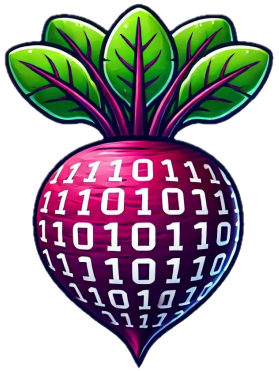
21 Million Bitcoin?

by BitRoot



1

21 Million Supply Cap?

- Bitcoin Supply Formula
- Bitcoin Source Code Analysis

2

Why can the Supply Cap **not** be changed?
(People have tried before!)

Basics Bitcoin Network

Who has control over Bitcoin within the Network?

Bitcoin Supply Formula

Each Term represents the **Bitcoin Block Reward** that goes to Bitcoin Miners within one Period (“Epoch”).

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

Bitcoin Supply Formula

Each Term represents the **Bitcoin Block Reward** that goes to Bitcoin Miners within one Period (“Epoch”).

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

The current Epoch

Bitcoin Supply Formula

Each Term represents the **Bitcoin Block Reward** that goes to Bitcoin Miners within one Period ("Epoch").

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

The maximal (initial) Block Reward.

The current Epoch

Bitcoin Supply Formula

Each Term represents the **Bitcoin Block Reward** that goes to Bitcoin Miners within one Period ("Epoch").

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

The maximal (initial) Block Reward.

The current Epoch

Reward halving after each Epoch

Bitcoin Supply Formula

Each Term represents the **Bitcoin Block Reward** that goes to Bitcoin Miners within one Period ("Epoch").

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

The maximal (initial) Block Reward.

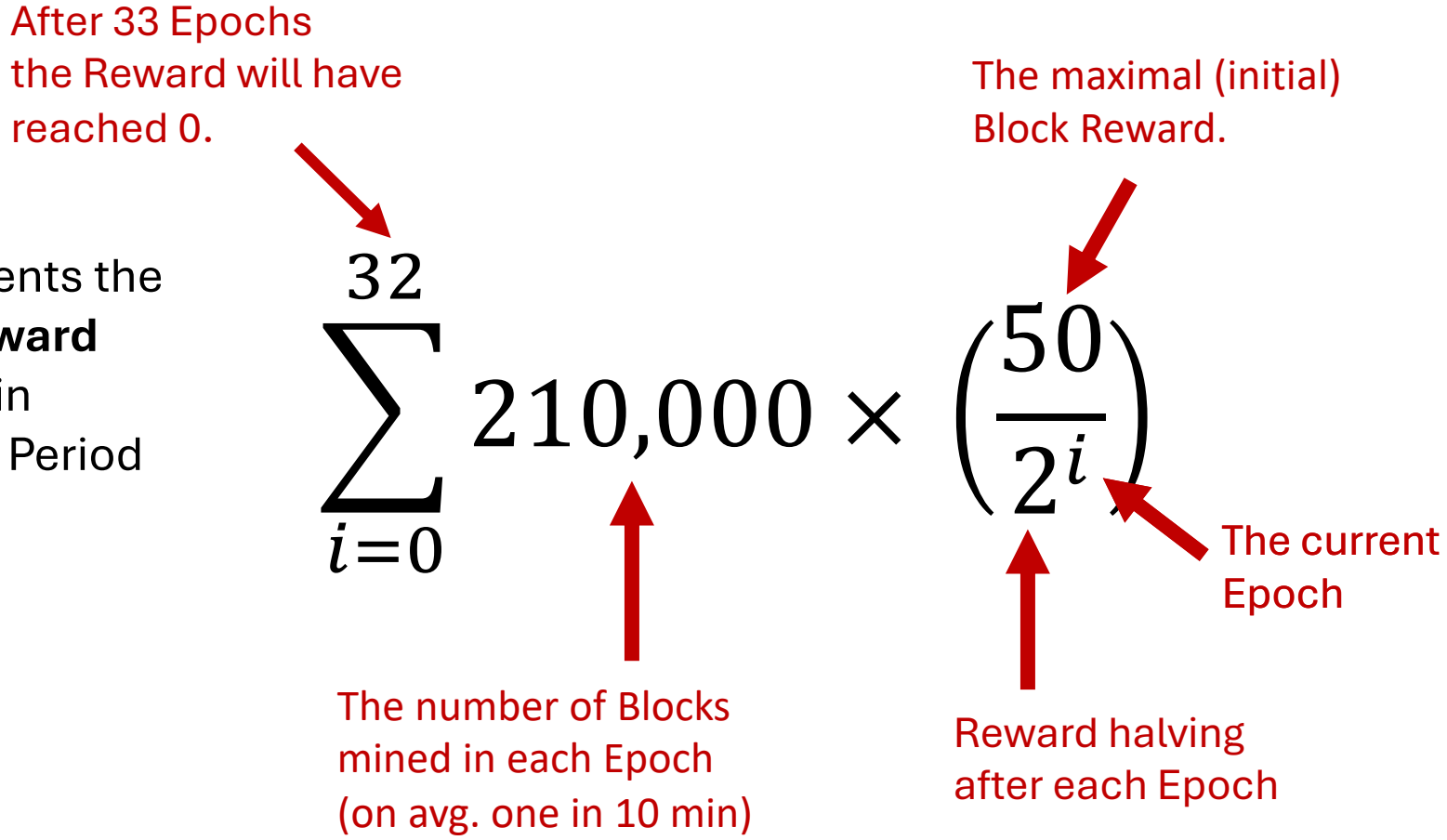
The current Epoch

The number of Blocks mined in each Epoch (on avg. one in 10 min)

Reward halving after each Epoch

Bitcoin Supply Formula

Each Term represents the **Bitcoin Block Reward** that goes to Bitcoin Miners within one Period ("Epoch").



Bitcoin Supply Formula

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right) = 210{,}000 \times \left(\frac{50}{2^0}\right) + 210{,}000 \times \left(\frac{50}{2^1}\right) + \ \ldots \ + 210{,}000 \times \left(\frac{50}{2^{32}}\right)$$

$$= \ 10{,}500{,}000 \quad + \quad 5{,}250{,}000 \quad + \cdots \ + \ 0.00000001$$

$$= \ 21{,}000{,}000$$

Bitcoin Supply Formula

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right) = 210{,}000 \times \left(\frac{50}{2^0}\right) + 210{,}000 \times \left(\frac{50}{2^1}\right) + \ \ldots \ + 210{,}000 \times \left(\frac{50}{2^{32}}\right)$$

$$= \ 10{,}500{,}000 \quad + \quad 5{,}250{,}000 \quad + \cdots \ + \ 0.00000001$$

$$= \ 21{,}000{,}000$$

In Practice: 20,999,999.9769 Bitcoin

System discards fractional results from the halving process.

Bitcoin Supply Formula

Variables required to define the Bitcoin Supply:

1. Number of Epochs: **33**
2. Blocks mined during each Epoch: **210,000**
3. The initial Block Reward was **50 Bitcoin**.
4. Bitcoin Supply **halved** at the end of each Epoch.

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

→ Max Bitcoin Supply: Almost 21,000,000

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code | Blame | 6048 lines (5365 loc) · 286 KB | Raw

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when right shift is undefined.
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blocks which will occur approximately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code Blame 6048 lines (5365 loc) · 286 KB Raw

The current block height

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when right shift is undefined.
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blocks which will occur approximately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code Blame 6048 lines (5365 loc) · 286 KB

The current block height

Contains rules for block and transaction validation. E.g. Blocks per Epoch (210,000)

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when right shift is undefined.
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blocks which will occur approximately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code Blame 6048 lines (5365 loc) · 286 KB

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when right shift is undefined.
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blocks which will occur approximately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

$$halvings = \frac{Current\ Block\ Height}{210{,}000}$$

In which halving cycle (epoch) are we currently?

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code | Blame | 6048 lines (5365 loc) · 286 KB | Raw

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when right shift is undefined.
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blocks which will occur approximately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

COIN = 100,000,000
BTC to SATS calculation

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code | Blame | 6048 lines (5365 loc) · 286 KB | Raw

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blocks which will occur approximately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

If ‘halving’ (epoch) is larger than **64**, no reward will be returned!

Source Code Analysis

? $\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$

bitcoin / src / validation.cpp ↑ Top

Code | Blame | 6048 lines (5365 loc) · 286 KB | Raw

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blocks which will occur approximately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

If ‘halving’ (epoch) is larger than **64**, no reward will be returned!

