
vinteum


42
| SÃO PAULO

Bitcoin Zero to Hero: Run Your own Bitcoin Node

Instituto 42

Edil Medeiros e Jão Noctus, 2024

42 | SÃO PAULO

vinteum

Edil Medeiros

Universidade de Brasília

prof@edil.com.br

@edil_medeiros

/@edilmedeiros

edil@nostrplebs.com

Jão Noctus

ZBD

jaonoctus@protonmail.com

PGP 0x782C165A293D6E18

jaonoctus.dev

₿SATS CONF 2024

BRAZIL'S LARGEST 100% BITCOIN EVENT

AUDIO | SÃO PAULO
NOVEMBER 8-9

Palestrante confirmado

Alex Gladstein

Human Rights Foundation

8 e 9 de Novembro | São Paulo

BSATS CONF 2024

Palestrante confirmado

niftynei (Lisa)

Base58⛓️🔓

8 e 9 de Novembro | São Paulo

₿SATS CONF 2024

Palestrante confirmado

Steven Roose

Ark

8 e 9 de Novembro | São Paulo

BSATS CONF 2024

Palestrante confirmado

Bruno Garcia

Bitcoin Core Dev

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Amiti Uttarwar

Bitcoin Core Dev

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Gloria Zhao

Bitcoin Core Dev

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Matt Odell

OpenSats / Ten31

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Obi Nwosu

Fedi

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Davidson Souza

Floresta/Utreexo

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

José Storopoli

Alpen Labs

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Níckolas Goline

NLightning

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Odudex

Krux

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Lorenzo Maturano

Bipa

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Plebhash

StratumV2

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Diego Kolling

NodeRunners Brasil

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Rudá Pellini

Arthur Inc.

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Felippe Hermes

BlockTrends

8 e 9 de Novembro | São Paulo

SATS CONF 2024

Palestrante confirmado

Raphaël Lima

Ideias Radicais

8 e 9 de Novembro | São Paulo

SATS CONF 2024

₿SATS CONF 2024

2 palcos simultâneos
100+ palestrantes
20+ Workshops
SatsKids
SatsMarket
SatsParty!

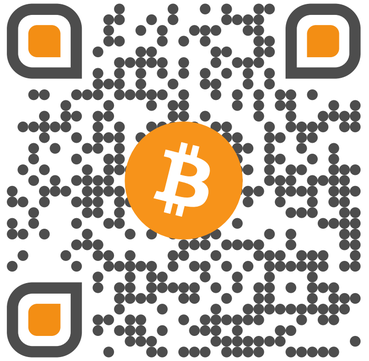
₿SATS CONF 2024

Cortesia

Bitcoin: A Peer-to-Peer Electronic Cash System

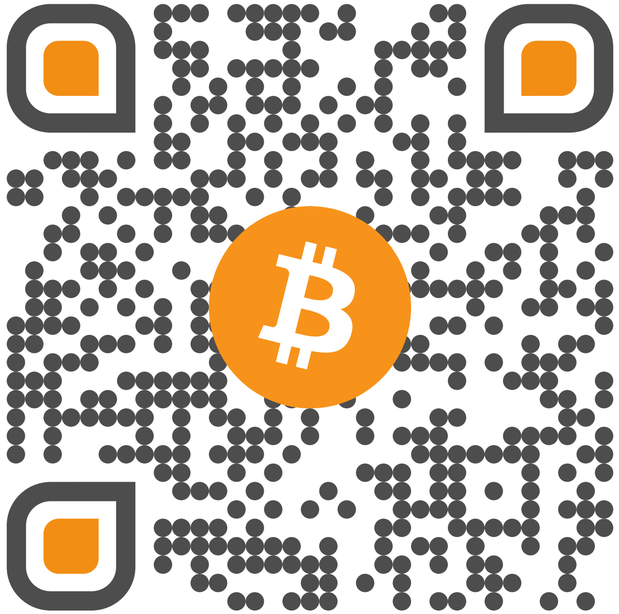
Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

