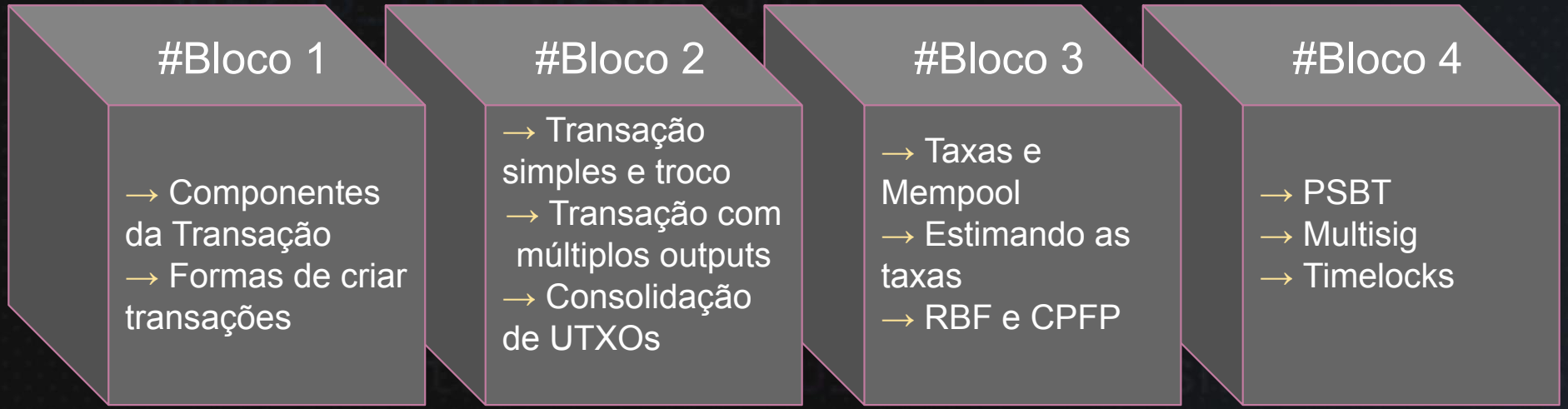
Transações
no Bitcoin Core e Signet

Anfitrião:
Rafael Penna

bitcoinCoders bitups

O Que Veremos!



bitcoinCoders bitups

O Que Veremos!

#Bloco 1

→ Componentes da Transação
→ Formas de criar transações

#Bloco 2

→ Transação simples e troco
→ Transação com múltiplos outputs
→ Consolidação de UTXOs