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code Blame 6048 lines (5365 loc) · 286 KB Raw

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when right shift is undefined.
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blo                    ately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

Bitshift Operator
The same effect as:

$$nSubsidy = \frac{nSubsidy}{2}$$

Source Code Analysis

Decimal to Binary Representation

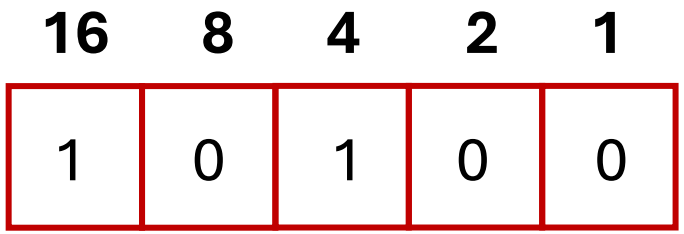
	32	16	8	4	2	1
Decimal: 1	0	0	0	0	0	1
Decimal: 2	0	0	0	0	1	0
Decimal: 3	0	0	0	0	1	1
Decimal: 35	1	0	0	0	1	1

Source Code Analysis

Right Bitshift Operator ...
moves all bits one entry to the right

Source Code Analysis

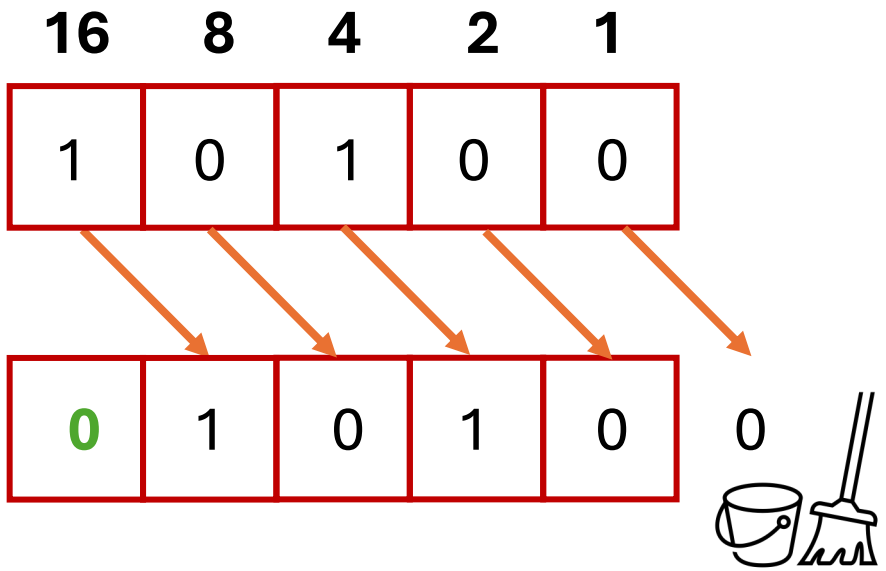
Right Bitshift Operator ...
moves all bits one entry to the right



Decimal: 20

Source Code Analysis

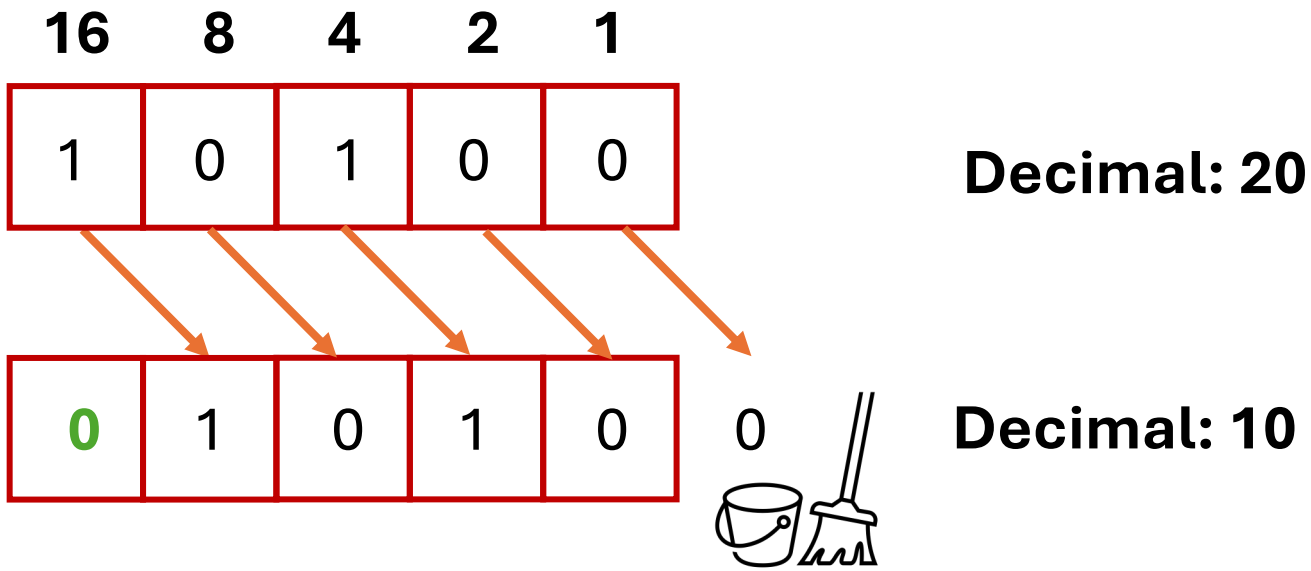
Right Bitshift Operator ...
moves all bits one entry to the right



Decimal: 20

Source Code Analysis

Right Bitshift Operator ...
moves all bits one entry to the right



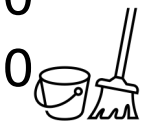
Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	1001010100000010111110010000000000	5,000,000,000

First Satoshi
(5 Billion)

Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	1001010100000010111110010000000000	5,000,000,000
1	0100101010000001011111001000000000	2,500,000,000



First Satoshi
(5 Billion)

Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	1001010100000010111110010000000000	5,000,000,000
1	0100101010000001011111001000000000	2,500,000,000
2	0010010101000000101111100100000000	1,250,000,000

First Satoshi (5 Billion)

Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	1001010100000010111110010000000000	5,000,000,000
1	0100101010000001011111001000000000	2,500,000,000
2	0010010101000000101111100100000000	1,250,000,000
3	0001001010100000010111110010000000	625,000,000

First Satoshi
(5 Billion)

Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	1001010100000010111110010000000000	5,000,000,000
1	0100101010000001011111001000000000	2,500,000,000
2	0010010101000000101111100100000000	1,250,000,000
3	0001001010100000010111110010000000	625,000,000
4	0000100101010000001011111001000000	312,500,000

First Satoshi
(5 Billion)

Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	1001010100000010111110010000000000	5,000,000,000
1	0100101010000001011111001000000000	2,500,000,000
2	0010010101000000101111100100000000	1,250,000,000
3	0001001010100000010111110010000000	625,000,000
4	0000100101010000001011111001000000	312,500,000
...		
...		
32	0000000000000000000000000000000001	1

First Satoshi
(5 Billion)

Last Satoshi

Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	1001010100000010111110010000000000	5,000,000,000
1	0100101010000001011111001000000000	2,500,000,000
2	0010010101000000101111100100000000	1,250,000,000
3	0001001010100000010111110010000000	625,000,000
4	0000100101010000001011111001000000	312,500,000
...		
...		
32	0000000000000000000000000000000001	1
33	0000000000000000000000000000000000	0

First Satoshi
(5 Billion)

Last Satoshi

Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	10010101000000101111100100000000	5,000,000,000
1	01001010100000010111110010000000	2,500,000,000
2	00100101010000001011111001000000	1,250,000,000
3	00010010101000000101111100100000	625,000,000
4	00001001010100000010111110010000	312,500,000
...		
...		
32	00000000000000000000000000000001	1
33	00000000000000000000000000000000	0
...		
63	00000000000000000000000000000000	0

First Satoshi
(5 Billion)

Last Satoshi

Source Code Analysis

Epoch	nSubsidy 64 Bit Integer in Binary Representation	nSubsidy Decimal
0	10010101000000101111100100000000	5,000,000,000
1	01001010100000010111110010000000	2,500,000,000
2	00100101010000001011111001000000	1,250,000,000
3	00010010101000000101111100100000	625,000,000
4	00001001010100000010111110010000	312,500,000
...		
...		
32	00000000000000000000000000000001	1
33	00000000000000000000000000000000	0
...		
63	00000000000000000000000000000000	0
64		UNDEFINED

First Satoshi
(5 Billion)

Last Satoshi

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code | Blame | 6048 lines (5365 loc) · 286 KB | Raw

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when right shift is undefined.
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 21 0                    4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

After 64 Bitshift Operations, the Code will return ‘undefined’ and ERROR!