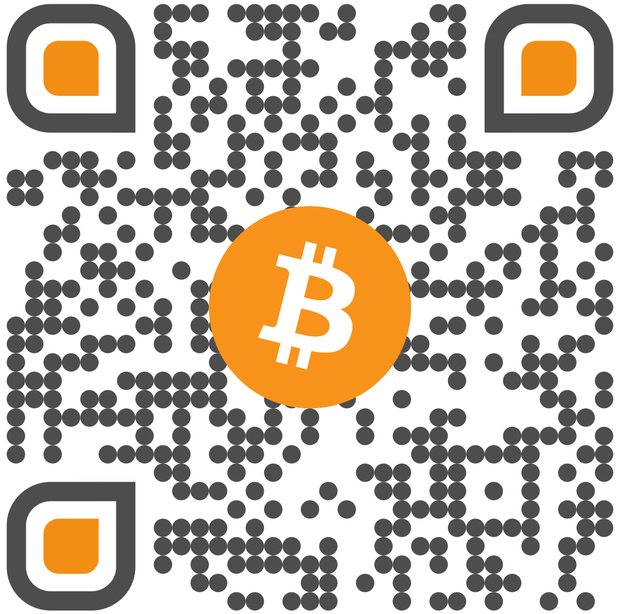


Agenda

- Step 1: Run a Bitcoin node
- Step 2: Create a wallet
- Step 3: Connect to other nodes
- Step 4: Create a payment
- Step 5: Confirm payments
- Step 6: Understanding transactions
- Step 7: Understanding blocks and the blockchain



https://edil.com.br/workshop42



Tenha um lugar para fazer anotações durante o workshop!

Step 1: Run a Bitcoin node

Explicar o que é um node

bitcoin.conf

```
daemon=1
txindex=1
signet=1
[signet]
signetchallenge=0014bdec02fe5ec499
cc2cb52dc160230643a84dd118
```

$ bitcoind

```
2024-10-28T15:02:55Z Signet with challenge 0014bdec02fe5ec499cc2cb52dc160230643a84dd118
2024-10-28T15:02:55Z Bitcoin Core version v27.1 (release build)
2024-10-28T15:02:55Z Signet derived magic (message start): 4930d6ea
2024-10-28T15:02:55Z Script verification uses 10 additional threads
2024-10-28T15:02:55Z Using the 'arm_shani(1way,2way)' SHA256 implementation
2024-10-28T15:02:55Z Default data directory /Users/jose.edil/Library/Application Support/
Bitcoin
2024-10-28T15:02:55Z Using data directory /Users/jose.edil/2-development/bitcoin/vinteum/
foss-program/infra-signet-server/datadir_test/signet
2024-10-28T15:02:55Z Config file: /Users/jose.edil/2-development/bitcoin/vinteum/foss-
program/infra-signet-server/datadir_test/bitcoin.conf
2024-10-28T15:02:55Z Config file arg: signet="1"
2024-10-28T15:02:55Z Config file arg: [signet] addnode="146.70.237.145:15745"
2024-10-28T15:02:55Z Config file arg: [signet] addnode="127.0.0.1:15745"
2024-10-28T15:02:55Z Config file arg: [signet] port=“15999”
…
```

$ bitcoin-cli **getblockchaininfo**


```
{
  "chain": "signet",
  "blocks": 0,
  "headers": 0,
  "bestblockhash": "00000008819873e925422c1ff0f99f7cc9bbb232af63a077a480a3633bee1ef6",
  "difficulty": 0.00112651529069818 6,
  "time": 1598918400,
  "mediantime": 1598918400,
  "verificationprogress": 1,
  "initialblockdownload": true,
  "chainwork": "0000000000000000000000000000000000000000000000000000049d414",
  "size_on_disk": 293,
  "pruned": false,
  "warnings": ""
}
```

```
$ bitcoin-cli help
$ bitcoin-cli help getblockchaininfo

getblockchaininfo

Returns an object containing various state info regarding blockchain processing.

Result:
{                                         (json object)
  "chain" : "str",                        (string) current network name (main, test, signet,
regtest)
  "blocks" : n,                           (numeric) the height of the most-work fully-validated
chain. The genesis block has height 0
…
}

Examples:
> bitcoin-cli getblockchaininfo
> curl --user myusername --data-binary '{"jsonrpc": "1.0", "id": "curltest", "method":
"getblockchaininfo", "params": []}' -H 'content-type: text/plain;' http://127.0.0.1:8332/
```

Step 2: Create a wallet

Explicar o que é uma carteira

Hardware Wallets na Prática

Construindo uma Wallet com BDK

Auto-custódia Básica

```
$ bitcoin-cli createwallet "<name>"


{
  "name": "wallet_edil"
}
```

```
$ bitcoin-cli getbalance

0.00000000
```

```
$ bitcoin-cli importdescriptors ‘[{“desc”: “<descriptor>”, “timestamp”: 0}]’


[
  {
    "success": true,
    "warnings": [
      "Range not given, using default keypool range"
    ]
  }
]
```

Qual é o saldo da sua carteira?

