Verifying an on-chain BMP vote

3 133 0 ✓ EXC BOOST

Written by **JavierGonzalez**

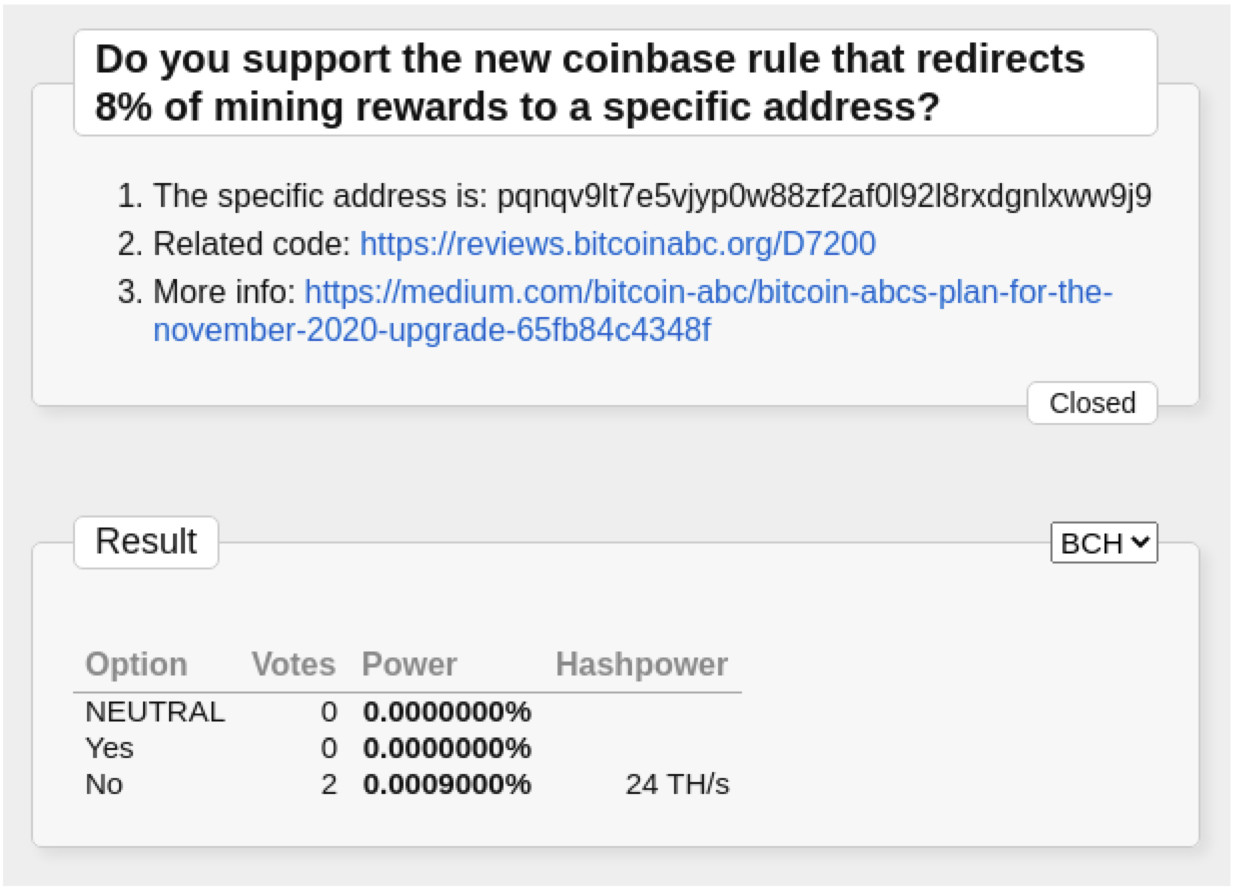
Edit article

140

5 days ago

In communities: none (Submit)

This is an example of BMP protocol hashpower voting.



Do you support the new coinbase rule that redirects 8% of mining rewards to a specific address?