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code Blame 6048 lines (5365 loc) · 286 KB Raw

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;
1671      // Force block reward to zero when right shift is undefined.
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,000 blocks which will occur approximately every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

Source Code Analysis

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

bitcoin / src / validation.cpp ↑ Top

Code Blame 6048 lines (5365 loc) · 286 KB Raw

```
1667
1668  CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams)
1669  {
1670      int halvings = nHeight / consensusPa
1671      // Force block reward to zero when r
1672      if (halvings >= 64)
1673          return 0;
1674
1675      CAmount nSubsidy = 50 * COIN;
1676      // Subsidy is cut in half every 210,0                y every 4 years.
1677      nSubsidy >>= halvings;
1678      return nSubsidy;
1679  }
1680
```

Why don't check:

```
if (halvings >= 33)
    return 0;
```

After we reached epoch number 33, nSubsidy will be 0 anyway!

Mathematical Adjustment - Bitcoin Supply from 21 to 210 Million

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

Mathematical Adjustment - Bitcoin Supply from 21 to 210 Million

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

Larger initial Bitcoin Reward

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{500}{2^i}\right)$$

Mathematical Adjustment - Bitcoin Supply from 21 to 210 Million

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{50}{2^i}\right)$$

Larger initial Bitcoin Reward

$$\sum_{i=0}^{32} 210{,}000 \times \left(\frac{500}{2^i}\right)$$

Increased Speed of Block creation

$$\sum_{i=0}^{32} 2{,}100{,}000 \times \left(\frac{50}{2^i}\right)$$

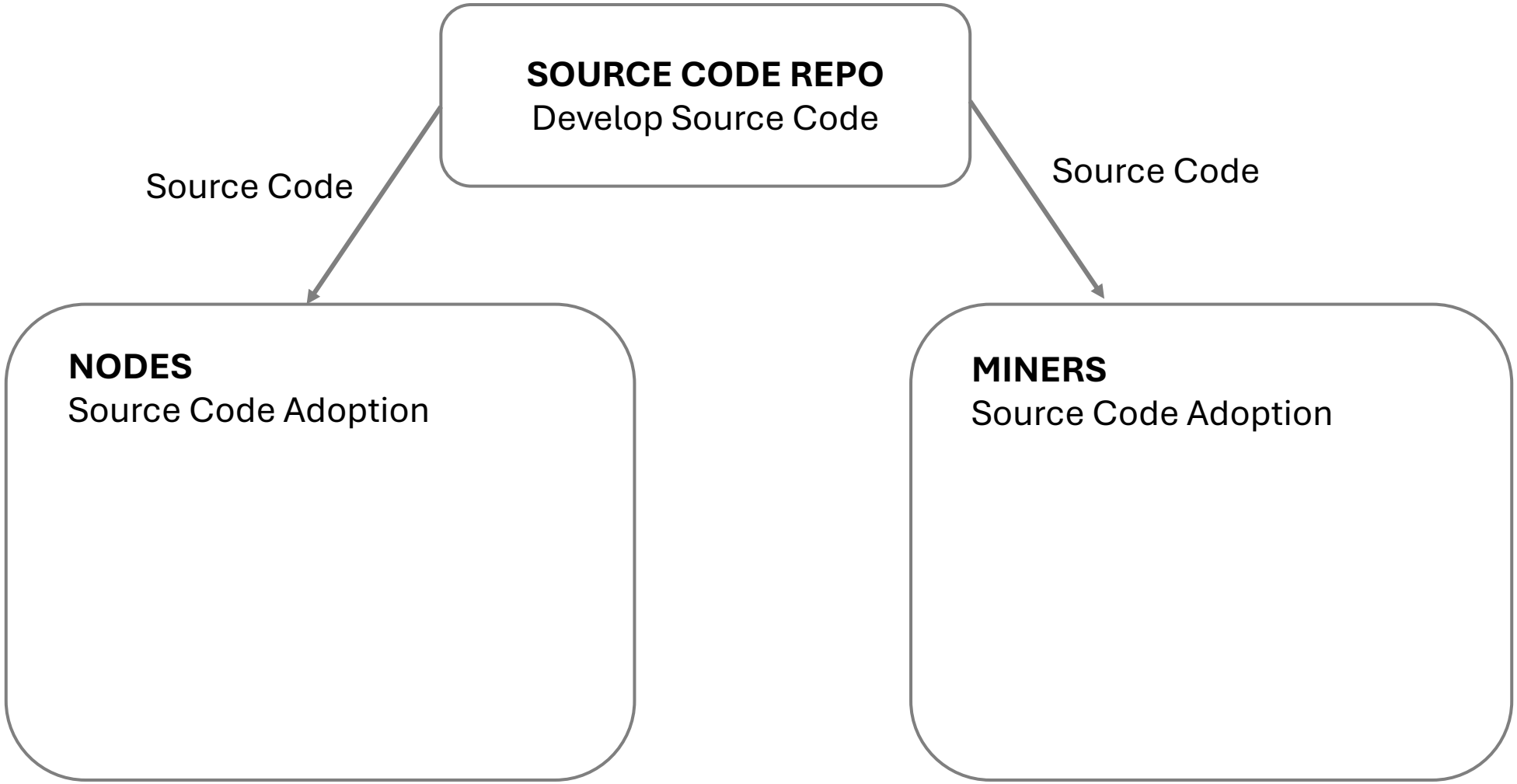
Why can the Supply Cap **not** be changed?

(People have tried!)

