Non Custodial Sidechains for Bitcoin utilizing Plasma Cash and Covenants

(research in progress)

Georgios Konstantopoulos
Independent Consultant & Researcher
Twitter: @gakonst / me@gakonst.com

Slides available: gakonst.com/cesc2019.pdf

Related Work

Plasma: Autonomous Scalable Smart Contracts, Poon, Buterin

Plasma EthResearch Forum, too many contributors

NOCUST – A Securely Scalable Commit-Chain, Khalil, Gervais, Felley

CoinCovenants using SCIP signatures, an amusingly bad idea, Maxwell

Preventing Consensus Fraud with Commitments and Single-Use-Seals, Todd

Minimal Viable Merged Consensus, Adler

...

How do we scale?

1. Increase semantic density of transactions

   (Segwit / MAST / Schnorr / Taproot / … / **Layer 2**)

2. ~~Bigger blocks~~

Where it all started.

Enabling Blockchain Innovations with Pegged Sidechains

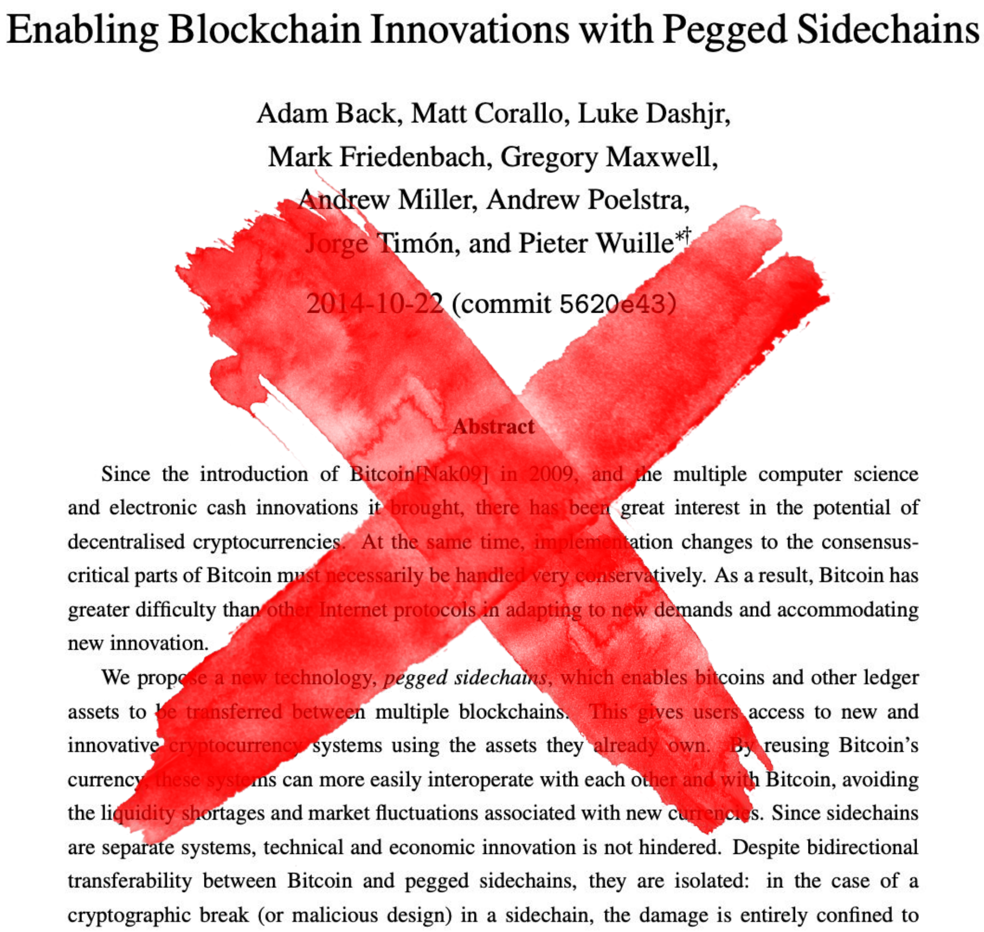
Adam Back, Matt Corallo, Luke Dashjr,
Mark Friedenbach, Gregory Maxwell,
Andrew Miller, Andrew Poelstra,
Jorge Timón, and Pieter Wuille*†

2014-10-22 (commit 5620e43)

Abstract

Since the introduction of Bitcoin[Nak09] in 2009, and the multiple computer science and electronic cash innovations it brought, there has been great interest in the potential of decentralised cryptocurrencies. At the same time, implementation changes to the consensus-critical parts of Bitcoin must necessarily be handled very conservatively. As a result, Bitcoin has greater difficulty than other Internet protocols in adapting to new demands and accommodating new innovation.

We propose a new technology, *pegged sidechains*, which enables bitcoins and other ledger assets to be transferred between multiple blockchains. This gives users access to new and innovative cryptocurrency systems using the assets they already own. By reusing Bitcoin's currency, these systems can more easily interoperate with each other and with Bitcoin, avoiding the liquidity shortages and market fluctuations associated with new currencies. Since sidechains are separate systems, technical and economic innovation is not hindered. Despite bidirectional transferability between Bitcoin and pegged sidechains, they are isolated: in the case of a cryptographic break (or malicious design) in a sidechain, the damage is entirely confined to



Enabling Blockchain Innovations with Pegged Sidechains

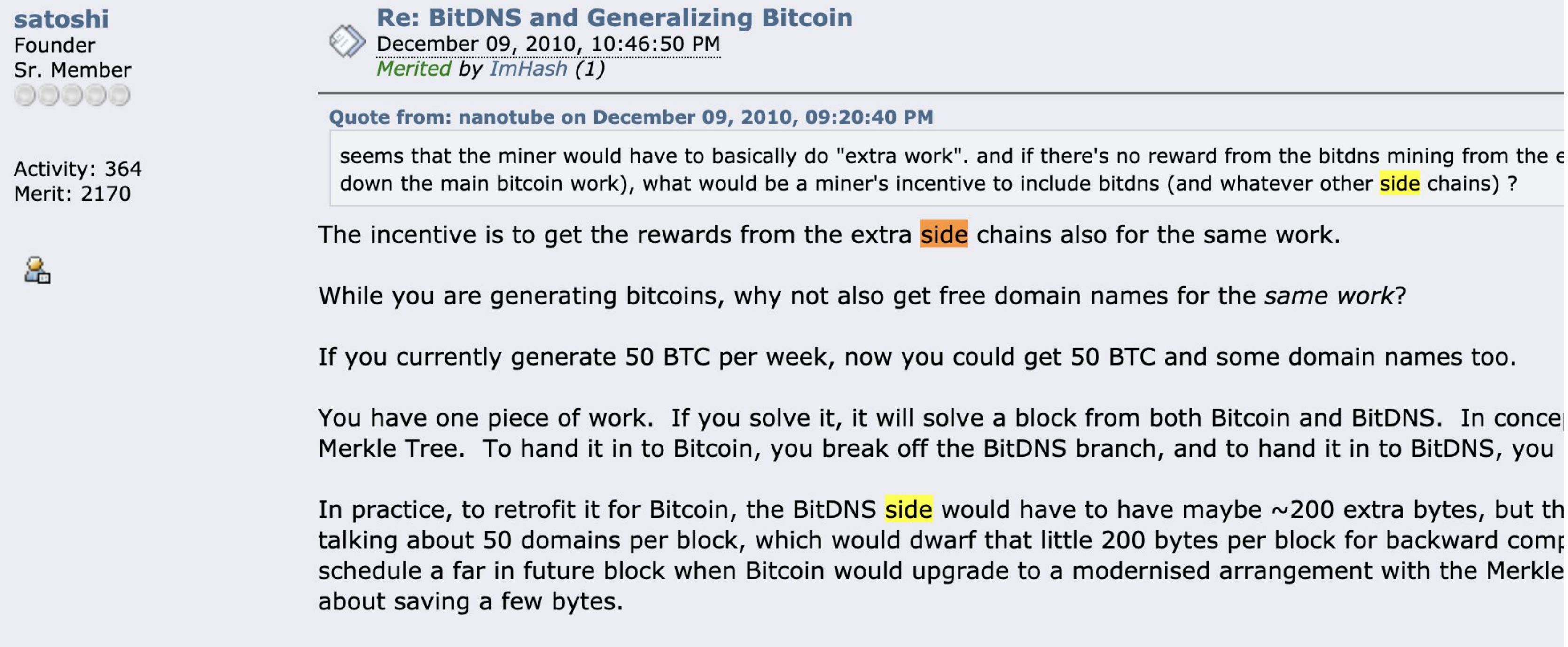
Adam Back, Matt Corallo, Luke Dashjr,
Mark Friedenbach, Gregory Maxwell,
Andrew Miller, Andrew Poelstra,
Jorge Timón, and Pieter Wuille*†