Step 3: Connect to other nodes

```
$ bitcoin-cli getpeerinfo


[
]
```

Em duplas, decidam quem será o nó A e quem será o nó B.

A. Conecte-se ao seu colega

```
$ bitcoin-cli addnode “<node IP>:<node port>" “onetry"
```

A+B. Verifique seus peers

Qual é o saldo da sua carteira?

Que tal conectar o seu node ao de outros 3 colegas?

Quantos blocos tem a sua blockchain?

Conecte-se ao mineradores!

Do Bloco Gênesis às Hashwars: Um Guia para Iniciantes sobre Mineração Soberana

E agora, quantos blocos tem a sua blockchain?

Anote o saldo da sua carteira!

Step 4: Create a payment

Em duplas, decidam quem será o nó A e quem será o nó B.

A. Crie um endereço para receber bitcoins do seu colega.

```
$ bitcoin-cli getnewaddress

tb1q3hn57lswjdmv5vnr4nff5uhwxuee0erugvy8g6
```

B. Crie um pagamento de para o seu colega. Anote o txid para investigarmos a transação posteriormente.

```
$ bitcoin-cli -named sendtoaddress address="<address>" amount=<amount>
fee_rate=<fee_rate>
```

```
4c42e0d3bf7ca1ac65107e12207ae6ed0d38428a4dbc1c7665aa767fb2dd953a
```

Qual é o balanço da sua carteira após o pagamento?

```
$ bitcoin-cli getbalances
```


```
{
  "mine": {
    "trusted": 7.46290747,
    "untrusted_pending": 1.00000000,
    "immature": 0.00000000
  },
  "lastprocessedblock": {
    "hash":
"0000011b6b8e072729f5e4500946c96530f2a296305b687ba18a20d3db7fef1e",
    "height": 303
  }
}
```

O que é o saldo pendente?

$ bitcoin-cli **getrawtransaction** “<txid>” 1


```
{
  "txid": "4c42e0d3bf7ca1ac65107e12207ae6ed0d38428a4dbc1c7665aa767fb2dd953a",
  "hash": "6fe60b8bceda707bbe681b35e029a04494ab2b63681e79a2612982df68089fb9",
  "version": 2,
  "size": 518,
  "vsize": 276,
  "weight": 1103,
  "locktime": 216,
  "vin": [
    {
      "txid": "35bbf26d0e2eeef9d30e501ff5a7e0980643ac27c0c8849d043b3783be6f6ea5",
      "vout": 0,
      "scriptSig": {
        "asm": "",
        "hex": ""
      },
      "txinwitness": [
        "304402200ca30fcf9b0897b4938de8a1e149a0856ee56b7ef45e7d1a9b02701aa5064e20022027f0721ee2214af24cf8f693aae605e0c4728dbd3c6cd0e3551a081d489f142301",
        "036128c701f1769889945de8ddd59e8e0c6d33553b56d6a7ef9c46aed39c49d793"
      ],
      "sequence": 4294967293
    },
    {
      "txid": "12604a90d38827ebfc3381307751d77a587a12e5dae5a6cb26482b1fce592900",
      "vout": 1,
      "scriptSig": {
        "asm": "",
        "hex": ""
      },
```

Onde foi parar a transação?

```
$ bitcoin-cli getrawmempool


[
  "4c42e0d3bf7ca1ac65107e12207ae6ed0d38428a4dbc1c7665aa767fb2dd953a"
]
```

Step 5: Confirm payments

Qual é o saldo da sua carteira após a mineração de um novo bloco?

Diving deeper!