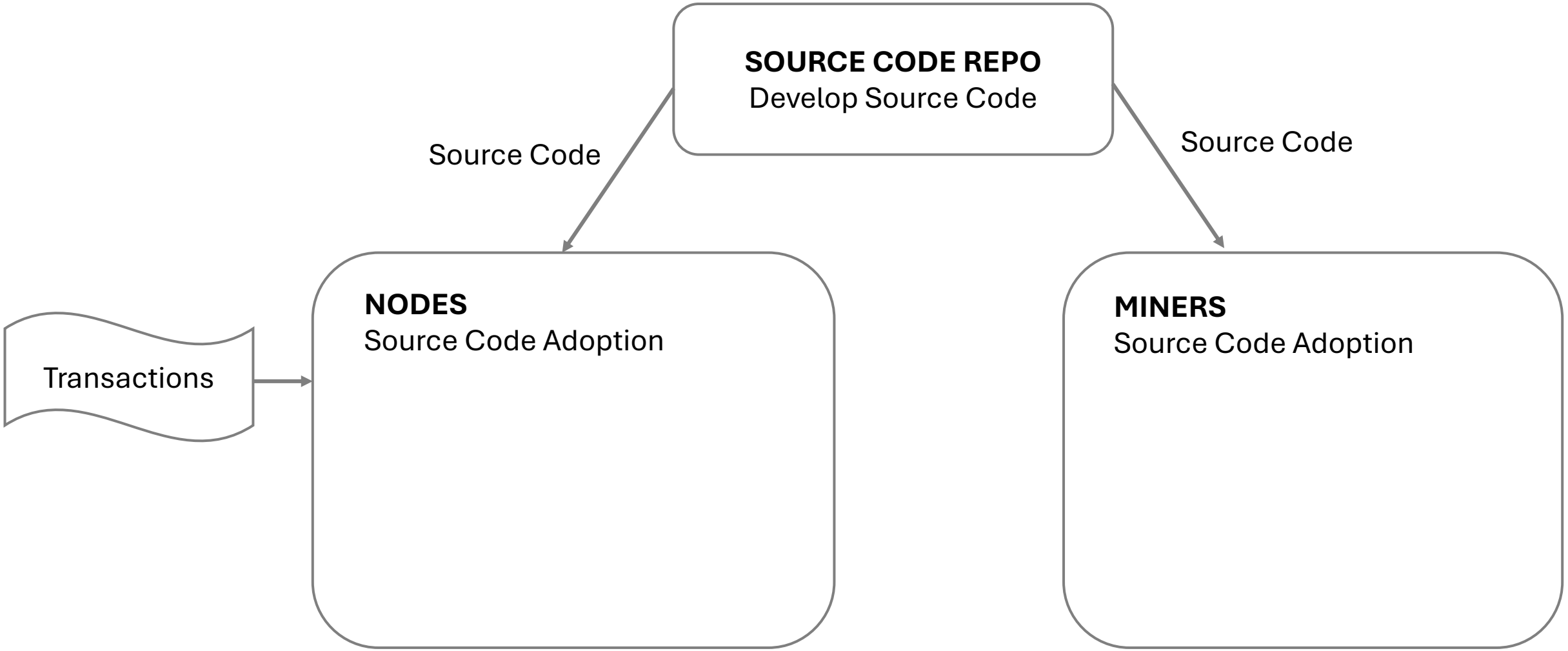
Bitcoin Network

SOURCE CODE REPO
Develop Source Code

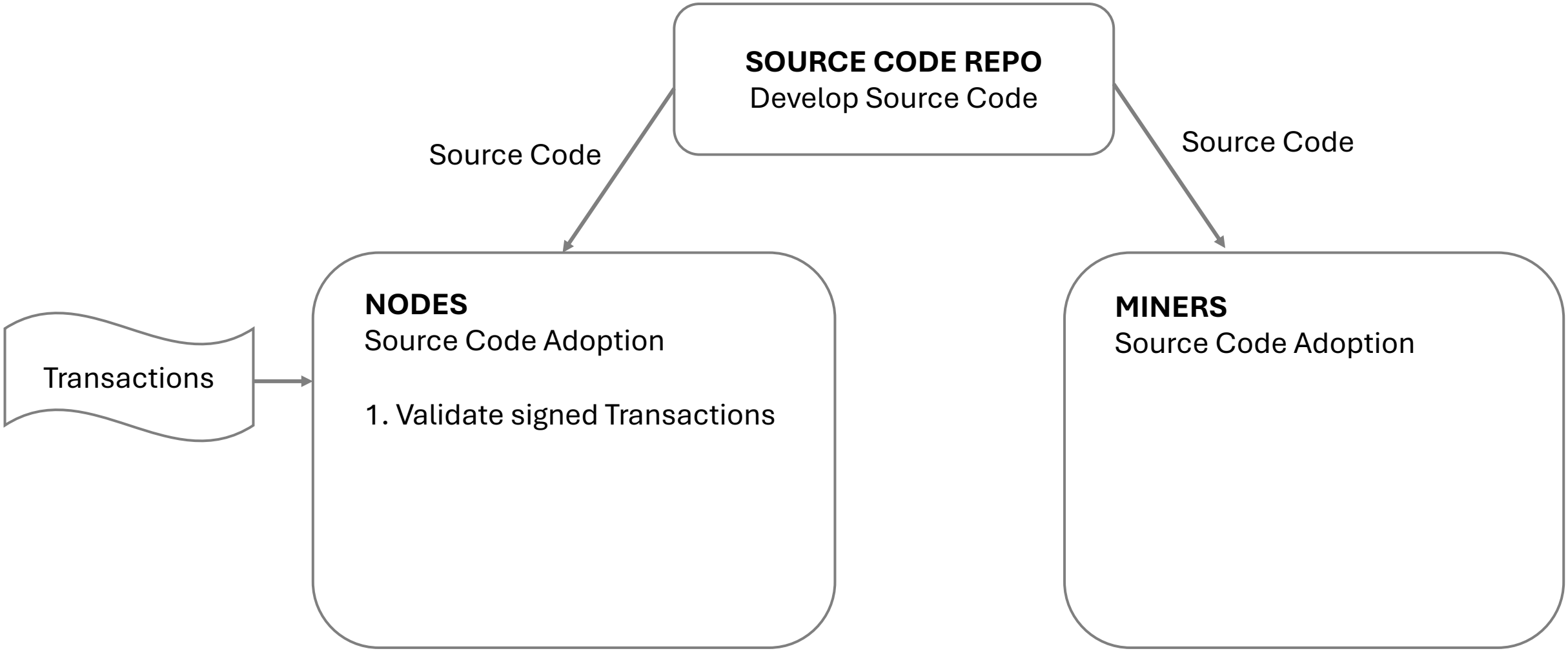
Bitcoin Network



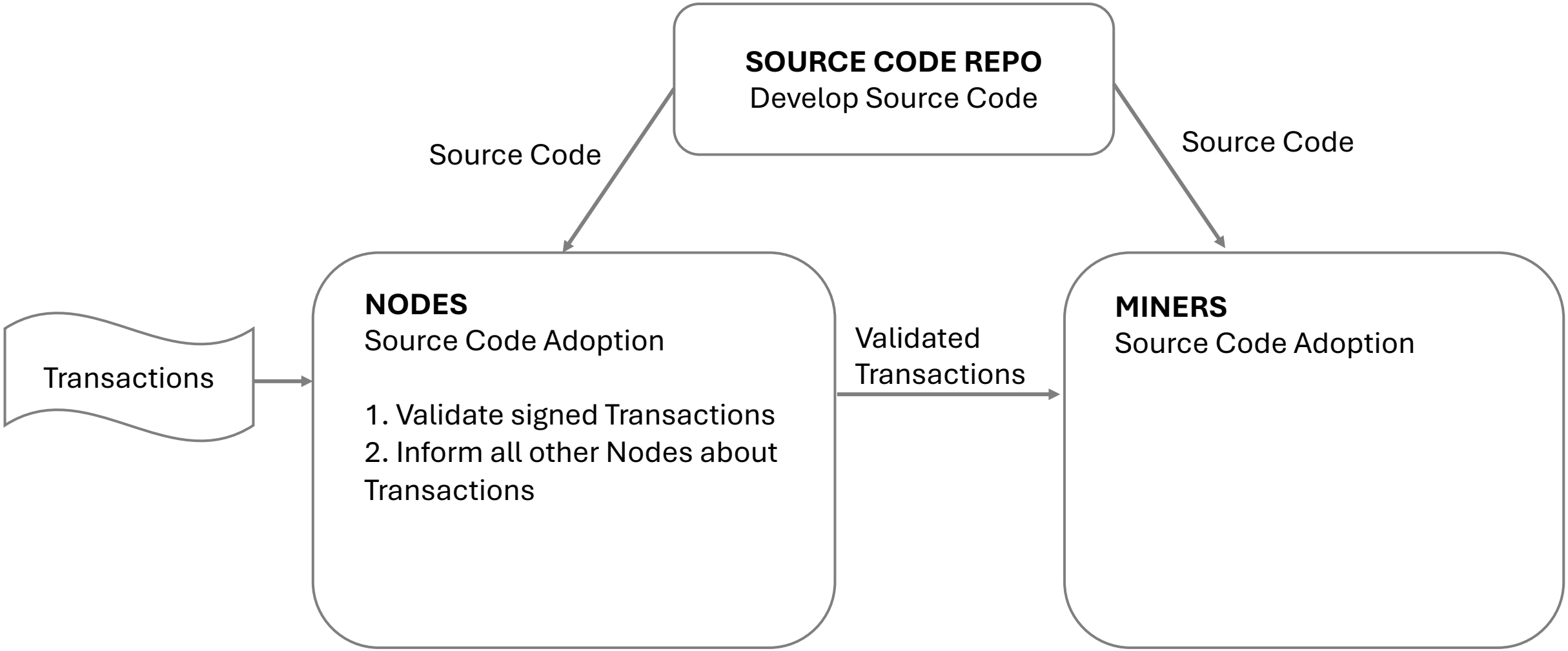
Bitcoin Network



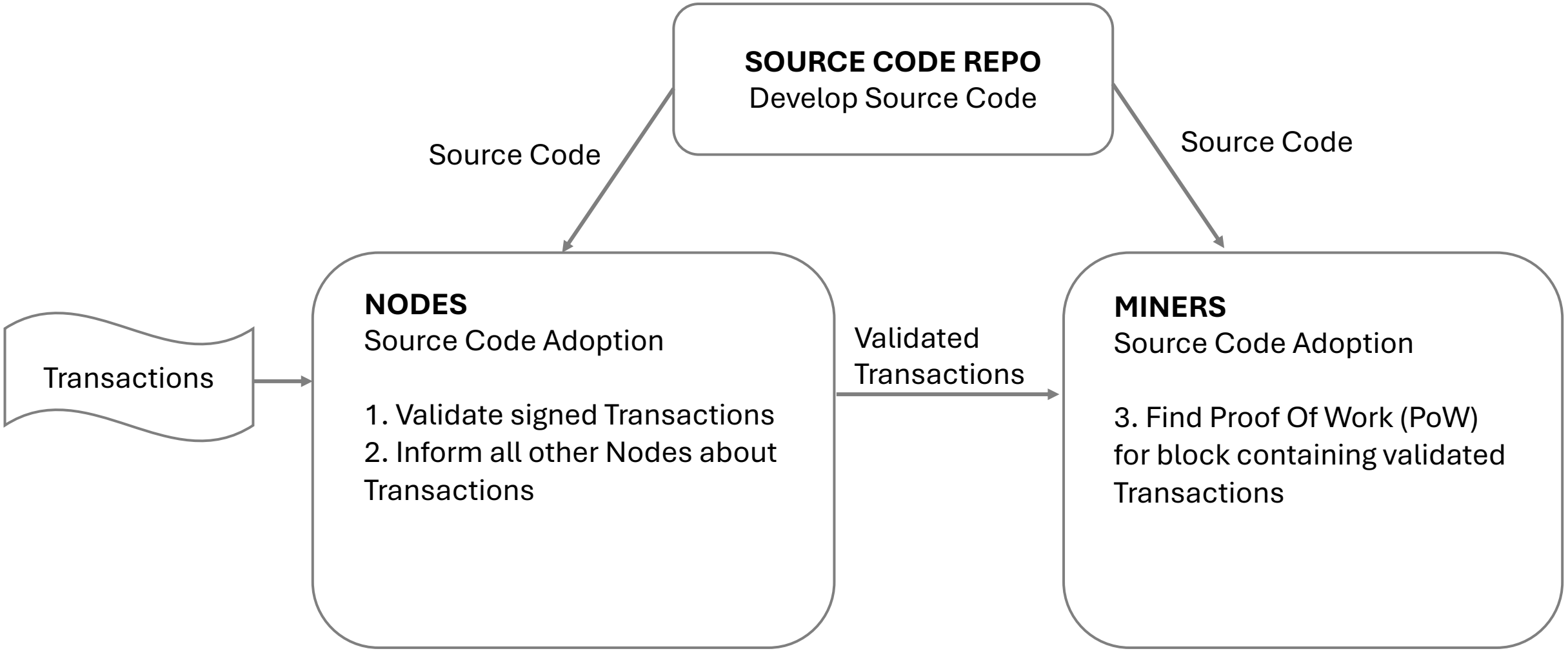