2014-10-22 (commit 5620e43)

Abstract

Since the introduction of Bitcoin[Nak09] in 2009, and the multiple computer science and electronic cash innovations it brought, there has been great interest in the potential of decentralised cryptocurrencies. At the same time, implementation changes to the consensus-critical parts of Bitcoin must necessarily be handled very conservatively. As a result, Bitcoin has greater difficulty than other Internet protocols in adapting to new demands and accommodating new innovation.

We propose a new technology, *pegged sidechains*, which enables bitcoins and other ledger assets to be transferred between multiple blockchains. This gives users access to new and innovative cryptocurrency systems using the assets they already own. By reusing Bitcoin's currency, these systems can more easily interoperate with each other and with Bitcoin, avoiding the liquidity shortages and market fluctuations associated with new currencies. Since sidechains are separate systems, technical and economic innovation is not hindered. Despite bidirectional transferability between Bitcoin and pegged sidechains, they are isolated: in the case of a cryptographic break (or malicious design) in a sidechain, the damage is entirely confined to



satoshi
Founder
Sr. Member

Activity: 364
Merit: 2170

Re: BitDNS and Generalizing Bitcoin
December 09, 2010, 10:46:50 PM
Merited by ImHash (1)

Quote from: nanotube on December 09, 2010, 09:20:40 PM

> seems that the miner would have to basically do "extra work". and if there's no reward from the bitdns mining from the e
> down the main bitcoin work), what would be a miner's incentive to include bitdns (and whatever other side chains) ?

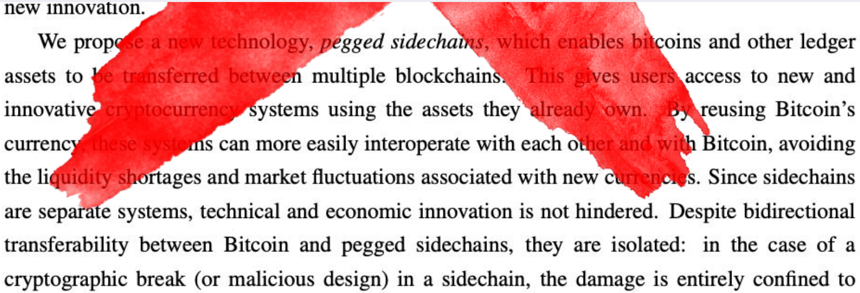
The incentive is to get the rewards from the extra side chains also for the same work.

While you are generating bitcoins, why not also get free domain names for the *same work*?

If you currently generate 50 BTC per week, now you could get 50 BTC and some domain names too.

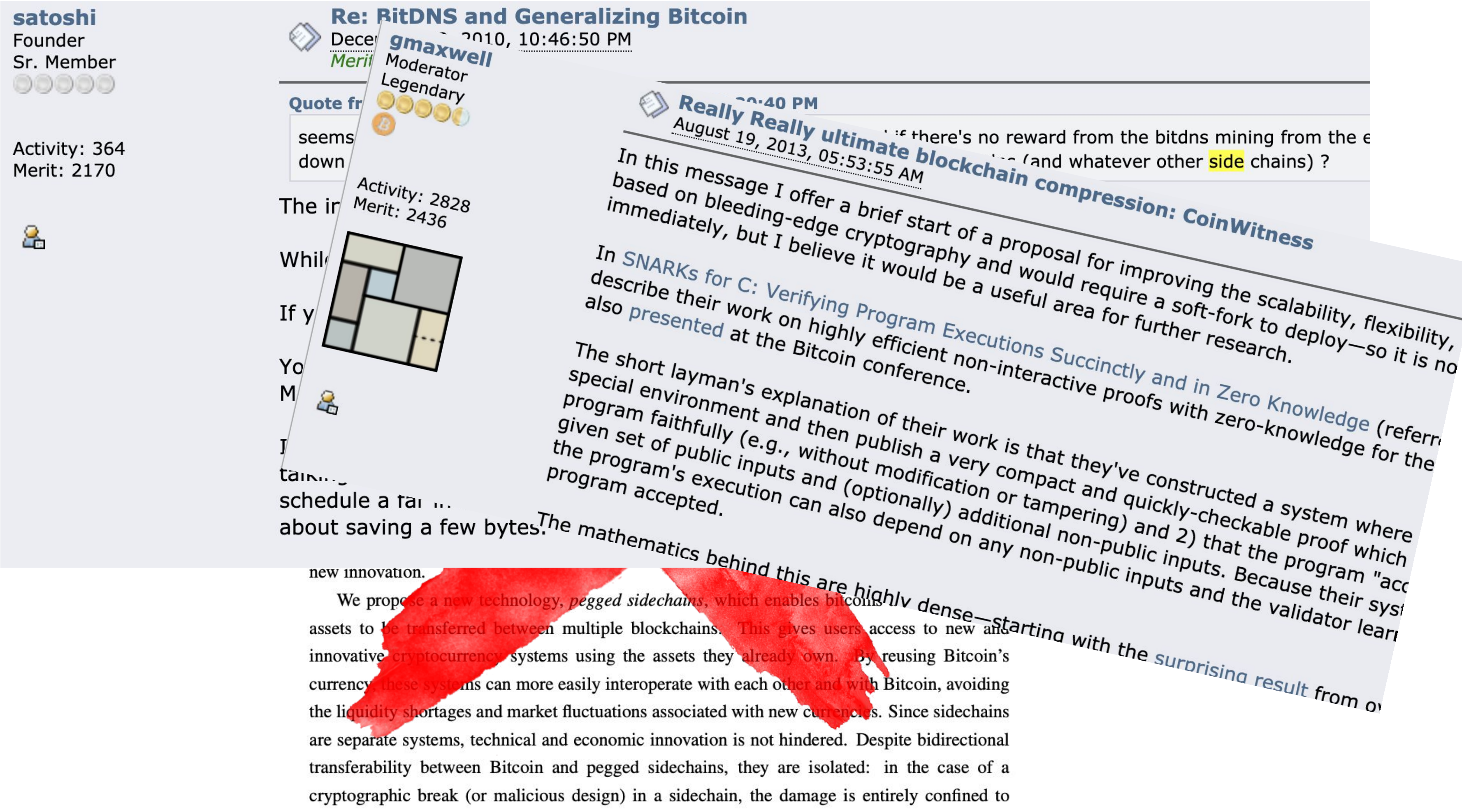
You have one piece of work. If you solve it, it will solve a block from both Bitcoin and BitDNS. In conce
Merkle Tree. To hand it in to Bitcoin, you break off the BitDNS branch, and to hand it in to BitDNS, you