Step 6: Understanding blocks and the blockchain

Qual foi o último bloco minerado?

```
$ bitcoin-cli getblockhash <height>
```

```
000002898e5d67b653370beb37701af9fa207a1ce3ed5a5099368219b908a4bf
```

$ bitcoin-cli **getblockheader** "<block hash>”


```
{
  "hash": "000002898e5d67b653370beb37701af9fa207a1ce3ed5a5099368219b908a4bf",
  "confirmations": 6,
  "height": 304,
  "version": 536870912,
  "versionHex": "20000000",
  "merkleroot": "4922268ff84988cc31a88bb42809072dc15cc2dc19906c841eeb195b5a621733",
  "time": 1730203899,
  "mediantime": 1730200899,
  "nonce": 1641808,
  "bits": "1e0377ae",
  "difficulty": 0.0011265152906981 86,
  "chainwork": "00000000000000000000000000000000000000000000000000000057f5abd4",
  "nTx": 2,
  "previousblockhash": "0000011b6b8e072729f5e4500946c96530f2a296305b687ba18a20d3db7fef1e",
  "nextblockhash": "000002ae2176c9ed5728f7816b26c9887a1f83d2cd3c820d70016e89851e3af2"
}
```

```
$ bitcoin-cli getblockheader <block hash> false
```

```
000000201ca9d4c086746e68cfbb7a27d44734ff9cb35412f1ad94332a50ff2691010
000a5497a5c64266cee635fb1789d6e617ccb41addd538bec854971f843a6f24b8cdb
d31f67ae77031e2a963200
```

Qual versão do cabeçalho do bloco é efetivamente transmitido na rede?

Qual informação está presente no cabeçalho do bloco que faltava nas transações?

Qual é a função da blockchain?

Step 7: Understanding transactions

$ bitcoin-cli **getrawtransaction** “<txid>” 1


```
{
  "txid": "4c42e0d3bf7ca1ac65107e12207ae6ed0d38428a4dbc1c7665aa767fb2dd953a",
  "hash": "6fe60b8bceda707bbe681b35e029a04494ab2b63681e79a2612982df68089fb9",
  "version": 2,
  "size": 518,
  "vsize": 276,
  "weight": 1103,
  "locktime": 216,
  "vin": [
    {
      "txid": "35bbf26d0e2eeef9d30e501ff5a7e0980643ac27c0c8849d043b3783be6f6ea5",
      "vout": 0,
      "scriptSig": {
        "asm": "",
        "hex": ""
      },
      "txinwitness": [
        "304402200ca30fcf9b0897b4938de8a1e149a0856ee56b7ef45e7d1a9b02701aa5064e20022027f0721ee2214af24cf8f693aae605e0c4728dbd3c6cd0e3551a081d489f142301",
        "036128c701f1769889945de8ddd59e8e0c6d33553b56d6a7ef9c46aed39c49d793"
      ],
      "sequence": 4294967293
    },
    {
      "txid": "12604a90d38827ebfc3381307751d77a587a12e5dae5a6cb26482b1fce592900",
      "vout": 1,
      "scriptSig": {
        "asm": "",
        "hex": ""
      },
```

Quais são os componentes da transação?

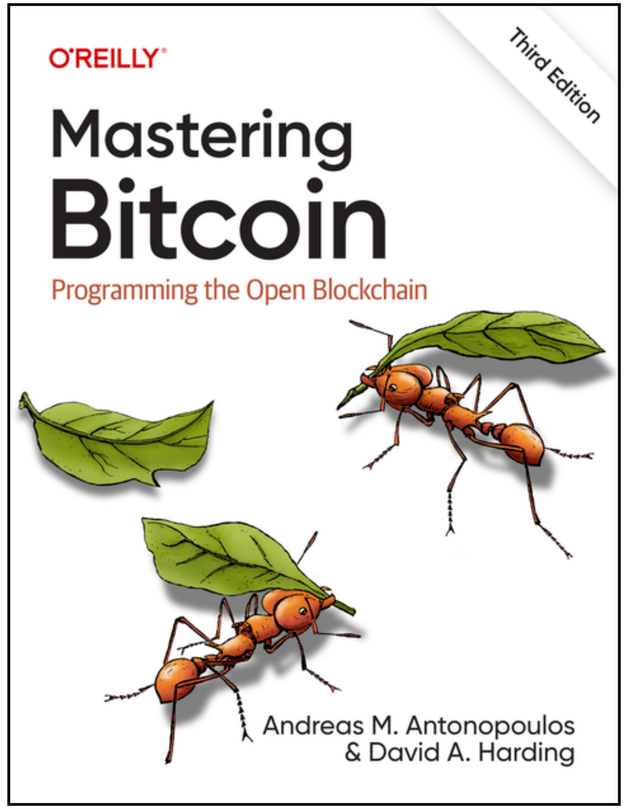
Quais as informações presentes em uma saída de uma transação?