#Bloco 3

→ Taxas e Mempool
→ Estimando as taxas
→ RBF e CPFP

#Bloco 4

→ PSBT
→ Multisig
→ Timelocks

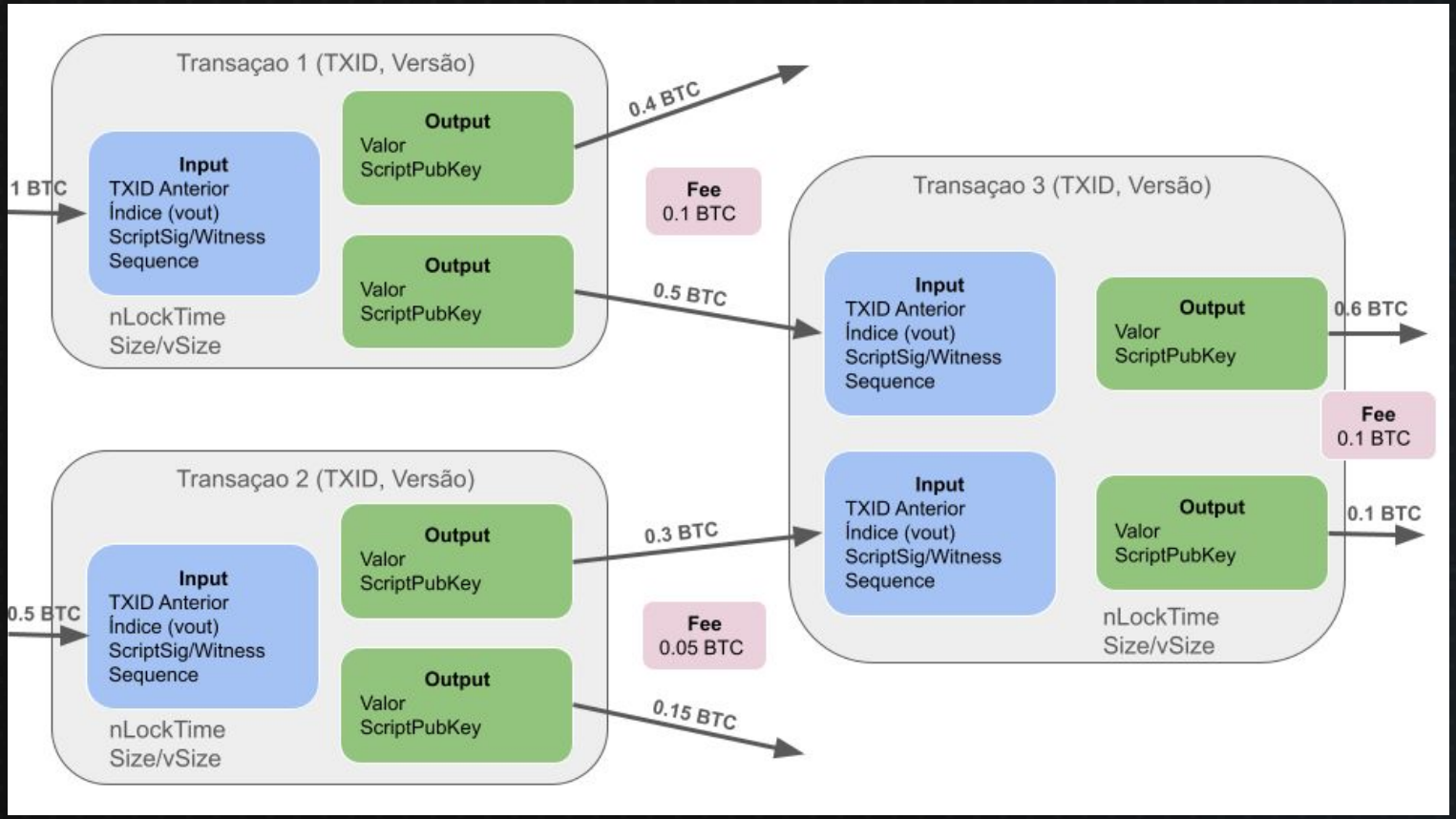
Fundamentos

🔑 Componentes de uma Transação Bitcoin

1. **Identificação**
   - **TXID**
   - **Versão**
2. **Inputs (entradas)**
   - **TXID anterior**
   - **Índice (vout)**
   - **ScriptSig / Witness**: dados de desbloqueio
   - **Sequence**: controle de tempo.
3. **Outputs (saídas)**
   - **Valor (satoshis)**
   - **ScriptPubKey**: regras de gasto do output.
4. **Metadados**
   - **nLockTime**
   - **Size / vSize**
5. **Taxa (Fee)**
   - Soma(inputs) – Soma(outputs)

bitcoinCoders bitups

Fundamentos



Fundamentos

🔑 Formas de criar transações no Bitcoin Core

1. **Forma simples**

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendtoaddress
"tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny" 0.00001
```

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendmany ""
"{\"tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny\":0.00002,\"tb1qecg3yd2gf7u3h8nfjvf45g48d
qts44xav0y0hl\":0.00003}"
```

Fundamentos

🔑 Formas de criar transações no Bitcoin Core

2. **Forma bruta**

→ Criar Transação

```
bitcoin-cli -datadir=. -rpcwallet=Carteira-Cofre createrawtransaction
"[{\"txid\":\"bdfa04982337f364e2f349d02de7d3c2a490f7567bdf0c1ba89a056922d76cad\",\"vout\
":2}]" "{\"tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny\":0.01586400}"
```

→ Assinar Transação

```
bitcoin-cli -datadir=. -rpcwallet=Carteira-Cofre signrawtransactionwithwallet
"0200000001ad6cd72269059aa81b0cdf7b56f790a4c2d3e72dd049f3e264f337239804fabd0200000000fdf
fffff01e0341800000000000160014d0434e673aab237ab07a3468103eb2ba48f2e6ad00000000"
```

Fundamentos

🔑 Formas de criar transações no Bitcoin Core

2. **Forma bruta**

```
→ Transmitir Transação
bitcoin-cli -datadir=. -rpcwallet=Carteira-Cofre sendrawtransaction
"020000000000101ad6cd72269059aa81b0cdf7b56f790a4c2d3e72dd049f3e264f337239804fabd020000000
0fdffffff01e034180000000000160014d0434e673aab237ab07a3468103eb2ba48f2e6ad0247304402205cd
3e73aec04af194683f16f02ea280490dc0ac7f7bfd6c7b3072bb3eaa5793102200ba0398b94c6f6eea149be4
dda728c7bdab3f7878cb1c94156c9eb17fa9210ba012103965d7a434dcb7eca746f247c980a61f29cad3e92c
9ca9efc98644fad186da32f00000000"
```

Fundamentos

🔑 Formas de criar transações no Bitcoin Core

3. Forma semi-automática

→ Criar Transação

```
bitcoin-cli -datadir=. -rpcwallet=Carteira-Cofre createrawtransaction "[]"
"{\"tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny\":0.000042}"
```

→ Completar Transação

```
bitcoin-cli -datadir=. -rpcwallet=Carteira-Cofre fundrawtransaction
"020000000001681000000000000160014d0434e673aab237ab07a3468103eb2ba48f2e6ad00000000"
```

Fundamentos

🔑 Formas de criar transações no Bitcoin Core

3. Forma semi-automática

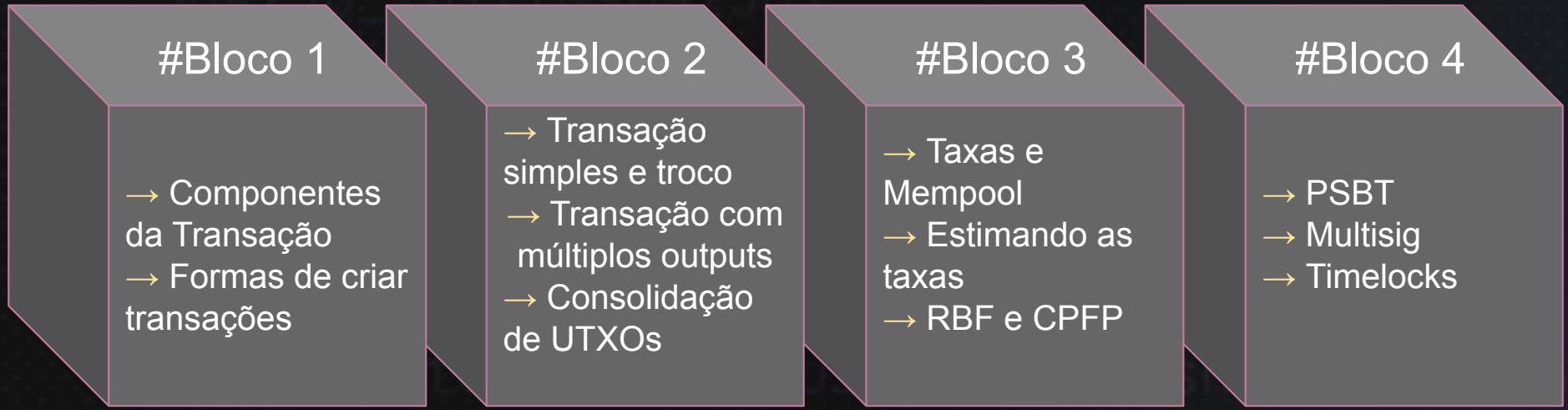
→ Assinar Transação

```
bitcoin-cli -datadir=. -rpcwallet=Carteira-Cofre signrawtransactionwithwallet
"0200000002ad6cd72269059aa81b0cdf7b56f790a4c2d3e72dd049f3e264f337239804fabd0100000000fdf
...34e673aab237ab07a3468103eb2ba48f2e6ad00000000"
```

→ Transmitir Transação

```
bitcoin-cli -datadir=. -rpcwallet=Carteira-Cofre sendrawtransaction
"020000000000102ad6cd72269059aa81b0cdf7b56f790a4c2d3e72dd049f3e264f337239804fabd010000000
0fdfffffffad6cd72269059aa81b0c...f65582f7d9e278dd3a8c71289b9493200000000"
```

O Que Veremos!