In practice, to retrofit it for Bitcoin, the BitDNS side would have to have maybe ~200 extra bytes, but th
talking about 50 domains per block, which would dwarf that little 200 bytes per block for backward comp
schedule a far in future block when Bitcoin would upgrade to a modernised arrangement with the Merkle
about saving a few bytes.



new innovation.

We propose a new technology, *pegged sidechains*, which enables bitcoins and other ledger assets to be transferred between multiple blockchains. This gives users access to new and innovative cryptocurrency systems using the assets they already own. By reusing Bitcoin's currency, these systems can more easily interoperate with each other and with Bitcoin, avoiding the liquidity shortages and market fluctuations associated with new currencies. Since sidechains are separate systems, technical and economic innovation is not hindered. Despite bidirectional transferability between Bitcoin and pegged sidechains, they are isolated: in the case of a cryptographic break (or malicious design) in a sidechain, the damage is entirely confined to



satoshi
Founder
Sr. Member

Activity: 364
Merit: 2170

Re: BitDNS and Generalizing Bitcoin
Dece… 2010, 10:46:50 PM
Merit

Quote fr… …0:40 PM

> seems … if there's no reward from the bitdns mining from the e
> down … (and whatever other side chains) ?

The ir

Whil

If y

Yo
M

I
talki…
schedule a fai…
about saving a few bytes.

gmaxwell
Moderator
Legendary

Activity: 2828
Merit: 2436

Really Really ultimate blockchain compression: CoinWitness
August 19, 2013, 05:53:55 AM

In this message I offer a brief start of a proposal for improving the scalability, flexibility, based on bleeding-edge cryptography and would require a soft-fork to deploy—so it is no immediately, but I believe it would be a useful area for further research.

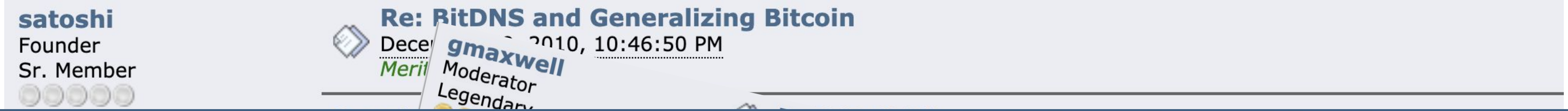
In SNARKs for C: Verifying Program Executions Succinctly and in Zero Knowledge (referr describe their work on highly efficient non-interactive proofs with zero-knowledge for the also presented at the Bitcoin conference.

The short layman's explanation of their work is that they've constructed a system where special environment and then publish a very compact and quickly-checkable proof which program faithfully (e.g., without modification or tampering) and 2) that the program "acc given set of public inputs and (optionally) additional non-public inputs. Because their syst the program's execution can also depend on any non-public inputs and the validator lear program accepted.

The mathematics behind this are highly dense—starting with the surprising result from o

new innovation.

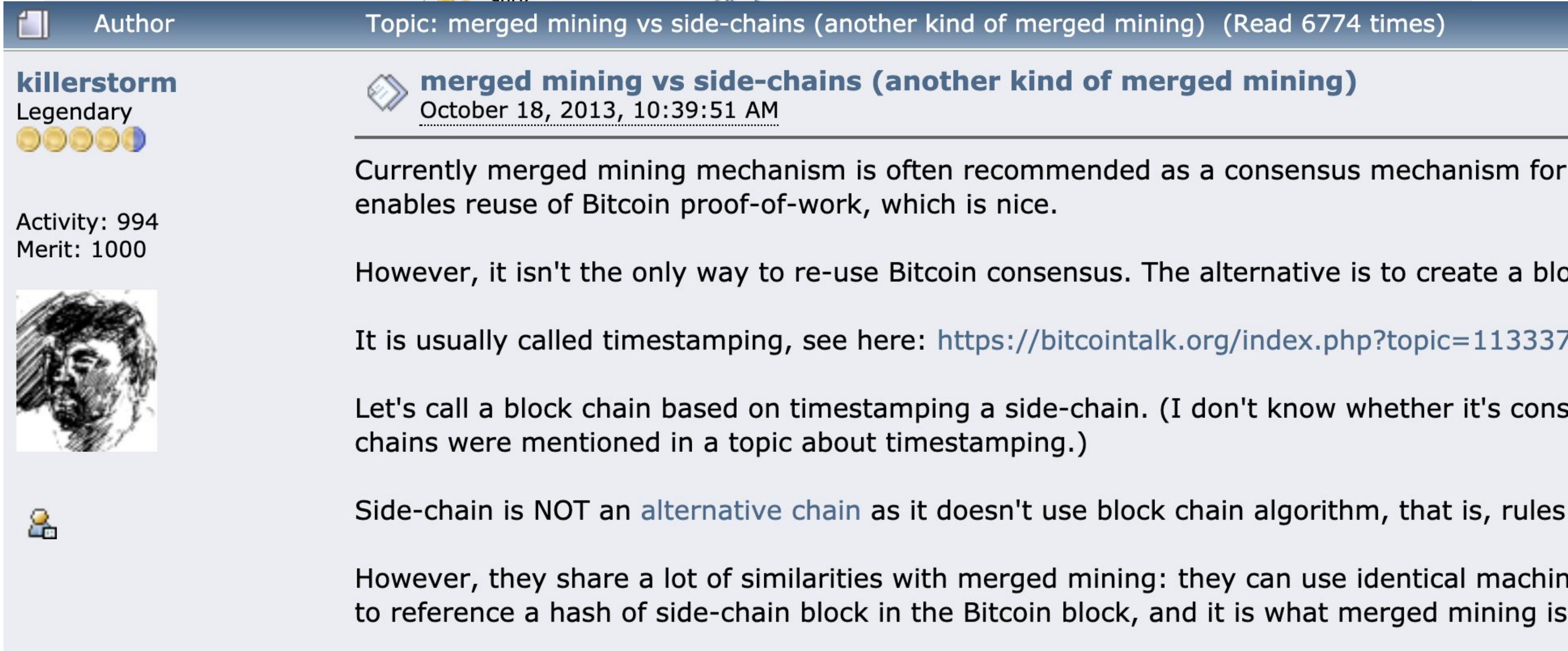
We propose a new technology, *pegged sidechains*, which enables bitcoins assets to be transferred between multiple blockchains. This gives users access to new and innovative cryptocurrency systems using the assets they already own. By reusing Bitcoin's currency, these systems can more easily interoperate with each other and with Bitcoin, avoiding the liquidity shortages and market fluctuations associated with new currencies. Since sidechains are separate systems, technical and economic innovation is not hindered. Despite bidirectional transferability between Bitcoin and pegged sidechains, they are isolated: in the case of a cryptographic break (or malicious design) in a sidechain, the damage is entirely confined to



satoshi
Founder
Sr. Member

Re: BitDNS and Generalizing Bitcoin
Dece 2010, 10:46:50 PM
Merit

gmaxwell
Moderator
Legendary



Author | Topic: merged mining vs side-chains (another kind of merged mining) (Read 6774 times)

killerstorm
Legendary

Activity: 994
Merit: 1000

merged mining vs side-chains (another kind of merged mining)
October 18, 2013, 10:39:51 AM

Currently merged mining mechanism is often recommended as a consensus mechanism for
enables reuse of Bitcoin proof-of-work, which is nice.

