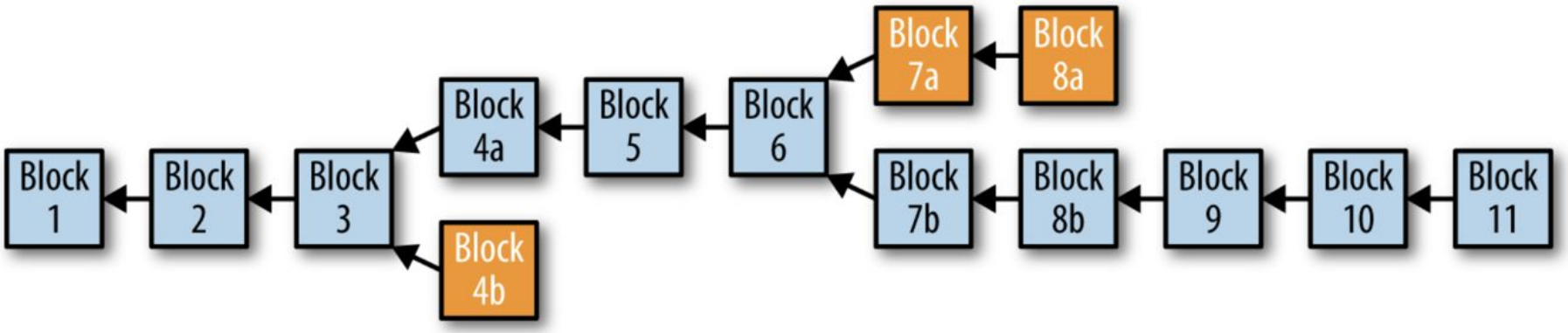

I mined a
new block:
Node
X
Node
Y
I mined a
new block:


Blockchain
Node
X
Node
Y
Blockchain

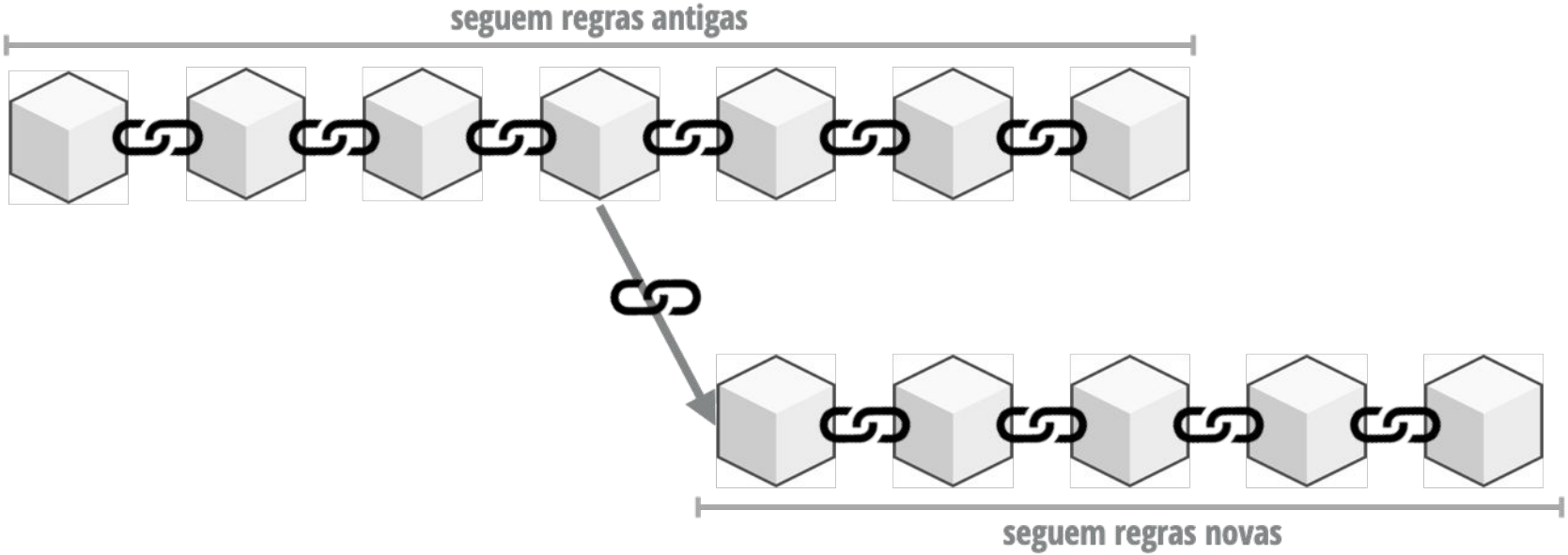

Blockchain
Node X
I mined a new block:
Node Y
Blockchain


Blockchain
Node
X
Node
Y
Blockchain

Forks do *blockchain*

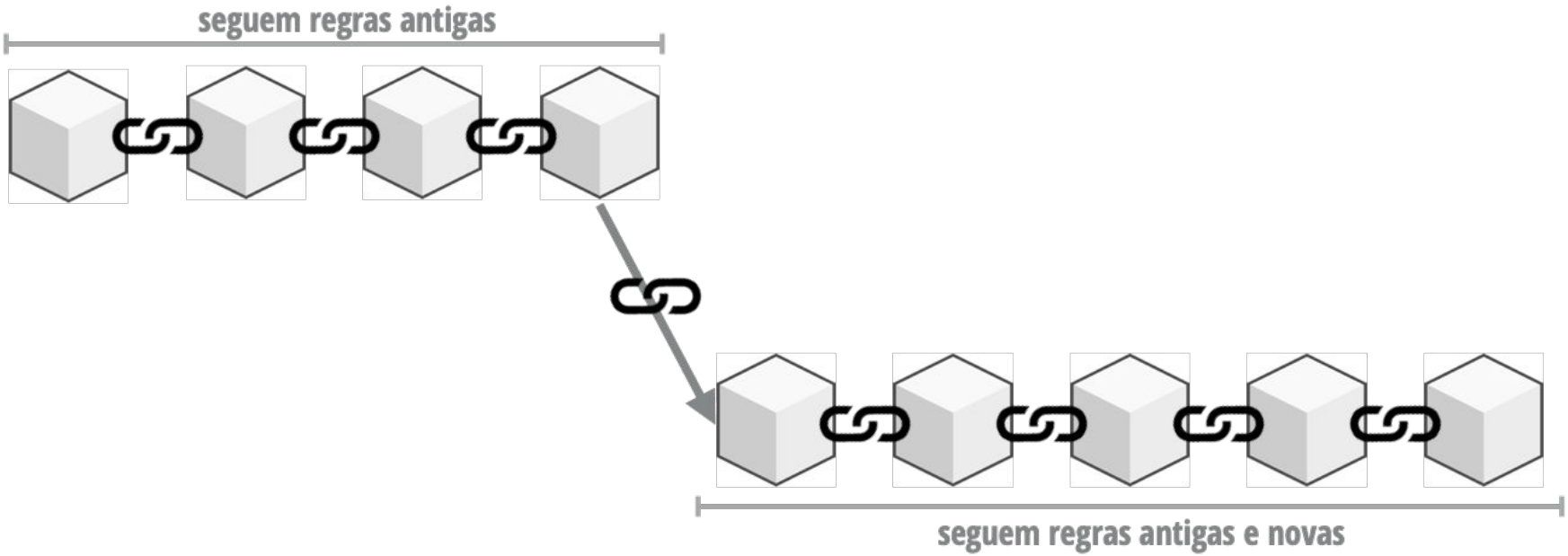


Fork induzido: *hard-fork*



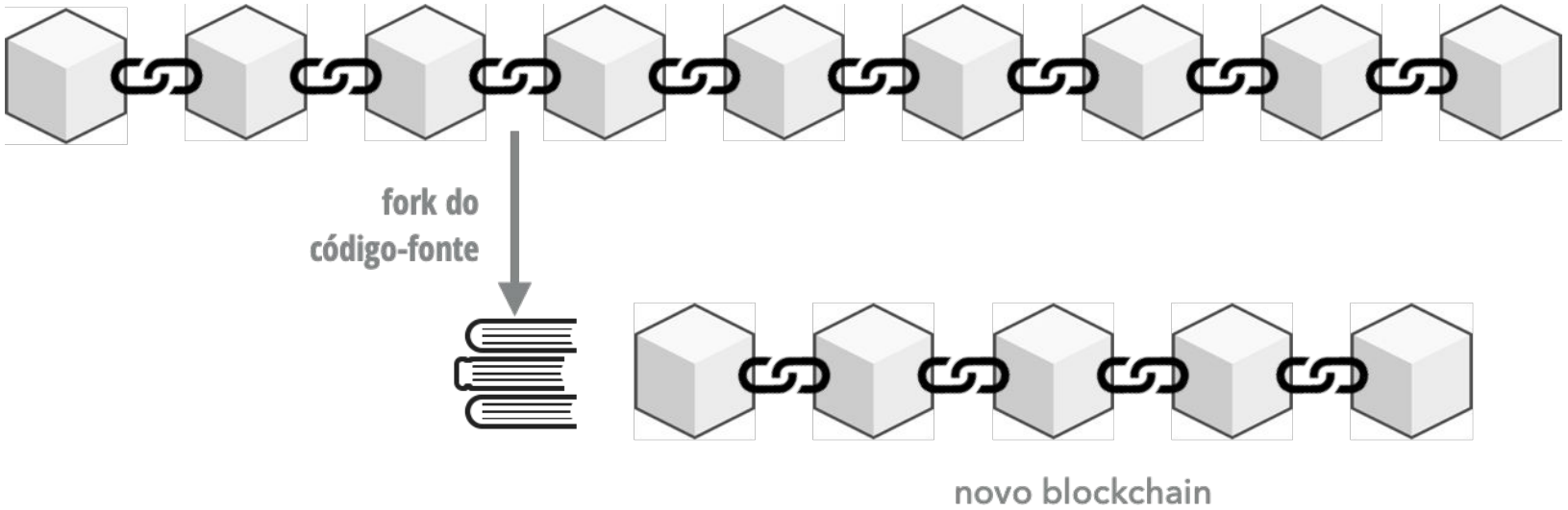
Hard fork: nós não-atualizados rejeitam transações e blocos com novas regras, gerando um *blockchain* divergente

Fork induzido: *soft-fork*



Soft fork: blocos violando novas regras se tornam obsoletos pela maioria de mineradores atualizados

Fork induzido: *source-code fork*



Fork do repositório Git para implementar um blockchain completamente novo

Exemplo: LiteCoin