Construções de Transações

🔑 Transação simples e o papel do troco

1️⃣ Enviar parte do valor, deixando o Core criar troco

a. Criar o hex bruto sem taxa nem troco

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre createrawtransaction
"[{\"txid\":\"c89fca03d30045154e127edf178464b07af2106a1975bbef482a44f75c88f1cd\",\"vout\
":0}]" "{\"tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny\":0.01}"
```

b. Completar com taxa e troco automático

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre fundrawtransaction
"0200000001cdf1885cf7442a48efbb75196a10f27ab0648417df7e124e154500d303ca9fc80000000000fdf
fffff0140420f0000000000160014d0434e673aab237ab07a3468103eb2ba48f2e6ad00000000"
```

Construções de Transações

🔑 Transação simples e o papel do troco

1️⃣ Enviar parte do valor, deixando o Core criar troco

c. Assinar e enviar

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre signrawtransactionwithwallet
"0200000001cdf1885cf7442a48efbb75196a10f27ab0648417df7e124e154500d303ca9fc80000000000fdf
fffff0213f208000000000160014bb55e56cf68903252533a7df3d614c291e3b0ecc40420f0000000000160
014d0434e673aab237ab07a3468103eb2ba48f2e6ad00000000"

bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendrawtransaction
"02000000000101cdf1885cf7442a48efbb75196a10f27ab0648417df7e124e154500d303ca9fc8000000000
...36734fc6ba4eb0c01210326e8e2ebd29d1e16a329df2dce8f276f2f65582f7d9e278dd3a8c71289b949320
0000000"
```

Construções de Transações

🔑 Transação simples e o papel do troco

2️⃣ Gastar todo o valor, subtraindo a taxa automaticamente

a. Enviar todo o saldo, subtraindo a taxa

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendtoaddress
tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny 0.01891639 "" "" true
```

subtractfeefromamount

b. Checar UTXO resultante

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre listunspent
```

bitcoinCoders bitups

Construções de Transações

🔑 Transação simples e o papel do troco

3️⃣ Gastar todo o valor ajustando a taxa manualmente

a. Calcular o valor de envio

```
VALOR -> 0.01891394-0.00000300
# 0.01891094
```

b. Criar a transação bruta com valor já descontado

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre createrawtransaction
"[{\"txid\":\"b7fdf34591095619b55f1da1b70686bf01e50db180d43e54c7272e3ebafbc06d\",\"vout\
":0}]" "{\"tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny\":0.01891094}"
```

Construções de Transações

🔑 Transação simples e o papel do troco

3️⃣ Gastar todo o valor ajustando a taxa manualmente

c. Assinar

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre signrawtransactionwithwallet
"02000000016dc0fbba3e2e27c7543ed480b10de501bf8606b7a11d5fb51956099145f3fdb70000000000fdf
fffff0116db1c0000000000160014d0434e673aab237ab07a3468103eb2ba48f2e6ad00000000"
```

d. Enviar

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendrawtransaction
"020000000001016dc0fbba3e2e27c7543ed480b10de501bf8606b7a11d5fb51956099145f3fdb7000000000
...d9e278dd3a8c71289b9493200000000"
```

e. Conferir o novo UTXO

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre listunspent
```

Construções de Transações

🔑 Transação com múltiplos outputs (batching)

a. Criar os endereços de destino

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre getnewaddress
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre getnewaddress
```

b. Criar a transação bruta com múltiplos outputs

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre createrawtransaction
"[{\"txid\":\"eba83baf0c18d138f9ae3628c3e763e4516cc0453a81ed70b00888bc3b3ae3fb\",\"vout\
":0}]"
"{\"tb1qacse5r77rpxupretvy24gu3dvs9an7y0q0ewa9\":0.0007,\"tb1q2dtnxalqx3n4n2yhrgvv430dfr
dmjdun2fd0c0\":0.0002}"
```

Construções de Transações

🔑 Transação com múltiplos outputs (batching)

c. Completar com taxa e troco

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre fundrawtransaction
"0200000001fbe33a3bbc8808b070ed813a45c06c51e463e7c32836aef938d1180caf3ba8eb0000000000fdf
...001453573377e0346759a8971a18cac5ed48dbb9379300000000"
```

d. Assinar e enviar

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre signrawtransactionwithwallet
"0200000001fbe33a3bbc8808b070ed813a45c06c51e463e7c32836aef938d1180caf3ba8eb0000000000fdf
...8cac5ed48dbb9379300000000"
```

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendrawtransaction
"02000000000101fbe33a3bbc8808b070ed813a45c06c51e463e7c32836aef938d1180caf3ba8eb000000000
...2ebd29d1e16a329df2dce8f276f2f65582f7d9e278dd3a8c71289b94932000000000"
```

Construções de Transações

🔑 Consolidação de UTXOs

Consolidar é juntar vários UTXOs pequenos em um único UTXO maior.

Vantagens: transações futuras com menos inputs (mais baratas) e carteira mais simples.

Cuidado: liga aqueles UTXOs entre si. Evite KYC/non-KYC e prefira fazer quando as taxas estão baixas.