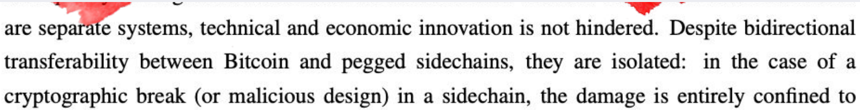
However, it isn't the only way to re-use Bitcoin consensus. The alternative is to create a blo

It is usually called timestamping, see here: https://bitcointalk.org/index.php?topic=113337

Let's call a block chain based on timestamping a side-chain. (I don't know whether it's cons
chains were mentioned in a topic about timestamping.)

Side-chain is NOT an alternative chain as it doesn't use block chain algorithm, that is, rules

However, they share a lot of similarities with merged mining: they can use identical machin
to reference a hash of side-chain block in the Bitcoin block, and it is what merged mining is



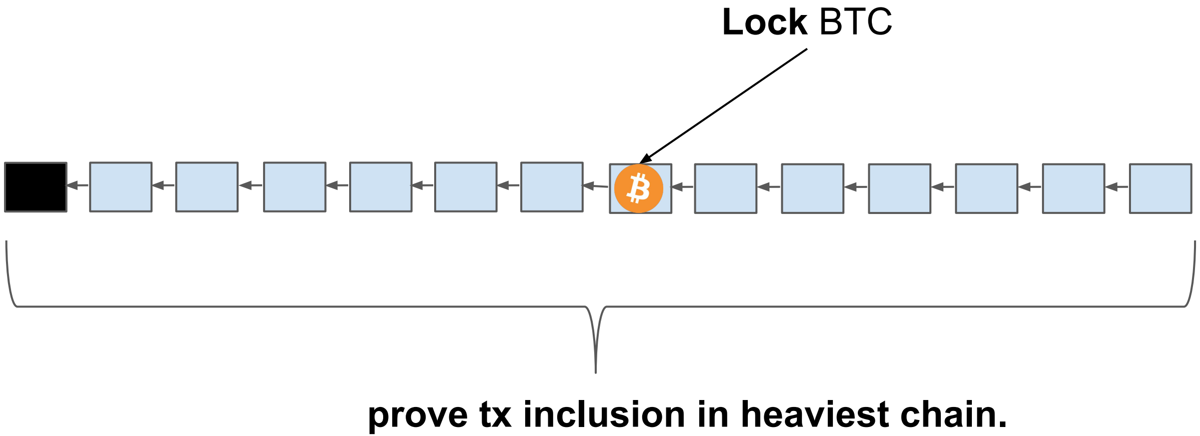
are separate systems, technical and economic innovation is not hindered. Despite bidirectional transferability between Bitcoin and pegged sidechains, they are isolated: in the case of a cryptographic break (or malicious design) in a sidechain, the damage is entirely confined to

How can a chain objectively observe another chain’s state?

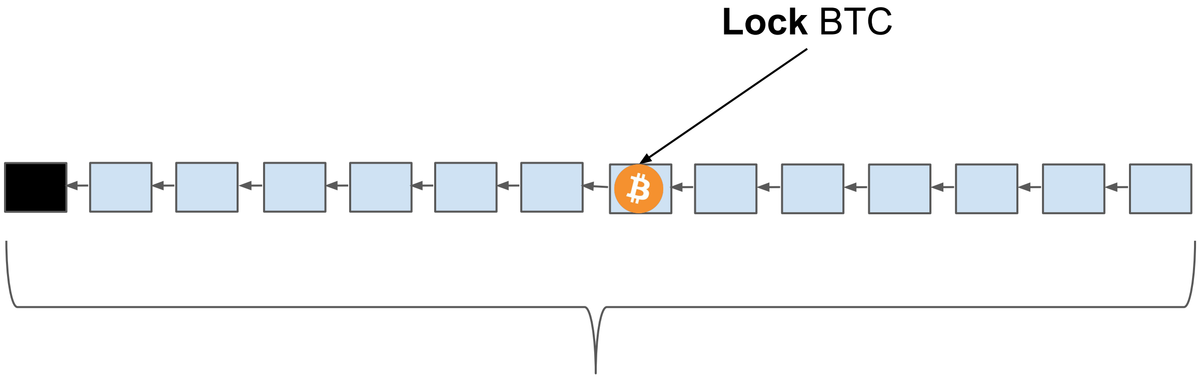
Work*!

*as long as we can verify the other chain's PoW algorithm.
Litecoin's scrypt → 20m gas on EVM 🤔

Simple Payment Verification - like a light client!



Simple Payment Verification - like a light client!

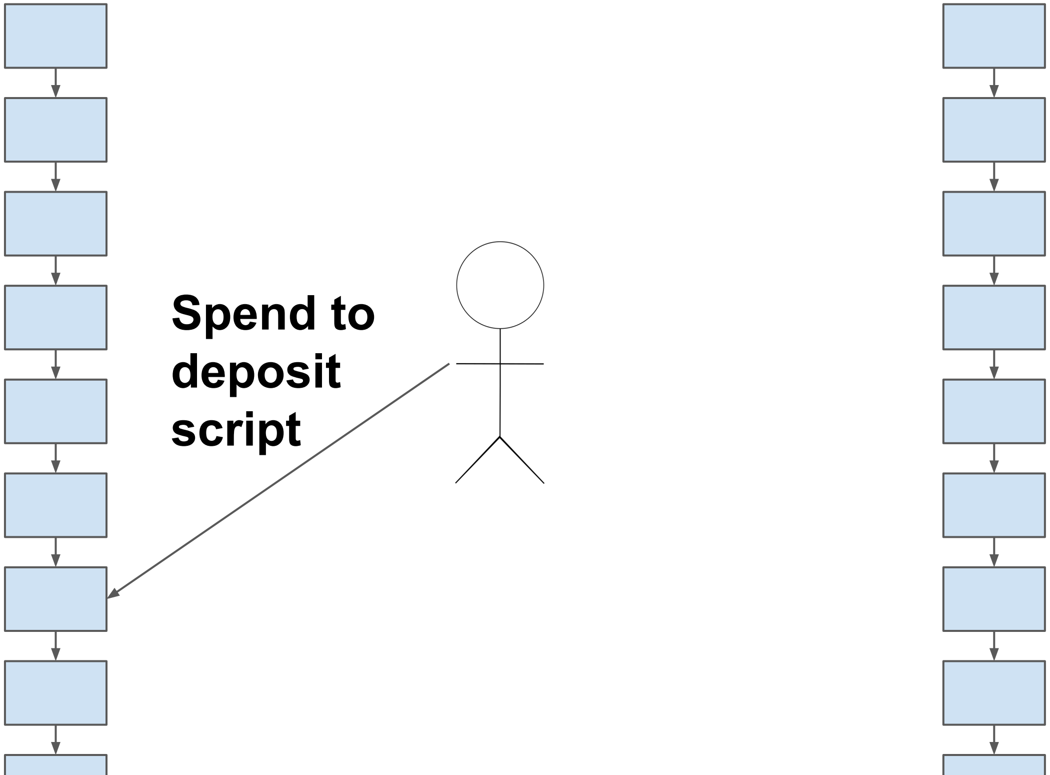


prove tx inclusion in heaviest chain

O(n), too expensive.