Bitcoin Network



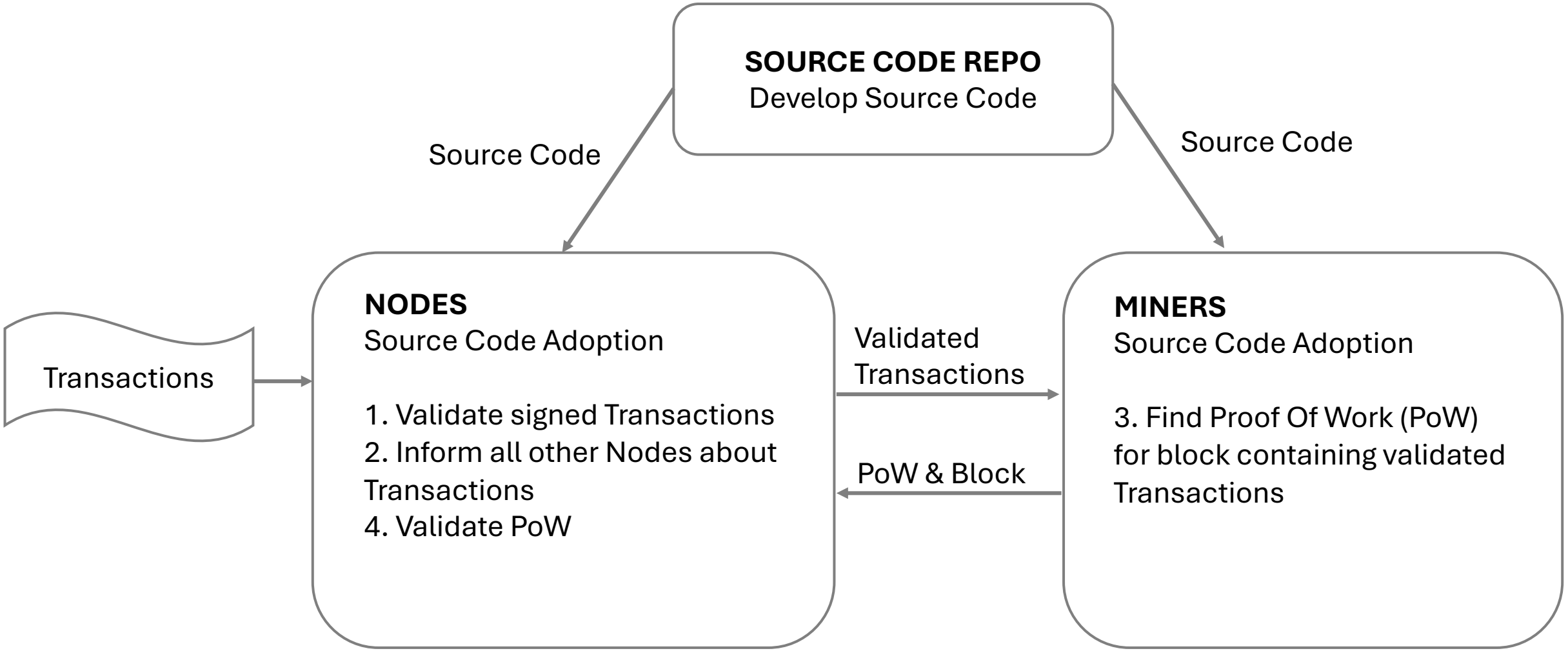
Bitcoin Network



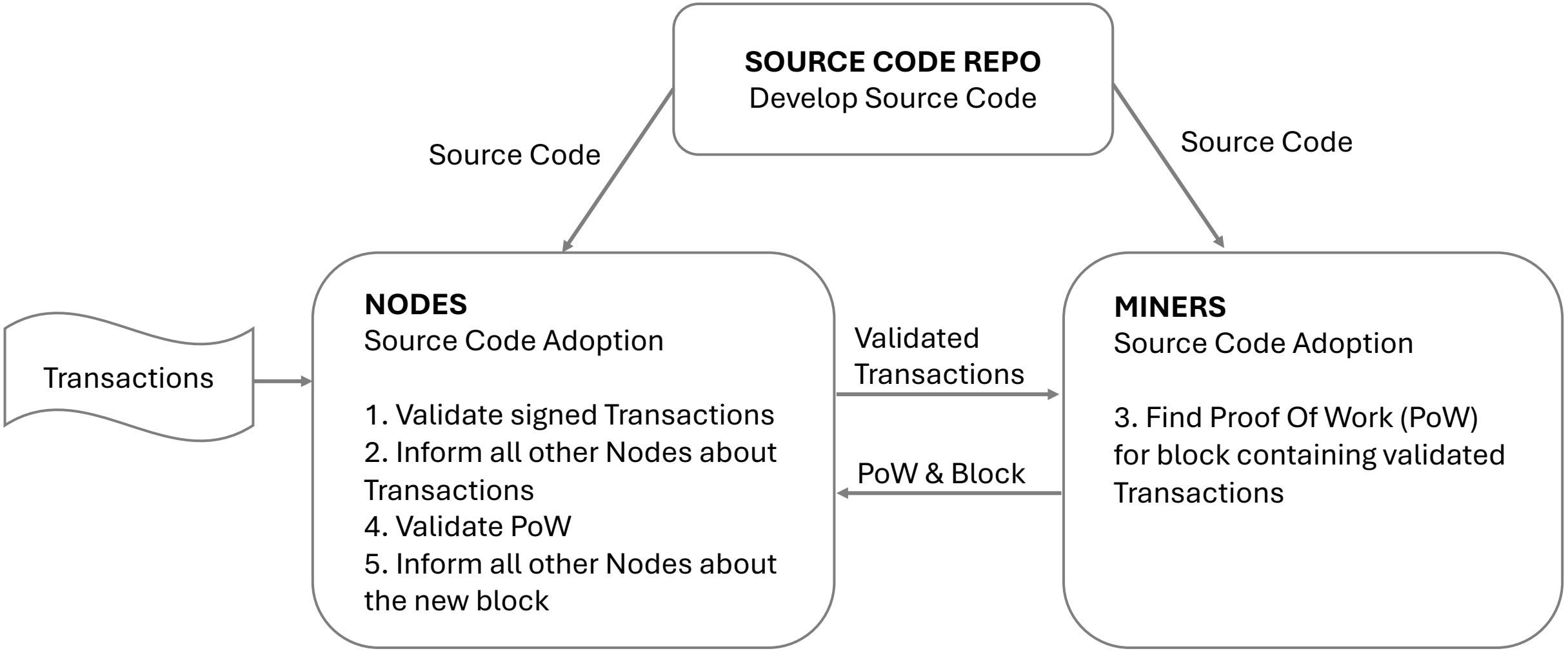
Bitcoin Network



Bitcoin Network



Bitcoin Network



Source Code Developers control Bitcoin?

Source Code Developers control Bitcoin?