Opções:

1 — **Com fundrawtransaction (sem troco via subtractFeeFromOutputs)**

2 — **Total controle (sem fundrawtransaction)**

Construções de Transações

🔑 Consolidação de UTXOs

2 — Total controle (sem fundrawtransaction)

a) Identificar UTXOs

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre listunspent
```

b) Gerar um endereço para receber a consolidação

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre getnewaddress
```

c) Calcular o valor de saída

```
VALOR=0.01890671-(0.00000300)
```

```
VALOR=0.01890371
```

Construções de Transações

🔑 Consolidação de UTXOs

2 — Total controle (sem fundrawtransaction)

d) Criar o raw sem espaço para troco

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre createrawtransaction
"[{\"txid\":\"6ef2d099033017e585bce90d8f8ff7247caabc7a4ae4af7f77052f4612dca675\",\"vout\
":0},{\"txid\":\"6ef2d099033017e585bce90d8f8ff7247caabc7a4ae4af7f77052f4612dca675\",\"vo
ut\":1},{\"txid\":\"6ef2d099033017e585bce90d8f8ff7247caabc7a4ae4af7f77052f4612dca675\",\
"vout\":2}]" "{\"tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny\":0.01890371}"
```

e) Assinar

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre signrawtransactionwithwallet
"020000000375a6dc12462f05777fafe44a7abcaa7c24f78f8f0de9bc85e517300399d0f26e0000000000fdf
...c00000000000160014d0434e673aab237ab07a3468103eb2ba48f2e6ad00000000"
```

Construções de Transações

🔑 Consolidação de UTXOs

2 — Total controle (sem fundrawtransaction)

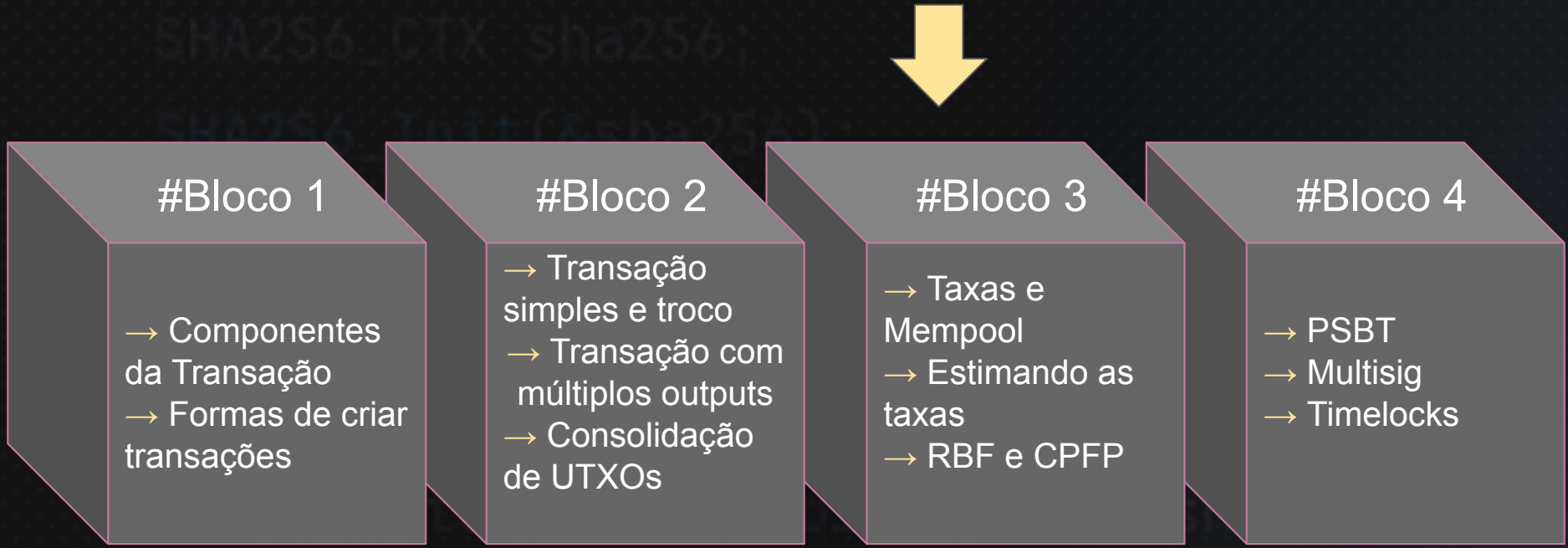
f) Enviar

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendrawtransaction
"0200000000010375a6dc12462f05777fafe44a7abcaa7c24f78f8f0de9bc85e517300399d0f26e000000000
...ecaf3e74012103f6d6ec7df5e08cb38cc65248071f38211f01fdddcc193f1270d475824e042a9e00000000
"
```

g) Listar os UTXOs para conferir

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre listunspent
```

O Que Veremos!



bitcoinCoders bitups

Taxas e Mempool

🔑 Estimando as taxas

1. Estimar a taxa necessária

```
bitcoin-cli -datadir="." estimatesmartfee 6
```

2. Prever o tamanho da transação

→ Transação com 1 input e 1 output no formato SegWit costuma ter ~141 vbytes.

→ `decoderawtransaction`

3. Calcular a fee total

fee_total = feerate × tamanho

Taxas e Mempool

🔑 Replace-by-Fee (RBF) e Child-Pays-for-Parent (CPFP)

Quando uma transação fica presa na mempool por ter uma taxa muito baixa, o Bitcoin Core oferece dois mecanismos diferentes para “destravar” a confirmação: Replace-by-Fee (RBF) e Child-Pays-for-Parent (CPFP).