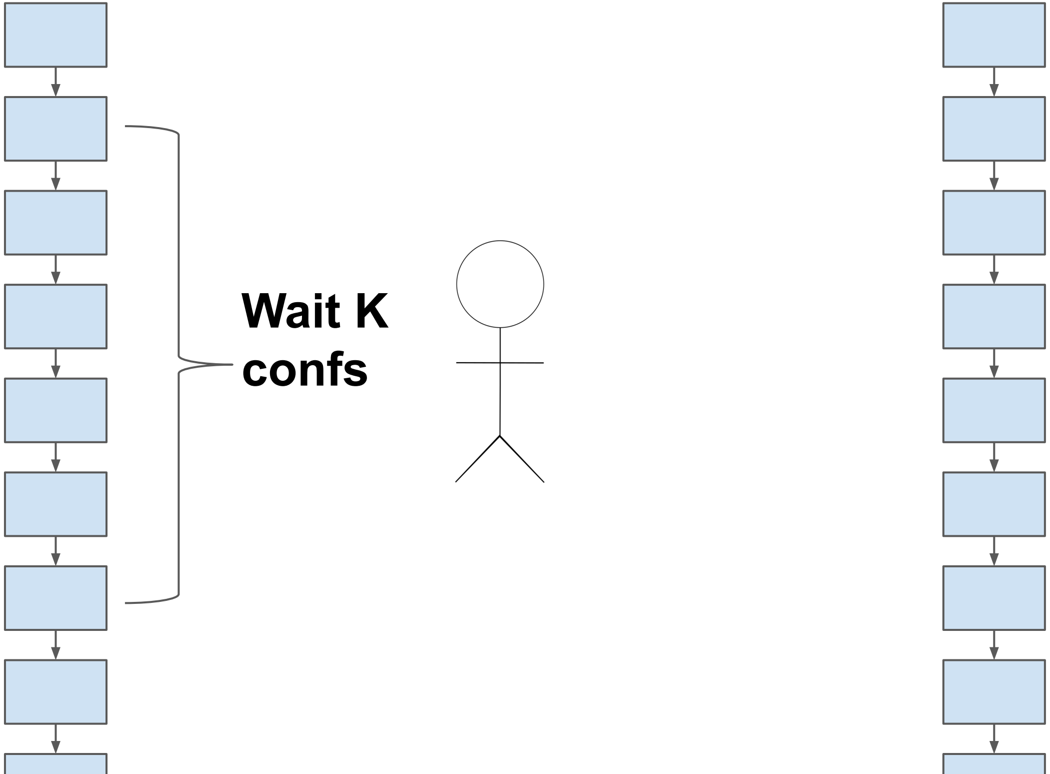
→ **NiPoPoWs / SNARKs / Stateless SPV**

PoW <> PoW sidechains: Deposit

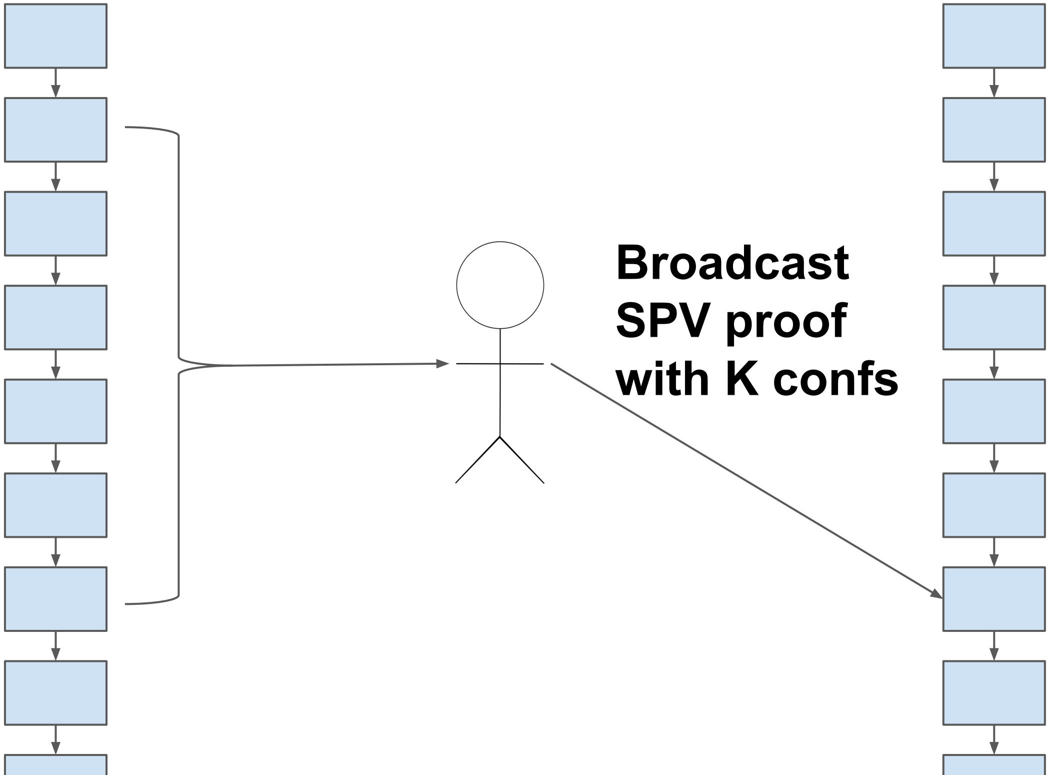


Spend to deposit script

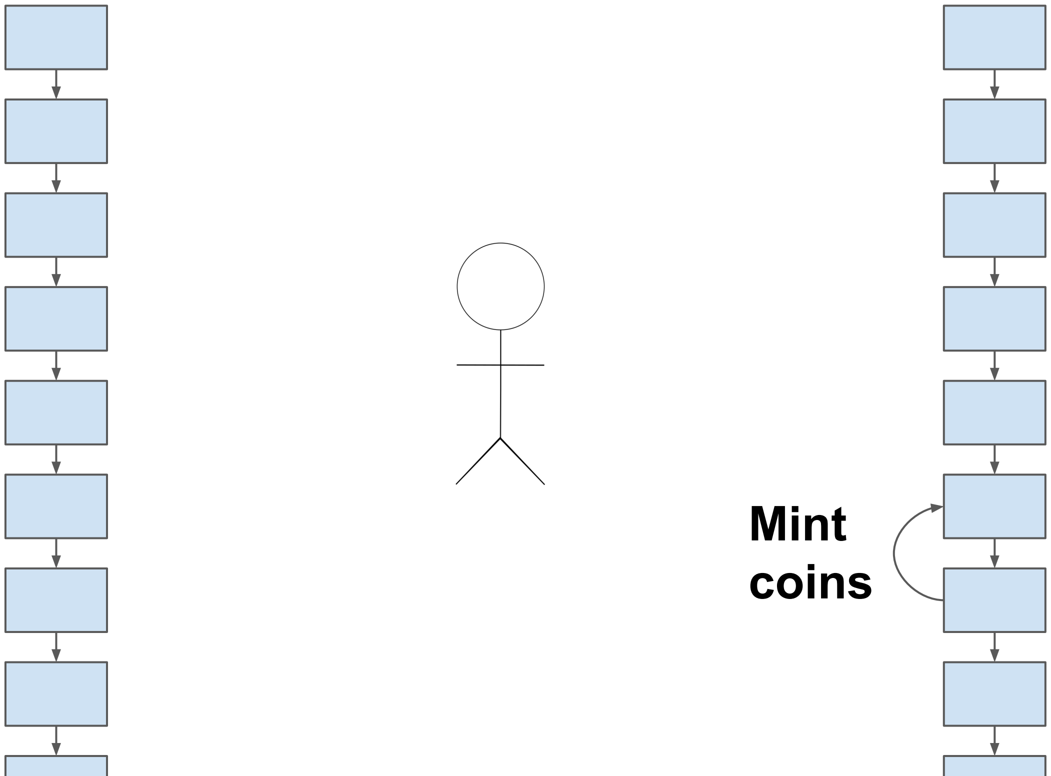