1. The specific address is: pqnqv9lt7e5vjyp0w88zf2af0l92l8rxdgnlxww9j9
2. Related code: https://reviews.bitcoinabc.org/D7200
3. More info: https://medium.com/bitcoin-abc/bitcoin-abcs-plan-for-the-november-2020-upgrade-65fb84c4348f

Closed

Result

BCH

Option	Votes	Power	Hashpower
NEUTRAL	0	**0.0000000%**	
Yes	0	**0.0000000%**	
No	2	**0.0009000%**	24 TH/s

Properties

- Universally verifiable.
- Decentralized (without a central authority to create the voting or census).
- Open to the public (anyone with hashpower can participate, on equal terms).
- Inmutable (indestructible like the blockchain, on-chain, forever).
- Precise, consistent, atomic.
- Conclusive (with a start and end date).
- With multiple and extensible parameters.
- Votes changeable until close of voting.
- Parallel validity vote.
- Big `block_window` hashpower avg calculation, but allowing 24h quick votings.
- Simple and sophisticated. Following the Satoshi whitepaper.

Verification

Obviously, with 0.0009% of BCH hashpower (24 TH/s) the vote outcome carries no weight. But the technical milestone achieved is a verifiable fact. And you can verify this beyond any doubt.

The most fun way is to run your own BMP server, but here we will do it manually.

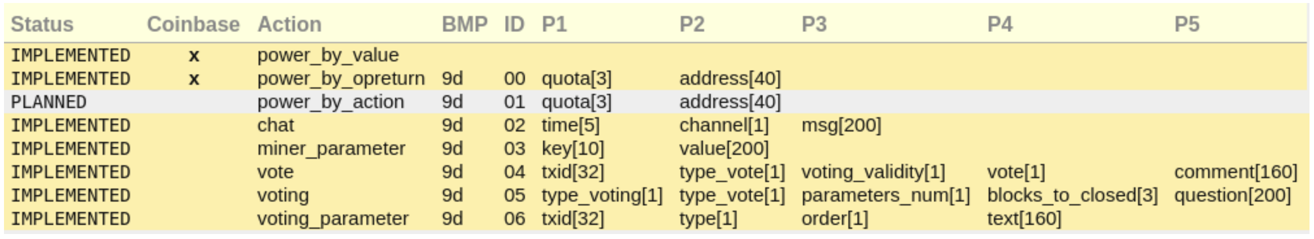
Action: voting

BMP link contains the following hash (SHA-256):
`535947de7669b7e17b6ceb99dc9bae22a8d076c99ae86fa8d920cd51966cb631`

Corresponds to this BCH transaction. It contains the following OP_RETURN hex:
`9d05000105001f80446f20796f7520737570706f727420746865206e657720636f696e6261736520726577617264206e657720736f667477617265206f6620746865206d696e696e672065766572790a626c6f636b3f`

```
9d05000105001f80446f20796f752073757070706f727420746865206e657720636f696e6261736
52072756c65207468617420726564697265637473203825206f66206d696e696e672072657761
72647320746f206120737065636966696320616464726573733f
```

The first byte `9d` is the BMP protocol prefix.



Status	Coinbase	Action	BMP	ID	P1	P2	P3	P4	P5
IMPLEMENTED	x	power_by_value							
IMPLEMENTED	x	power_by_opreturn	9d	00	quota[3]	address[40]			
PLANNED		power_by_action	9d	01	quota[3]	address[40]			
IMPLEMENTED		chat	9d	02	time[5]	channel[1]	msg[200]		
IMPLEMENTED		miner_parameter	9d	03	key[10]	value[200]			
IMPLEMENTED		vote	9d	04	txid[32]	type_vote[1]	voting_validity[1]	vote[1]	comment[160]
IMPLEMENTED		voting	9d	05	type_voting[1]	type_vote[1]	parameters_num[1]	blocks_to_closed[3]	question[200]
IMPLEMENTED		voting_parameter	9d	06	txid[32]	type[1]	order[1]	text[160]	

The second byte `05` is the action: `voting`. To create a new poll.

The third byte `00` is the type of voting: `explorative`. It means that the result is not intended to be binding. The purpose of this vote is simply to discover information about the consensus.

The fourth byte `01` is the type of vote: `one_option`. It means, each vote will choose one option. In the future there will be preferential, multiple and other more advanced types of votes.

The fifth byte `05` in decimal is 5 and is the number of *parameters* of this poll.

The next three bytes `001f80` in decimal is 8064 and is the number of blocks of duration. Voting started in block 648919, then it is closed exactly in 656983. We will call this block `close_block`.