Qual é a função do campo scriptPubKey?

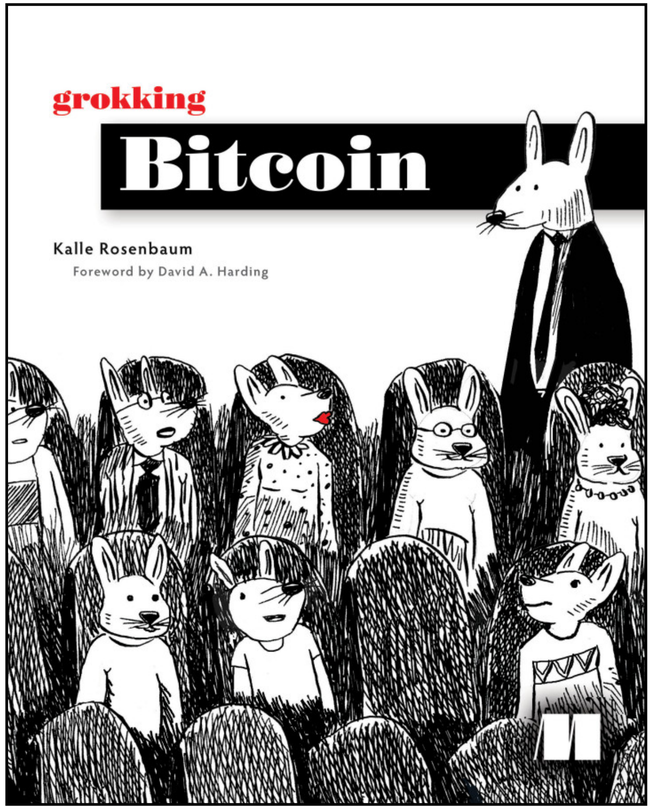
O que é um UTXO?

Quais as informações presentes em uma entrada de uma transação?

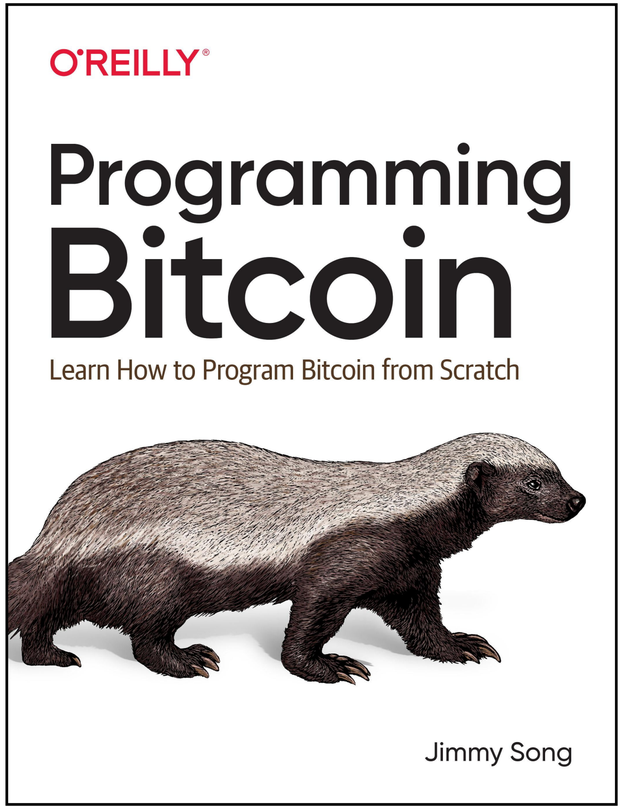
Qual informação não está presente nas transações?