Taxas e Mempool

🔑 Replace-by-Fee (RBF)

1) Criar a transação com RBF habilitado

```
bitcoin-cli -datadir="." -named sendtoaddress
address="tb1qmx7xn97urd2lmmthju57l4f459f5dngwv2gm5l" amount=0.001 replaceable=true
```

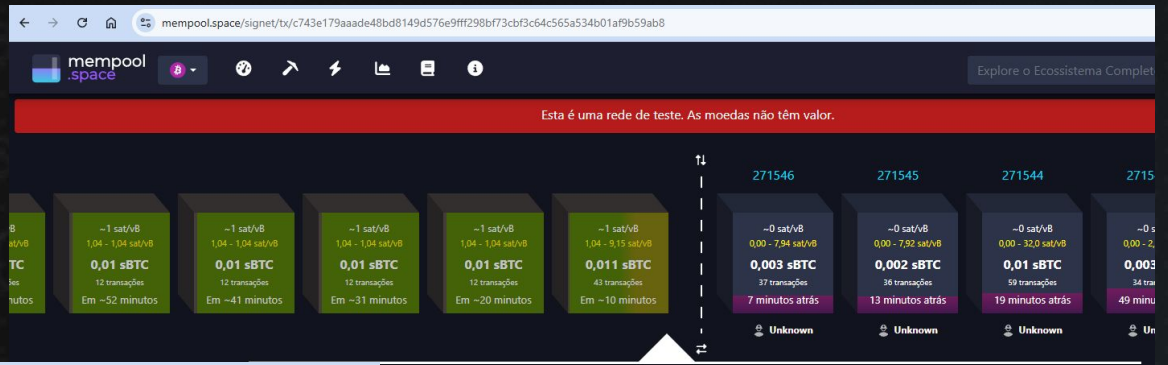
2) Aumentar a taxa

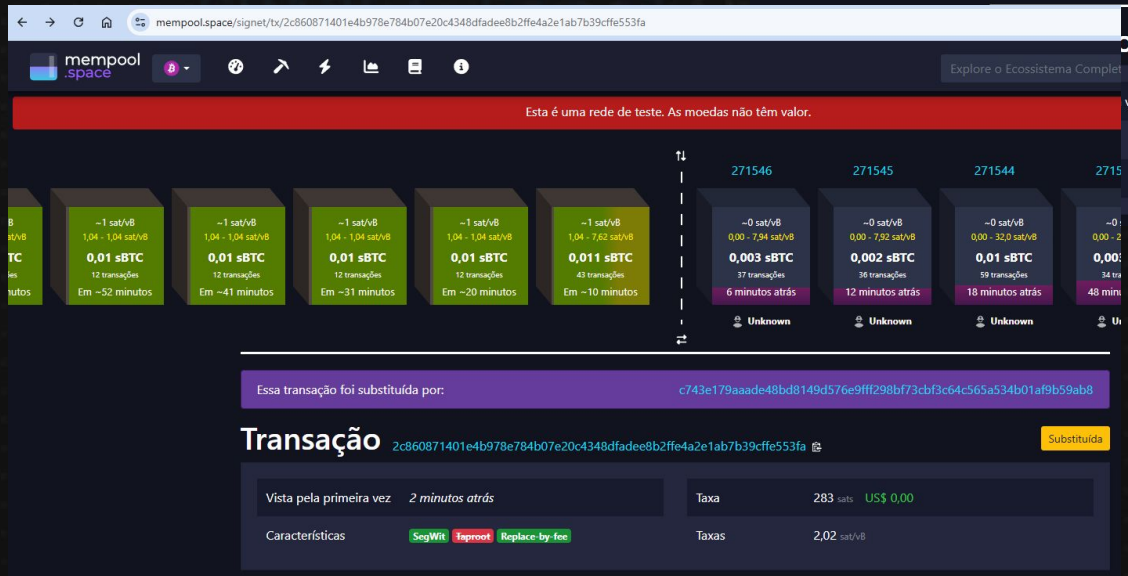
```
bitcoin-cli -datadir="." bumpfee
860da2e3a9d4b1278432e053023151303f07661349c0a9732a396c4f0829c20e
```

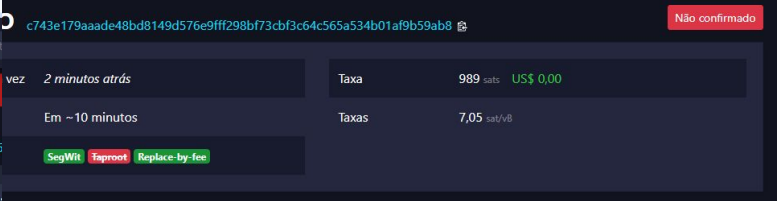
Taxas e Mempool

🔑 Replace-by-Fee (RBF)

1) Conferir na mempool







bitcoinCoders bitups

Taxas e Mempool

🔑 Child-Pays-for-Parent (CPFP)

1. **Criar uma transação para ter um UTXO não confirmado**

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendtoaddress
"tb1q2dtnxalqx3n4n2yhrgvv430dfrdmjdun2fd0c0" 0.0002
# 45d9c6faeb2d4b43a7f7c7152874b5a653e637cd78cfc96b31e2330d8df31085

bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre listunspent 0 0
```

2. **Criar a transação filha com taxa alta**

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre createrawtransaction
"[{\"txid\":\"45d9c6faeb2d4b43a7f7c7152874b5a653e637cd78cfc96b31e2330d8df31085\",\"vout\
":0}]" "{\"tb1qacse5r77rpxupretvy24gu3dvs9an7y0q0ewa9\":0.00009}"
```

Taxas e Mempool

🔑 Child-Pays-for-Parent (CPFP)