The following bytes are the question of the voting. Decoded with `hex2bin()` is:
`Do you support the new coinbase rule that redirects 8% of mining rewards to a specific address?`

Parameters: points

Now we must find the 5 additional parameters, in order to have the voting complete.

This transaction includes the following OP_RETURN hex:
`9d06535947de7669b7e17b6ceb99dc9bae22a8d076c99ae86fa8d920cd51966cb631000154686520737065636966696320616464726573732069733a2070716e7176396c74376535766a7970307738387a66326166306c39326c38727864676e6c787777396a39`

The second byte `06` is the action: `voting_parameter`. Is used for additional metadata associated with the voting.

The next 32 bytes are:
`535947de7669b7e17b6ceb99dc9bae22a8d076c99ae86fa8d920cd51966cb631`
And it is the hash of the initial transaction of this vote. Both transactions have the same address as input. The same voting author. Thus, voting is defined by multiple TX and at the same time is atomic and immutable.

The next byte `00` is the type of parameter: `point`. This is a text that is part of the question statement, in the form of numbered points.

The next byte `01` is the point order. Specifies the order in which the numbered points should be presented.

The following bytes is the text of the point. Decoded with `hex2bin()` is:
`The specific address is: pqnqv9lt7e5vjyp0w88zf2af0l92l8rxdgnlxww`

There are two more parameters of the same type, which you will find here and here.

Parameters: options

The `voting` action says it has 5 parameters. We have verified 3. Two more to go.

This transaction includes the following OP_RETURN hex:
`9d06535947de7669b7e17b6ceb99dc9bae22a8d076c99ae86fa8d920cd51966cb6310101596573`

The first byte `9d` is the BMP protocol prefix.

The second byte `06` is the action: `voting_parameter`.

The next 32 bytes is the hash of the voting transaction.

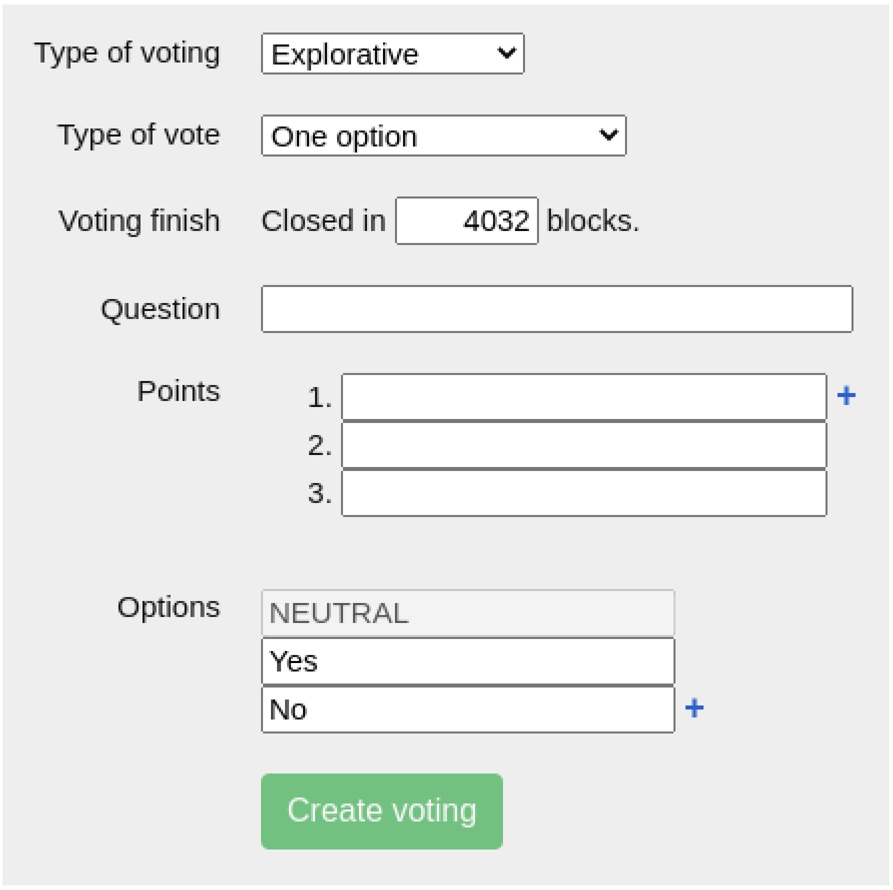
The next byte `01` is the type of parameter: `option`. This is a text that is part of the question statement, in the form of numbered points.