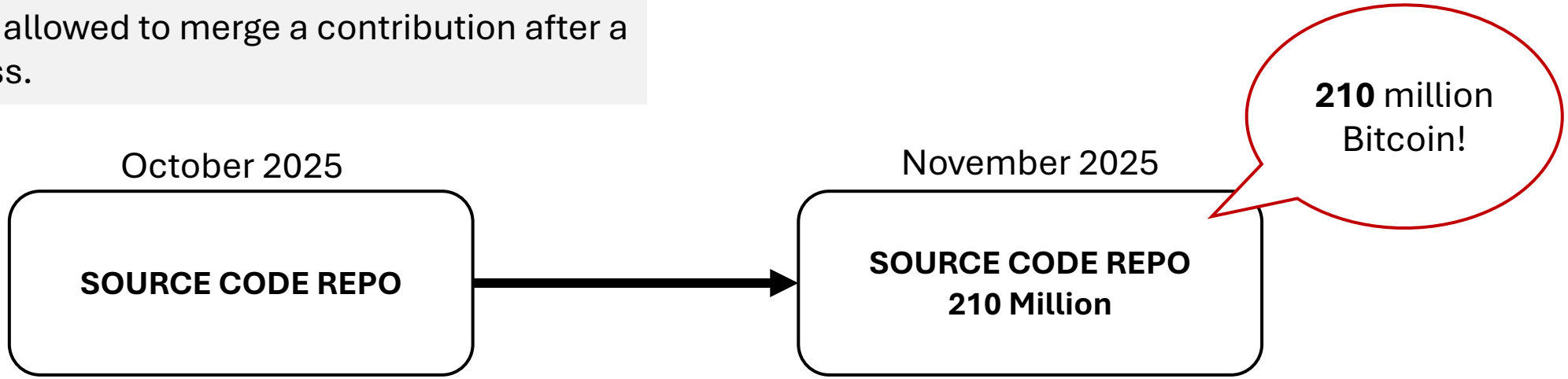
Anyone can contribute to the Source Code!

Maintainers are allowed to merge a contribution after a reviewing process.

Source Code Developers control Bitcoin?

Anyone can contribute to the Source Code!

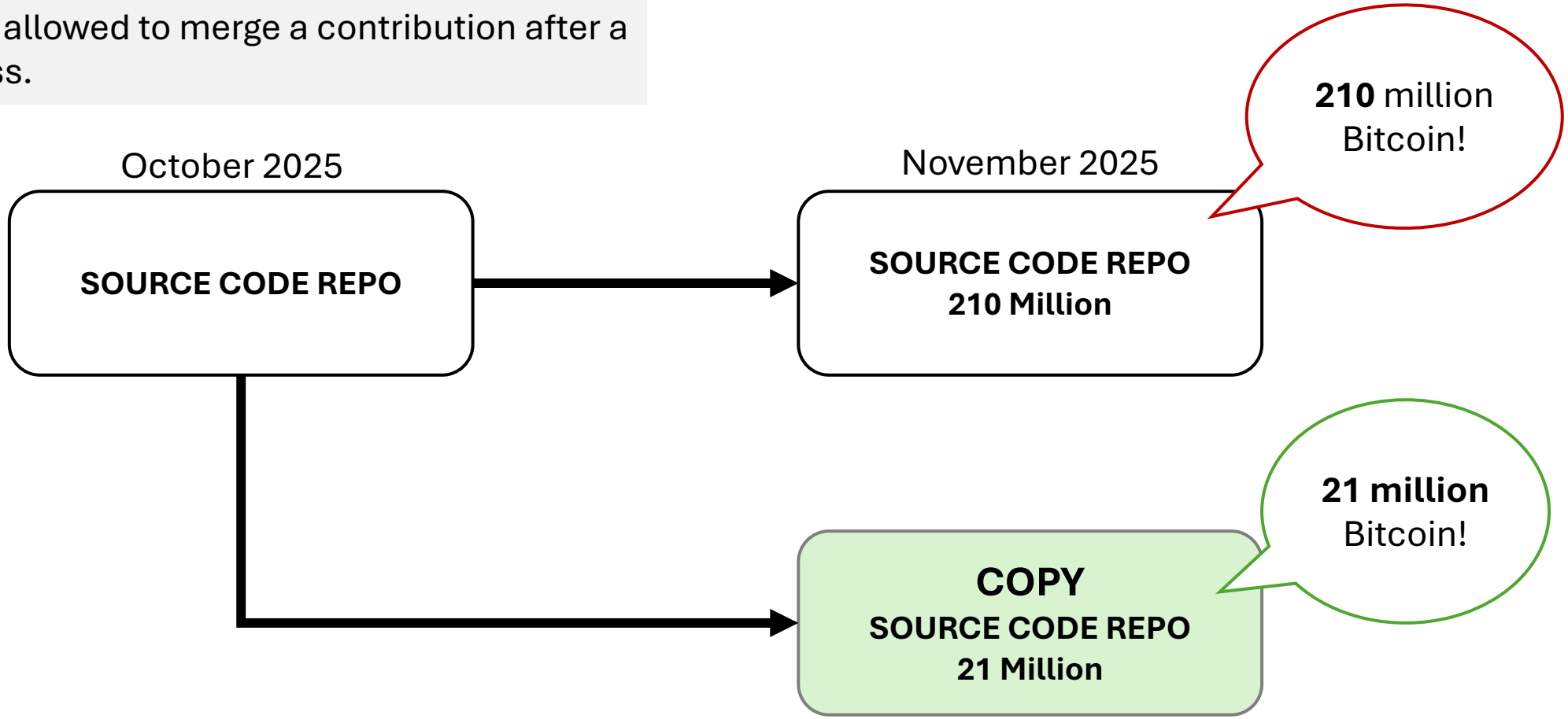
Maintainers are allowed to merge a contribution after a reviewing process.



Source Code Developers control Bitcoin?

Anyone can contribute to the Source Code!

Maintainers are allowed to merge a contribution after a reviewing process.

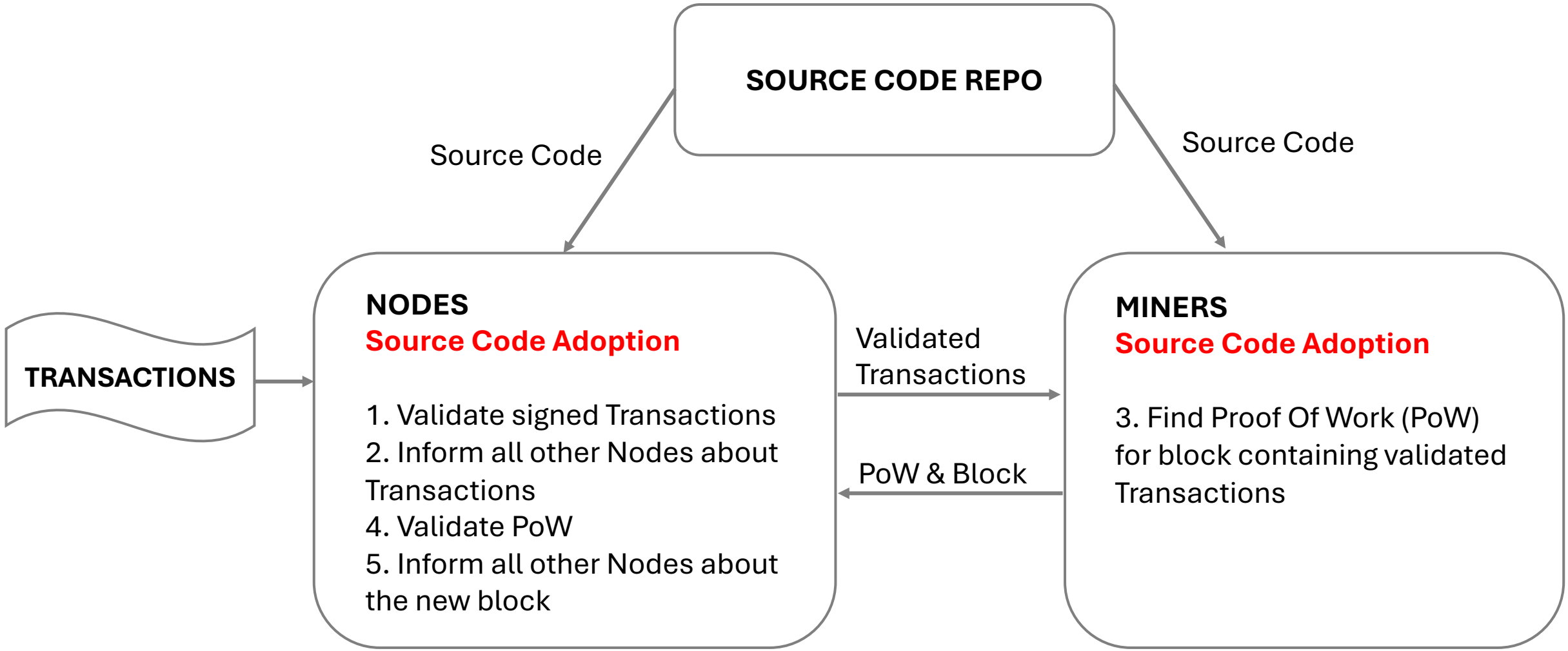


Source Code Developers control Bitcoin?