PoW <> PoW sidechains: Deposit



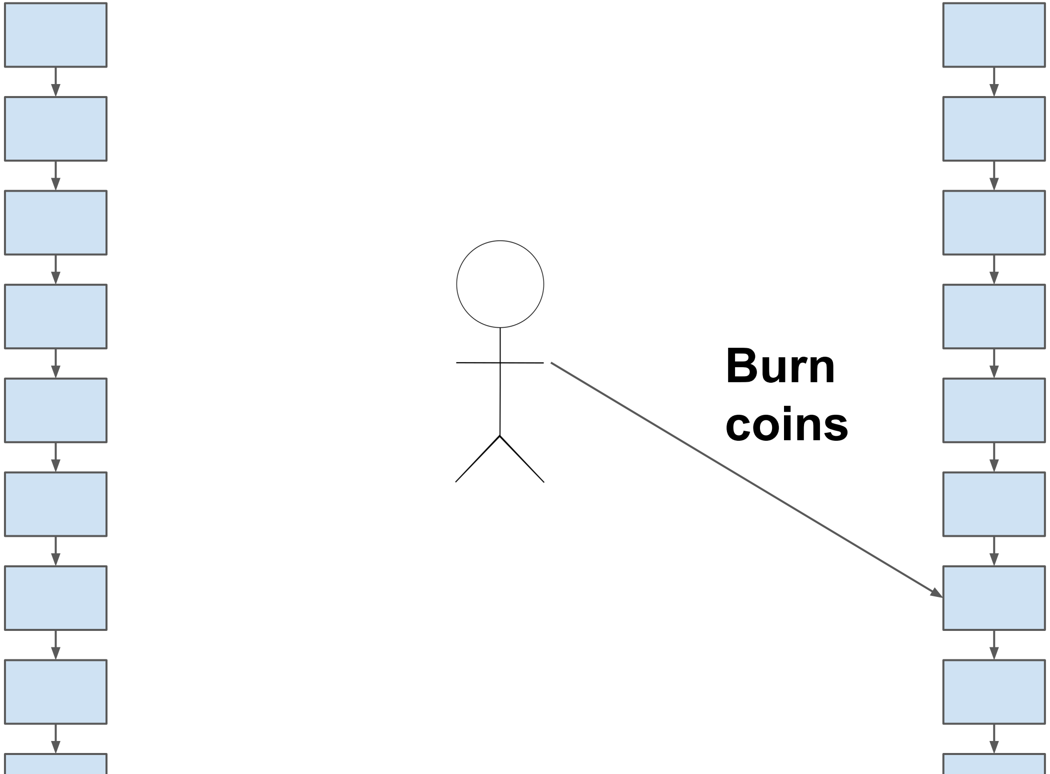
PoW <> PoW sidechains: Deposit



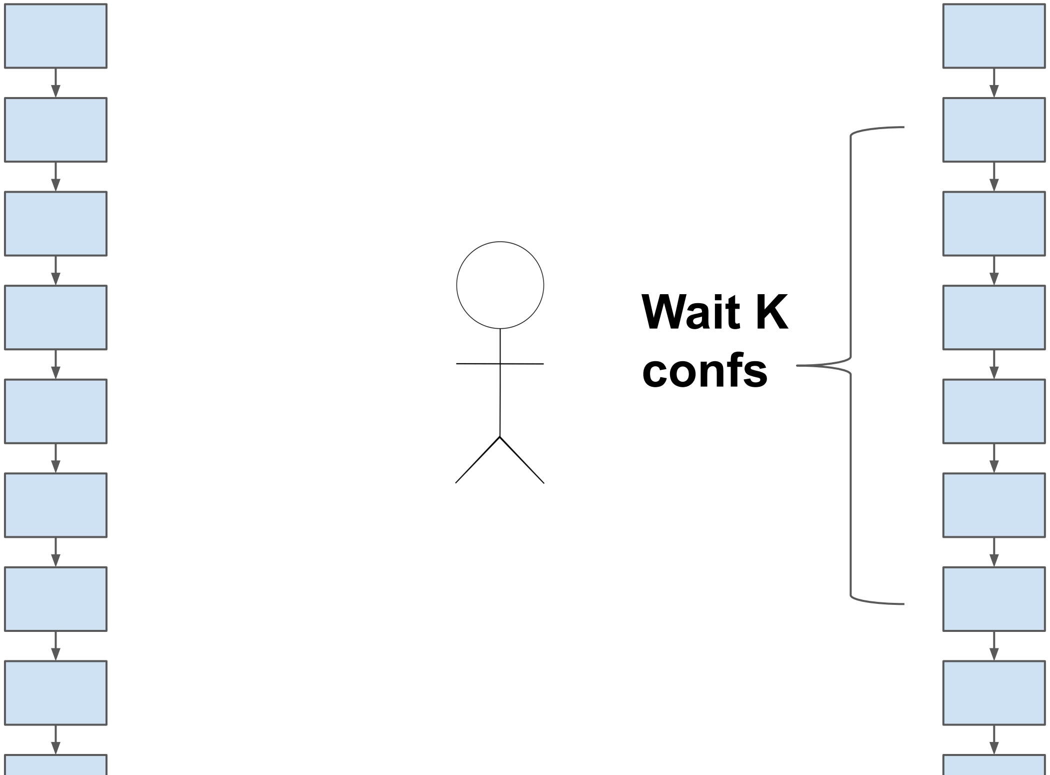
PoW <> PoW sidechains: Deposit



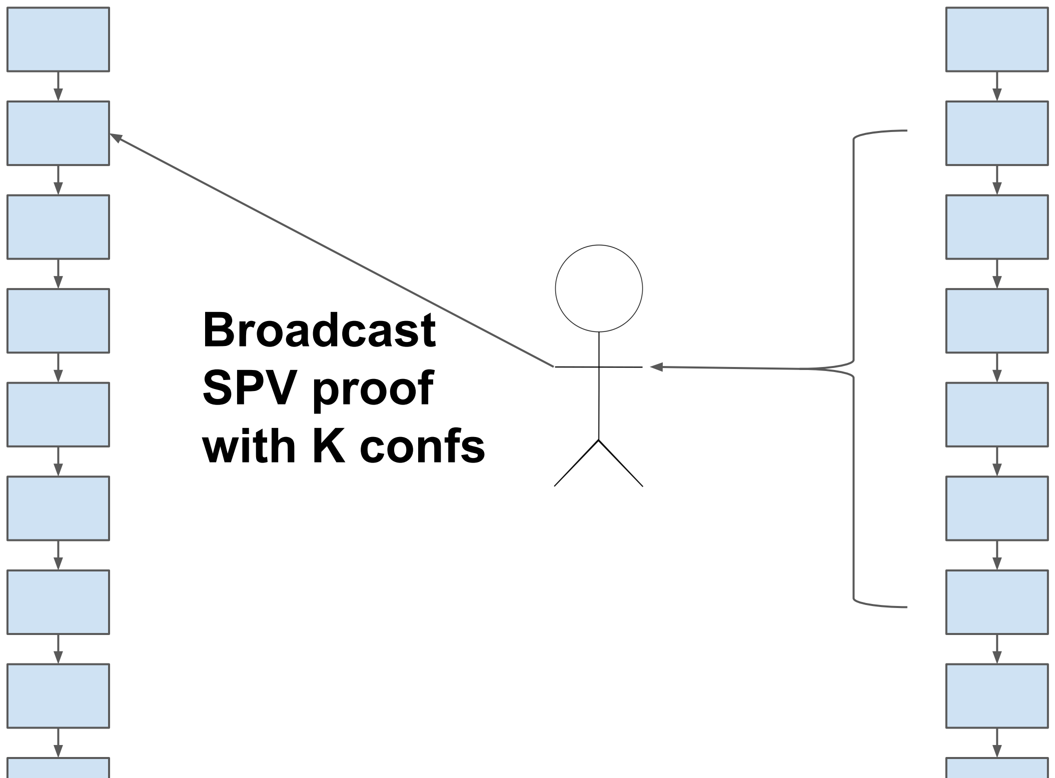