The next byte `01` is the option order. Specifies the order in which the point should be presented in the vote selector.

The following bytes `596573` is the text of the option. Decoded is: `Yes`

This is the last parameter, which is processed in the same way. It is the `No` option.

Thus, voting is completely created. It was created with this clean web interface.



Type of voting	Explorative
Type of vote	One option
Voting finish	Closed in 4032 blocks.
Question	
Points	1.
	2.
	3.
Options	NEUTRAL
	Yes
	No

Votes

This transaction includes the following OP_RETURN hex: `9d04535947de7669b7e17b6ceb99dc9bae22a8d076c99ae86fa8d920cd51966cb631010102`

The second byte `04` is the action: `vote`. These are the votes. Valid when included in a block while the vote is open.

The next 32 bytes are the voting hash transaction.

The next byte `01` is the type of vote: `one_election`. In the future, other elements and concepts can be voted. For example, chat messages.

The next byte `01` is the validity of the voting. `00` means that the voting is not valid, `01` the voting is valid. This allows a parallel and independent vote, on the wording of the voting itself. Independent of the option you have voted.

The next byte `02` is the voted option! `00` is always `Neutral`. In this case: `01` is `Yes` and `02` is `No`.

This is the second vote. And it is processed in the same way.

Hashpower

When the voting is closed, the voting result is computed by adding up the hashpower of each option. The option with the most hashpower is the winner.

In the first vote, the input and output include the same address:
`bitcoincash:qpjf08mqnw5g2v4y9yta50s3exek3yp6fcwddtteak`

The same address in legacy format:
`1AAtD721LQekC6ncHbAp4ScKxSwR7fFeYT`

It is the author of the vote.

This is a **coinbase transaction**. It is very special. Only a miner can do this. It is the first transaction of the block. This transaction has no inputs. It receives the fees and the block reward. This is how all Bitcoins originated.

The coinbase transaction includes multiple output addresses. Miners. The address of the first vote, ending in `FeYT`, is included.

It receives `0.04374541 BCH`. The block total output is `6.25150088 BCH`.
Then `FeYT` address has received **0.7%** of the reward of that block.

The difficulty of that 653362 block is `359263959306.8658`.
Only this number is needed to calculate the hashpower of the block with:
`Difficulty * 2^32 / 60 = 2571711593090800000 = 2.57 EH/s`
This is 2,570,000,000,000,000,000 hashes per second for that block!

Then, if `FeYT` received 0.7% of the block incentive, proportionally the BMP calculates its hashpower at `18 PH/s`. This signaling method, called `power_by_value`, is by default and it is compatible with all blocks. There are two more sophisticated methods of signaling that I will explain on another article.

The BMP `block_window` is the last 4032 blocks for hashpower calculation. Then, when the voting was close, it counted the hashpower of each miner, between 652951 `close_block - block_window` and 656983 `close_block`.

Between these blocks, `FeYT` participated in the creation of: 653326 and 653435.

The BMP calculates the proportional hashpower and adds it all up. It does the same with the second participant. And since they have voted the same, it totals it, resulting in the `24 TH/s` for the `No` option.

This new context is processed with this piece of PHP code. On every block.

21

$ 7.82 + 2

Encourage the author to write more!

Written by **JavierGonzalez**

Edit article

140

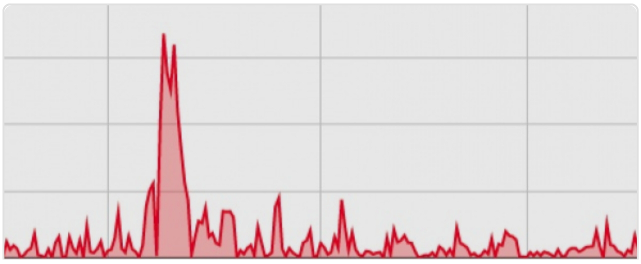
5 days ago

In communities: none (Submit)

Enjoyed this article? **Earn Bitcoin Cash by sharing it!** Explain
...and you will also help the author collect more tips.

More articles by JavierGonzalez

为什么比特币现金需要BMP系统？



Executive Hashpower

Why I Sold All My Hex