Yes!
Developers can introduce significant changes!

And No!
- *Anyone* can introduce source code changes!
- *Anyone* can copy a repository!
- No one can force owners of miners and nodes to run a specific software.

Owner of Miners and Nodes control Bitcoin?



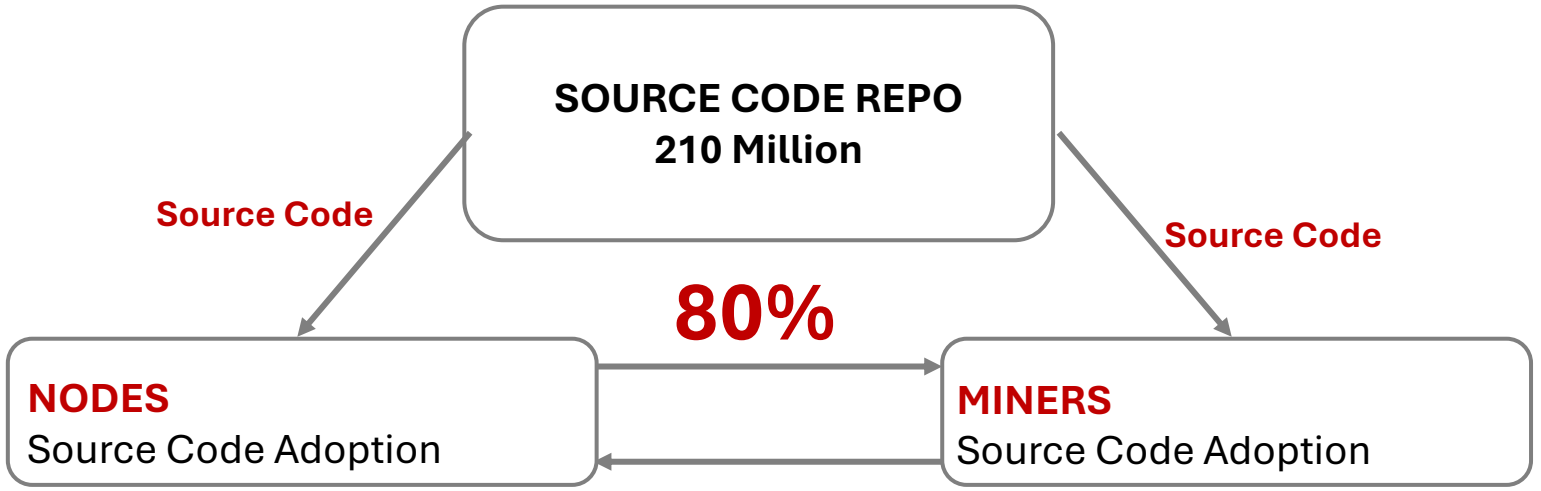
Owner of Miners and Nodes control Bitcoin?

Yes!
Miners and nodes can choose which repositories to upgrade from.

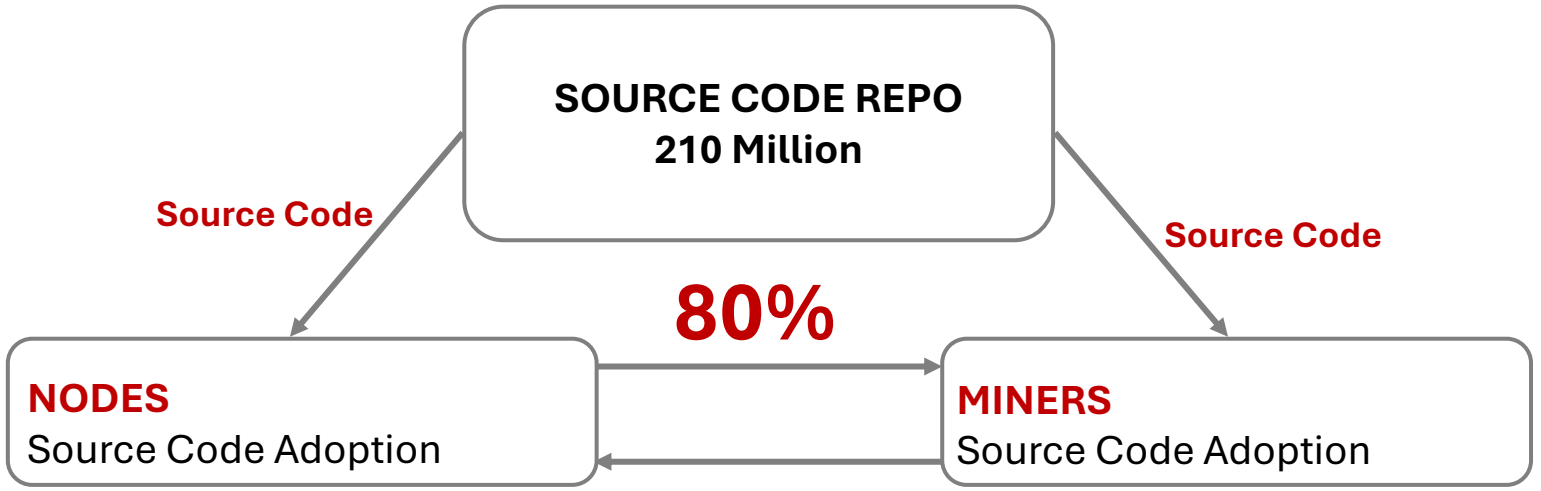
And No!

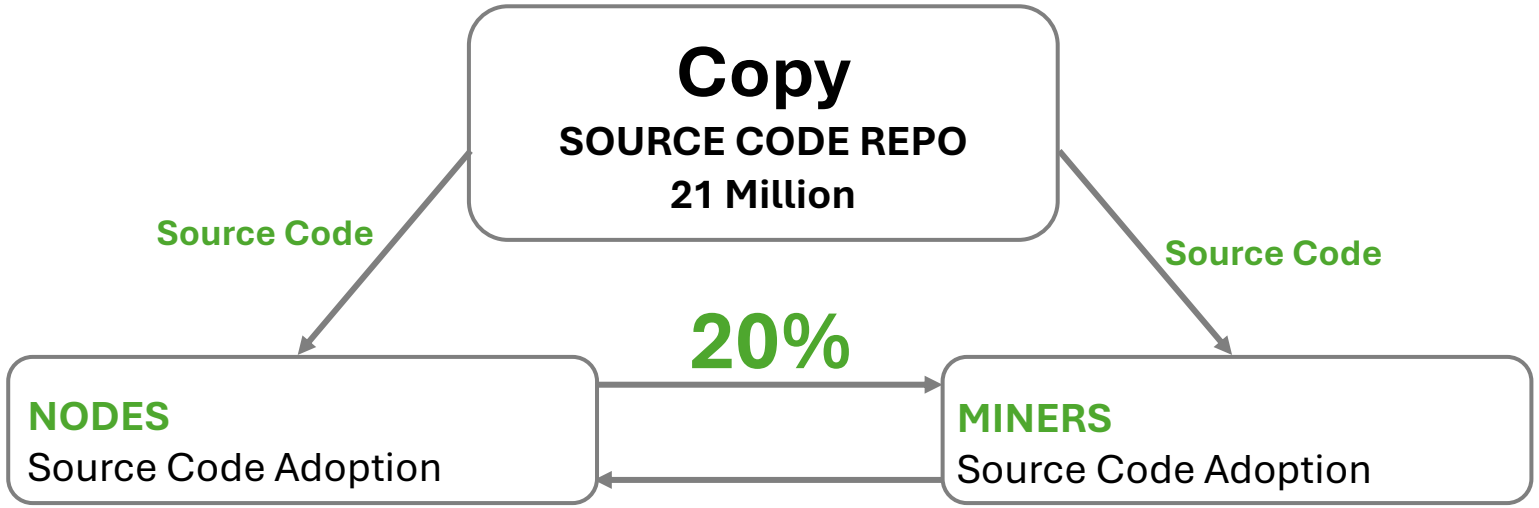
- Anyone can run a node or miner using their *preferred* source code repository.
- No one can force Bitcoin hodlers to execute transactions through specific nodes and miners.