PoW <> PoW sidechains: Withdraw



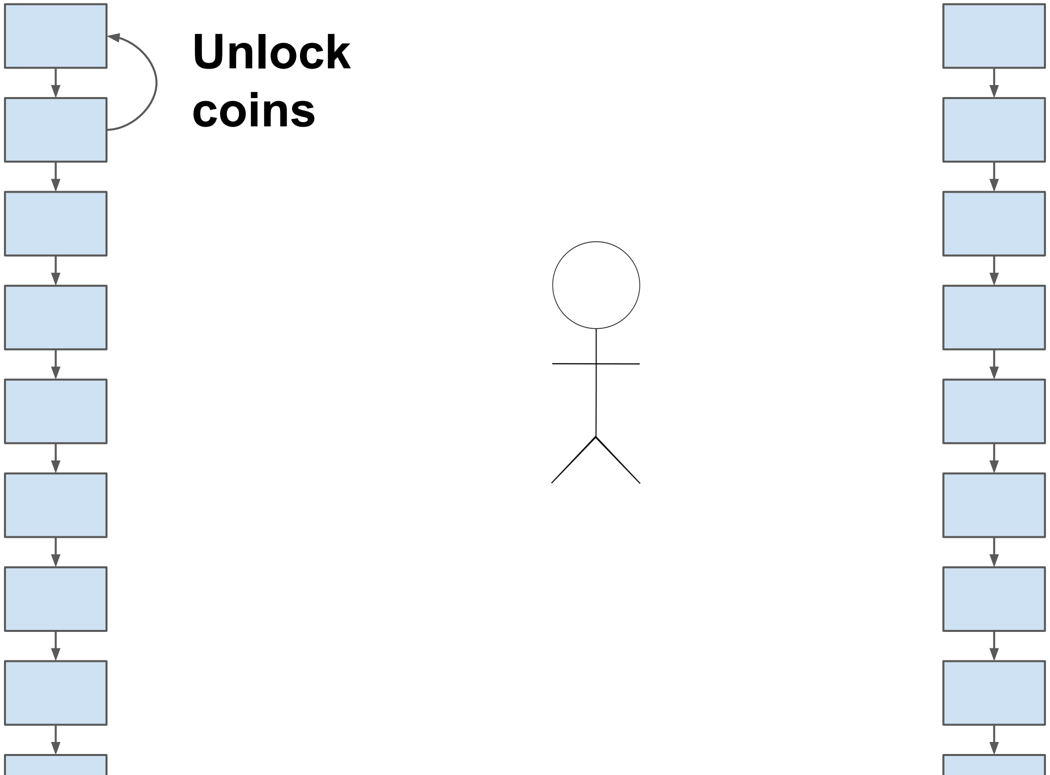
PoW <> PoW sidechains: Withdraw



PoW <> PoW sidechains: Withdraw

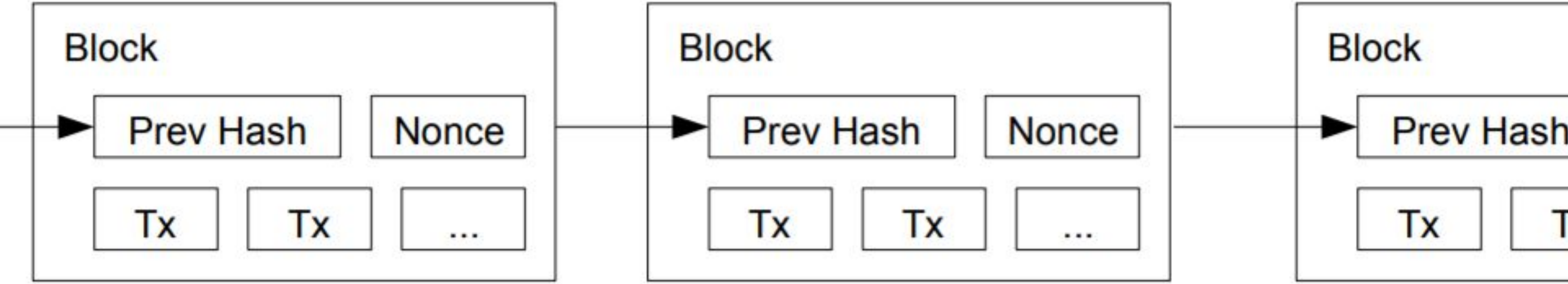


PoW <> PoW sidechains: Withdraw



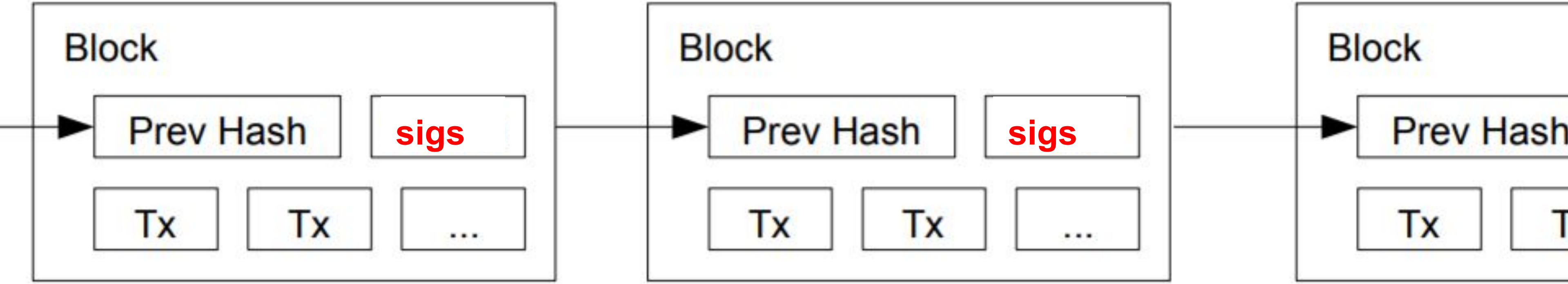