3. Completar a transação com taxa elevada

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre fundrawtransaction
02000000018510f38d0d33e2316bc9cf78cd37e653a6b5742815c7f7a7434b2debfac6d9450000000000fdff
...08f2b611554722d640bd9f88f00000000 "{\"feeRate\":0.00002500}"
```

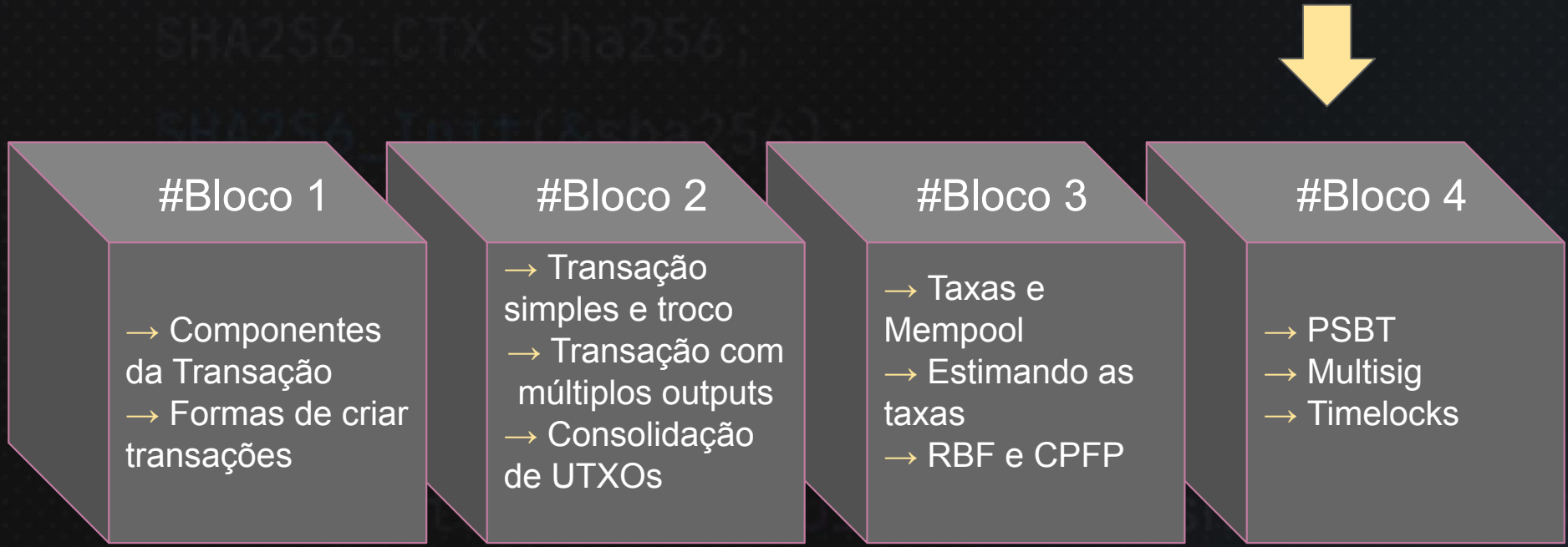
4. Assinar

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre signrawtransactionwithwallet
"02000000018510f38d0d33e2316bc9cf78cd37e653a6...fde184dc08f2b611554722d640bd9f88f00000000
"
```

5. Enviar

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendrawtransaction
"020000000001018510f38d0d33e2316bc9cf78cd37e653a6b5742815c7f7a7434b2debfac6d945000000000
...2f34a796012102eaa654af38987d55ce4be8670c180a8bcba95c402943da5d043409e564ede40b00000000
"
```

O Que Veremos!



Recursos Avançados

🔑 **PSBT – Transações Parcialmente Assinadas**

🔑 **Multisig**

🔑 **Timelocks**

Recursos Avançados

🔑 PSBT – Transações Parcialmente Assinadas

A PSBT é um padrão definido na BIP174 que permite criar, compartilhar e assinar transações de forma colaborativa.

Separa as **etapas**:

1. **Criação da transação bruta**: quem seleciona UTXOs e define os destinos.
2. **Assinatura**: cada participante assina apenas sua parte, em seu próprio dispositivo.
3. **Finalização/Broadcast**: montagem final e envio à rede.

Recursos Avançados

🔑 PSBT – Transações Parcialmente Assinadas

1. **Criar a PSBT (seleciona UTXOs e define saídas)**

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre walletcreatefundedpsbt "[]"
"[{\"tb1q6pp5uee64v3h4vr6x35pq04jhfy09e4dq5zeny\":0.0001}]" 0
"{\"subtractFeeFromOutputs\":[0]}" true
```

2. **Assinar parcialmente (na mesma carteira)**

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre walletprocesspsbt
"cHNidP8BAMMCAAAAA4sBbZJ46cI7pgHOjPKH6BywpkWiVIqr8Oycy4hXNUv1AAAAAAD9////hRDzjQ0z4jFryc9
4zTfmU6a1dCgVx/enQ0st6/rG2UUB...VQAAIABAACAAAAAgAAAAAAAAAAACICA6DcftZPeUqyEaEf1j6N1UQzw7
q8OoFDgjFR9WTnxwfDGARxEBlUAACAAQAAgAAAAIABAAAACwAAAAA="
```

Recursos Avançados

🔑 PSBT – Transações Parcialmente Assinadas

3. Finalizar e transmitir

Se complete = false, usaríamos o "psbt" retornado:

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre finalizepsbt "<PSBT_ASSINADA>"
```

Como complete = true, não precisa chamar finalizepsbt. Basta pegar o valor de hex e transmitir:

```
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendrawtransaction
"020000000001038b016d9278e9c23ba601ce8cf287e81cb0a645a2548aabf0ec9ccb8857354bf50000000000
0fdffffff8510f38d0d33e2316bc9...41549da218468c32ab3b655012103f6d6ec7df5e08cb38cc65248071f
38211f01fdddcc193f1270d475824e042a9e00000000"
```

Recursos Avançados

🔑 Multisig

Multisig é um esquema de custódia no Bitcoin em que um gasto só é válido quando um número mínimo pré-definido de chaves entre várias participantes assina a transação, como “2-de-3” ou “3-de-5”, aumentando a segurança e permitindo controle compartilhado.

Vantagens:

→ **Resistência a roubo/perda**: 1 chave comprometida não gasta os fundos (em 2-de-3); perda de 1 chave ainda permite recuperação.

→ **Herança/backup**: distribuição de chaves entre pessoas/cofres diferentes.

→ **Governança**: políticas de aprovação (m-de-n) para empresas.