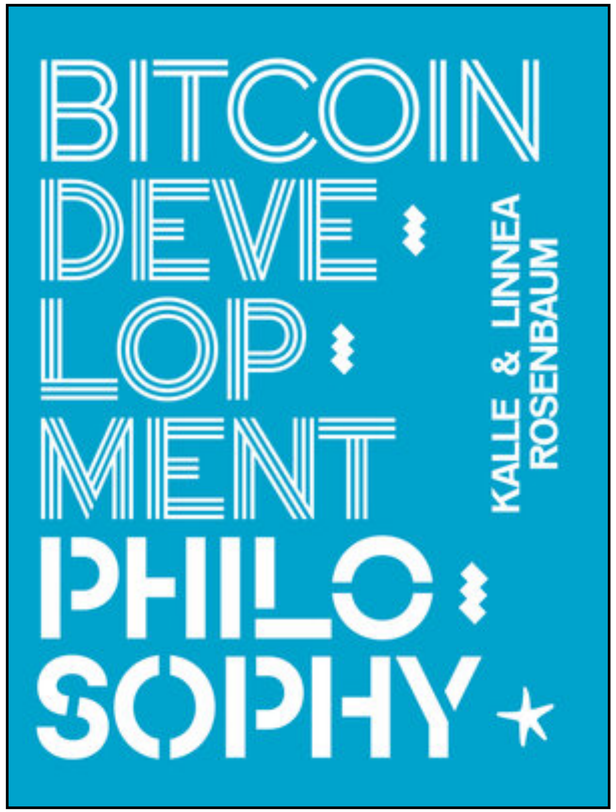
Most comprehensive



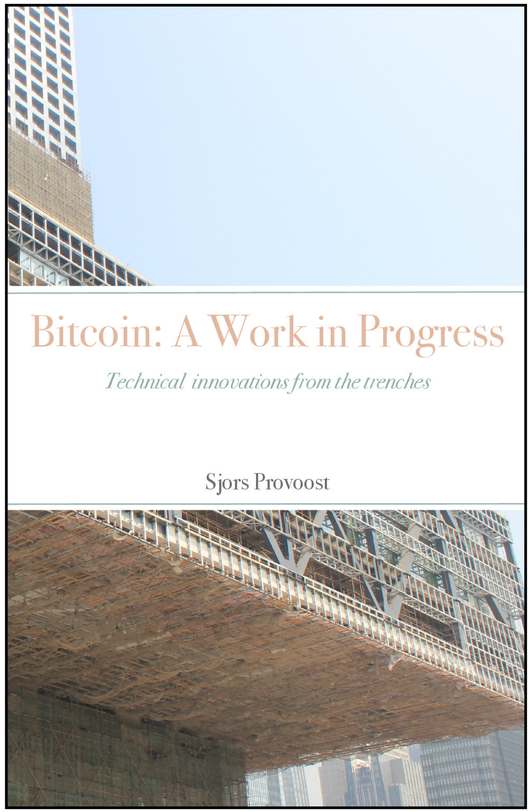
Best pedagogy



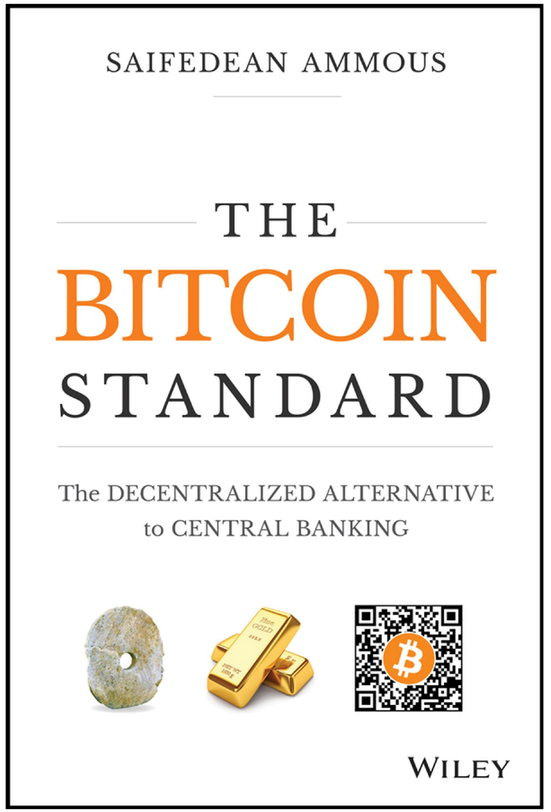
Most hands-on



Bitcoin development ethos



Bitcoin history



Bitcoin economics

Edil Medeiros

Universidade de Brasília

prof@edil.com.br

@edil_medeiros

/@edilmedeiros

edil@nostrplebs.com

Jão Noctus

ZBD

jaonoctus@protonmail.com

PGP 0x782C165A293D6E18

jaonoctus.dev


vinteum


42
| SÃO PAULO