What happens if...



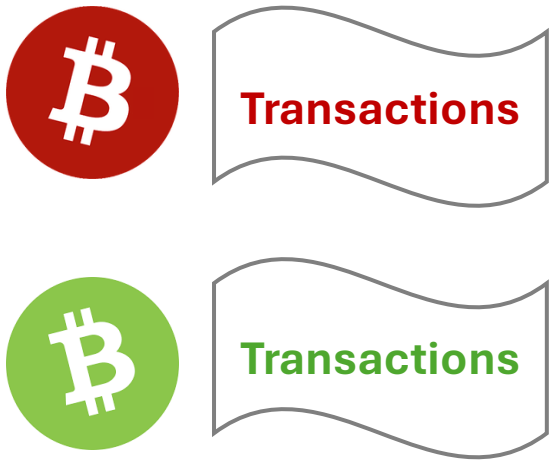
What happens if...





Hodlers control Bitcoin?

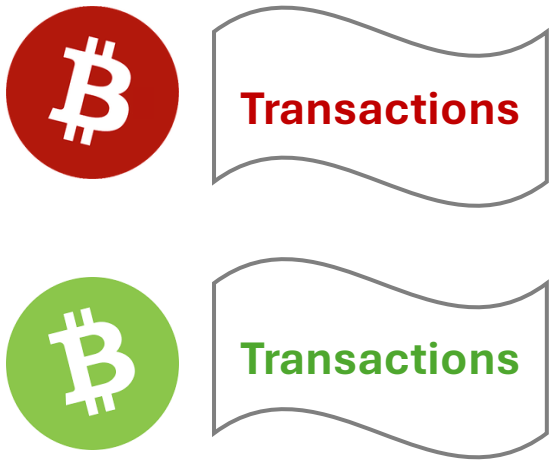
Owners **receive an equivalent amount of the new Bitcoin version** on the new, separate blockchain.



Hodlers control Bitcoin?

Owners **receive an equivalent amount of the new Bitcoin version** on the new, separate blockchain.

All Hodlers can now **trade both** Bitcoin Versions.



Hodlers control Bitcoin?

Yes!
You can decide which Bitcoin is more valuable through your buying and selling power.

And No!
You might lose out by acting against the majority, as your individual decision is not enough to control adoption.

Previous Attempt to increase the Bitcoin Supply

24th November 2017

Bitcoin Diamond was a hard fork of the Bitcoin blockchain to increase the Bitcoin supply to **210 Million.**

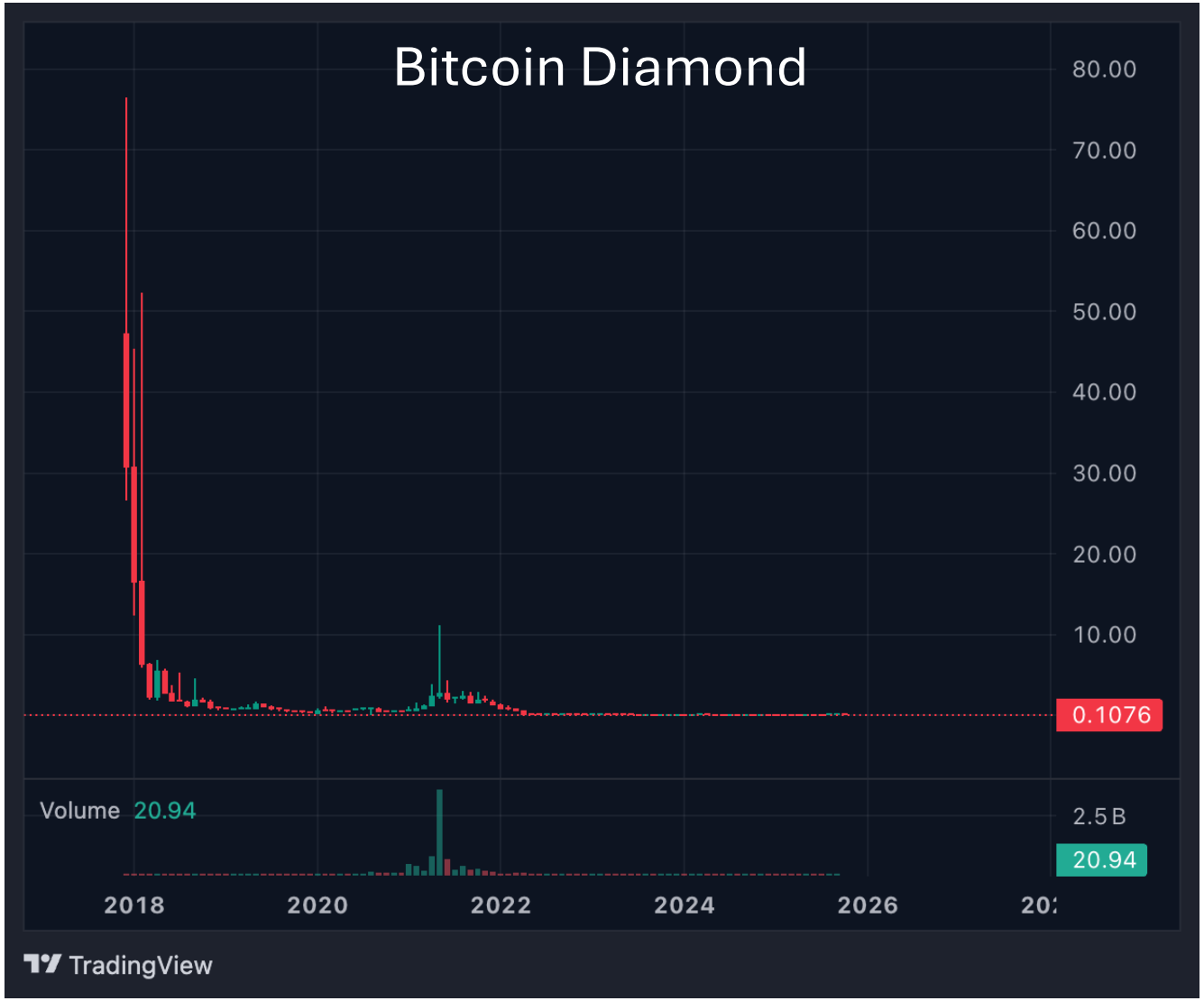
Previous Attempt to increase the Bitcoin Supply

24th November 2017

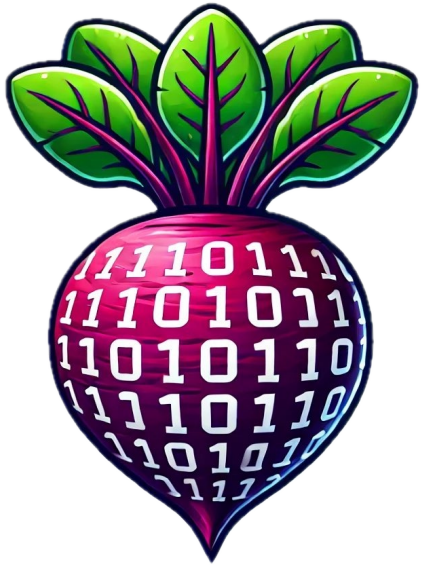
Bitcoin Diamond was a hard fork of the Bitcoin blockchain to increase the Bitcoin supply to **210 Million.**

November 2017
Bitcoin: $8

November 2025
Bitcoin: $ 115,000
Bitcoin Diamond: $0.10

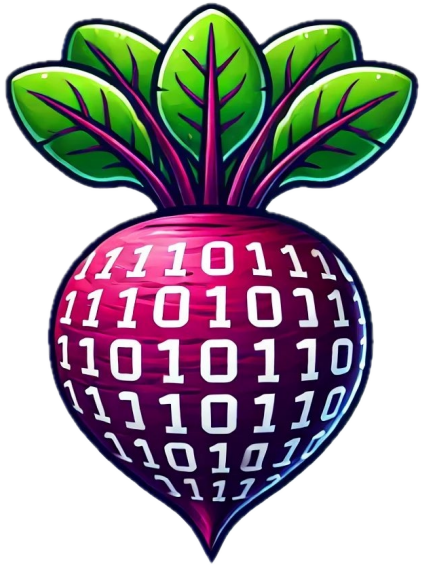


Yes, everyone and **no, no one** has control over Bitcoin and it’s supply cap.



bitroot.me

bitroot@zaps.lol



bitroot.me

bitroot@zaps.lol

cyphermunkhouse.com