Recursos Avançados

🔑 Multisig

1) Criar três carteiras (participantes)

```
bitcoin-cli -datadir="." createwallet carteira1 false false "" true true
bitcoin-cli -datadir="." createwallet carteira2 false false "" true true
bitcoin-cli -datadir="." createwallet carteira3 false false "" true true
```

2) Obter os xpubs (com fingerprint/derivação)

```
bitcoin-cli -datadir="." -rpcwallet=carteira1 listdescriptors
bitcoin-cli -datadir="." -rpcwallet=carteira2 listdescriptors
bitcoin-cli -datadir="." -rpcwallet=carteira3 listdescriptors
```

Recursos Avançados

🔑 Multisig

3) Criar carteira multisig (watch-only) e importar os descriptors multisig (recebimento e troco)

```
bitcoin-cli -datadir="." createwallet multisig-2of3 true true "" true true
```

```
bitcoin-cli -datadir="." -rpcwallet=multisig-2of3 importdescriptors
[{\"desc\":\"wsh(sortedmulti(2,[ba39c771/84h/1h/0h]tpubDCYmREHcMiW...EoKGDWj7op1atN6xeCotwk/0/*,[1b69360a/84h/1h/0h]tpubDD2gr2TXupxDBjtL...sYRUcDZtRNzd7rPeS9Fg3HMBC4x/0/*,[fd4d4620/84h/1h/0h]tpubDDnWDjzpV82Wxdr7woANy...pd6m1JogodyjLe5fRYvQ5KT/0/*))#ed8gy0hl\",\"active\":true,\"timestamp\":\"now\"},{\"desc\":\"wsh(sortedmulti(2,[ba39c771/84h/1h/0h]tpubDCYmREHcMiWLvGJ4go...7op1atN6xeCotwk/1/*,[1b69360a/84h/1h/0h]tpubDD2gr2TXupxDBjtLUnFkWVsNMYNNcP...YRUcDZtRNzd7rPeS9Fg3HMBC4x/1/*,[fd4d4620/84h/1h/0h]tpubDDnWDjzpV82Wxdr7woANy...5FqnUFabN5zbQjexYNTCW8QC16NUpd6m1JogodyjLe5fRYvQ5KT/1/*))#tcmu8qku\",\"active\":true,\"internal\":true,\"timestamp\":\"now\"}]
```

bitcoinCoders bitups

Recursos Avançados

🔑 Multisig

4) Receber fundos na multisig

```
bitcoin-cli -datadir="." -rpcwallet=multisig-2of3 getnewaddress
*Envie alguns sats para o endereço gerado
bitcoin-cli -datadir="." -rpcwallet=Carteira-Cofre sendtoaddress
"tb1q6aj46rpyc5h7ak6cnxe3xzfzqlep4krpx0mnngzufu4jqrzul65s04pakd" 0.0001
```

5) Criar a PSBT de gasto (na multisig)

```
bitcoin-cli -datadir="." -rpcwallet=multisig-2of3 walletcreatefundedpsbt "[]"
"[{\"tb1qqu7d6zmh5mpf07xd7t9aehe0re63hfqk2wh2ew6045zc5svshphq7j4eek\":0.00001}]" 0
"{\"subtractFeeFromOutputs\":[0]}" true
```

Recursos Avançados

🔑 Multisig

6) Assinar em duas carteiras (Carteira1 e Carteira2, separadamente)

- **Sequencial**: saída da carteira1 (‘psbt’ já com 1 assinatura) entra na carteira2 → geralmente já volta ‘complete: true’.
- **Paralelo**: cada um assina a **mesma PSBT original** e depois você combina.

```
bitcoin-cli -datadir="." -rpcwallet=carteira1 walletprocesspsbt
"cHNidP8BAIkCAAAAAbALDAmCzaLpmGLM3HosXOuodp03BnzZOWlsWzDBGdb6AQAAAAD9////AigjAAAAAAAAIgA
gffrg1wuWTLU//NFkDdFKcLTiMQqI...A9mgIUrpO50byq0KR8I+D6N9AEUT0z4nVY2olSI48CaxGP1NRiBUAACAA
QAAgAAAAIAAAAAAAgAAAAA="
bitcoin-cli -datadir="." -rpcwallet=carteira2 walletprocesspsbt
"cHNidP8BAIkCAAAAAbALDAmCzaLpmGLM3HosXOuodp03BnzZOWlsWzDBGdb6AQAAAAD9////AigjAAAAAAAAIgA
gffrg1wuWTLU//NFkDdFKcLTiMQqI...A9mgIUrpO50byq0KR8I+D6N9AEUT0z4nVY2olSI48CaxGP1NRiBUAACAA
QAAgAAAAIAAAAAAAgAAAAA="
```

bitcoinCoders bitups

Recursos Avançados

🔑 Multisig

7) Combine os 2 PSBT assinados

```
bitcoin-cli -datadir="." combinepsbt
"[\"cHNidP8BAIkCAAAAAbALDAmCzaLpmGLM3HosXOuodp03BnzZOWlsWzDBGdb6AQAAAAD9////AigjAAAAAAAA
IgAgffrg1wuWTLU//NFkDdFKcLTiM...QAAIABAACAAAAAgAAAAAACAAAAIgID2aAhSuk7nRvKrQpHwj4Po30ARRP
TPidVjaiVIjjwJrEY/U1GIFQAAIABAACAAAAAgAAAAAACAAAAAA==\",\"cHNidP8BAIkCAAAAAbALDAmCzaLpmG
LM3HosXOuodp03BnzZOWlsWzDBGdb6AQAAAAD9////AigjAAAAAAAAIgAgffrg1wuWTLU//NFkDdFKcLTiMQqIGX
...BAACAAAAAgAAAAAACAAAAIgID2aAhSuk7nRvKrQpHwj4Po30ARRPTPidVjaiVIjjwJrEY/U1GIFQAAIABAACAA
AAAgAAAAAACAAAAAA==\"]"
```

Recursos Avançados

🔑 Multisig

8) Finalizar e transmitir

```
bitcoin-cli -datadir="." finalizepsbt
"cHNidP8BAIkCAAAAAbALDAmCzaLpmGLM3HosXOuodp03BnzZOWlsWzDBGdb6AQAAAAD9////AigjAAAAAAAAIgA
gffrg1wuWTLU//NFkDdFKcLTiMQqI...AAAAAAAIAAAAA"

bitcoin-cli -datadir="." sendrawtransaction
"02000000000101b00b0c0982cda2e99862ccdc7a2c5ceba8769d37067cd939696c5b30c119d6fa010000000
0fdffffff022823000000000022...12103f87599a3b7aa97c0711de3ebc98c76f789638ebe29c368cc421d
5fcc72b6adfb53ae00000000"
```

Recursos Avançados

🔑 Timelocks

No Bitcoin, **timelock** é o nome genérico para mecanismos que determinam quando uma transação ou um determinado UTXO pode ser gasto. Em vez de apenas exigir uma assinatura válida, o protocolo também **verifica tempo ou altura de bloco**, criando condições de “espera obrigatória”.

- nLockTime: um campo dentro da própria transação que define o bloco mínimo (ou um timestamp) a partir do qual a rede aceita incluí-la em um bloco. É simples e útil para “agendar” pagamentos, mas quem possui as chaves pode descartar essa transação e criar outra sem o bloqueio.

- CheckLockTimeVerify (CLTV): uma **operação de script** que grava a restrição diretamente no **UTXO**. Nesse caso, o dinheiro “herda” a trava: nenhum gasto é possível antes da altura/tempo definido, mesmo que todos os donos das chaves queiram mudar.

Recursos Avançados

🔑 Timelocks

- nLockTime:

```
bitcoin-cli -datadir="." getblockcount
# 271141

bitcoin-cli -datadir="." -rpcwallet="Carteira-Cofre" createrawtransaction "[]"
"{\"tb1q4asvsg6hhv6vwxzq5uy05wf6zzusp8me5xmlzv\":0.0001}" 271570
```

*Completar, assinar e enviar a transação normalmente

Recursos Avançados

🔑 Timelocks

- CheckLockTimeVerify (CLTV):

1. Criar uma nova carteira

```
bitcoin-cli -datadir="." createwallet carteira_cltv
```

2. Criar um novo endereço e obter informações

```
bitcoin-cli -datadir="." -rpcwallet="carteira_cltv" getnewaddress
```

```
bitcoin-cli -datadir="." -rpcwallet="carteira_cltv" getaddressinfo
tb1q84ysz0zng7m8yad6zrps3uyn6reu977v85x0ga
```

```
bitcoin-cli -datadir="." -rpcwallet="carteira_cltv" getdescriptorinfo
"wsh(and_v(v:pk(03a63fe04543a8e2ca0194ba811ae076eb350c0708f5e08925db020a59bbbecf79),after(271220)))"
```

Recursos Avançados

🔑 Timelocks

- CheckLockTimeVerify (CLTV):

3. Derivar um novo endereço a partir do descritor

```
bitcoin-cli -datadir="." -rpcwallet="carteira_cltv" deriveaddresses "wsh(and_v(v:pk(03a63fe04543a8e2ca0194ba811ae076eb350c0708f5e08925db020a59bbbecf79),after(271220)))#gk4hzcry"
```

4. Enviar saldo para o endereço timelock

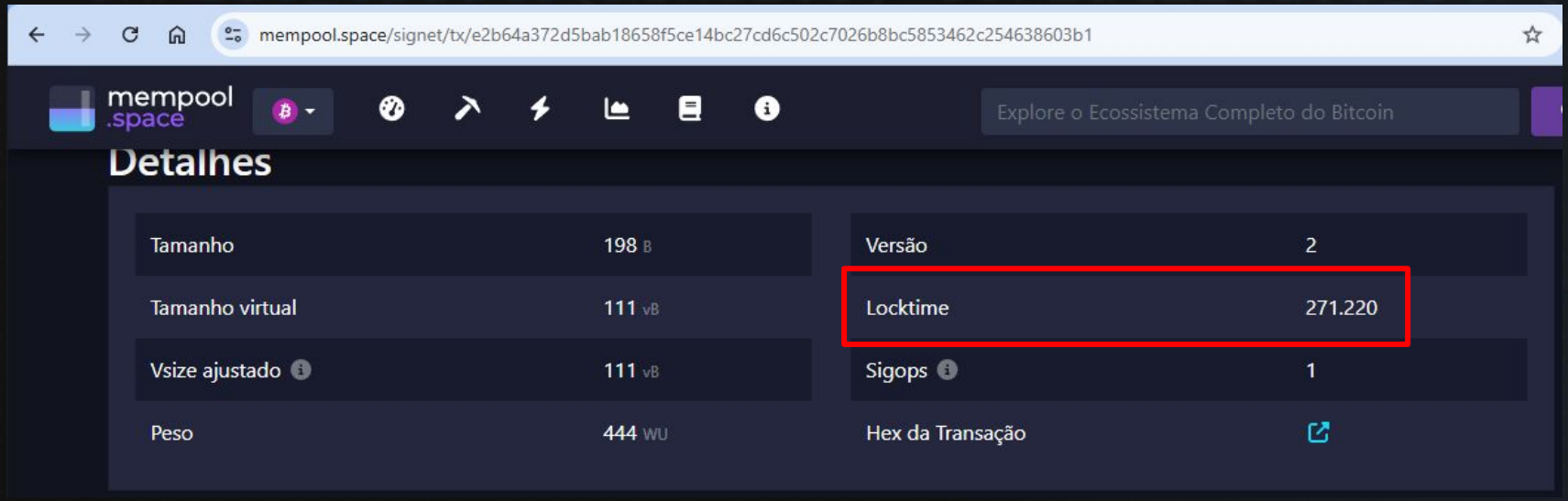
```
bitcoin-cli -datadir="." -rpcwallet="Carteira-Cofre" sendtoaddress "tb1qk89xu9tsy6v5yycevkfsjeaug2qdtqv6wcp7uk4gff24eje4239q486uq5" 0.0001
```

Recursos Avançados

🔑 Timelocks

- CheckLockTimeVerify (CLTV):

5. Verificar Locktime na mempool



FIM

Obrigado!

Grupo Whatsapp - Bitup Coders
https://chat.whatsapp.com/IB3rBamPe6QIvfncMBb3nF?mode=ems_copy_c

Rafael Penna
rapennas@gmail.com

bitcoinCoders bitups