Proof of Stake sidechains?

Proof of Work block



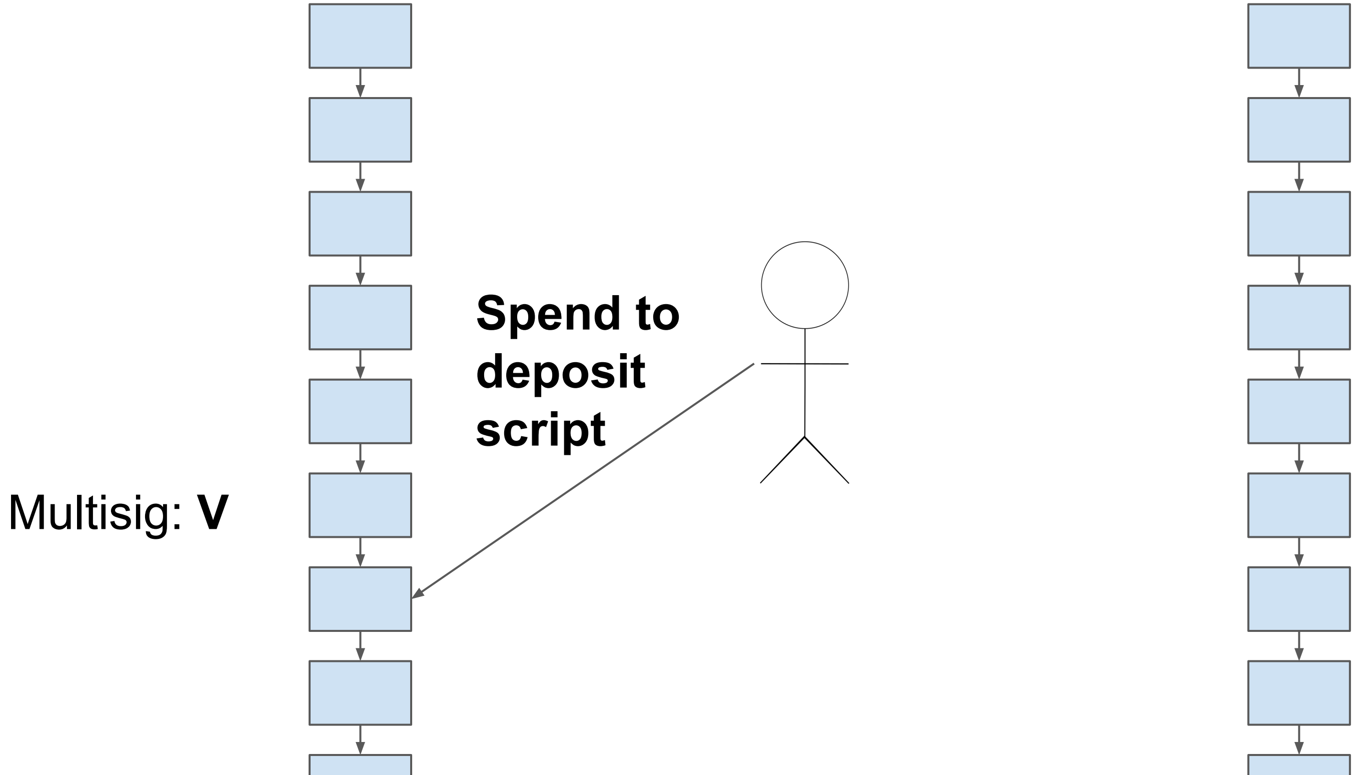
accept if h(block) < T

Proof of Stake block

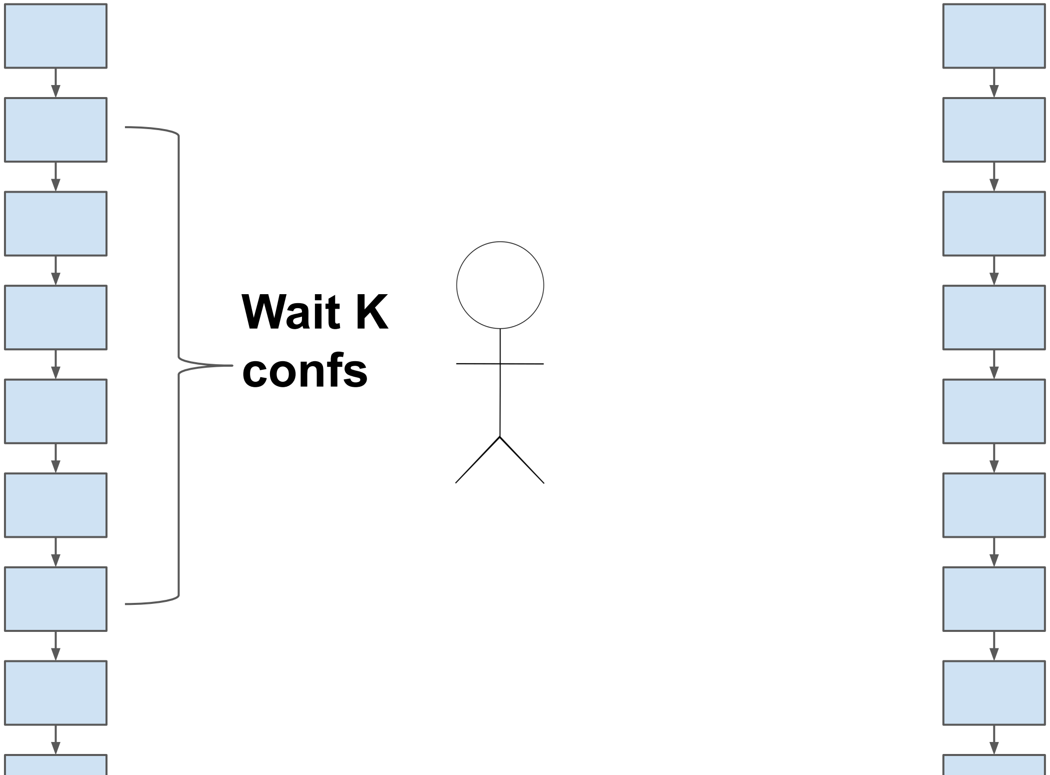


accept if stake(sigs) > ⅔ total stake

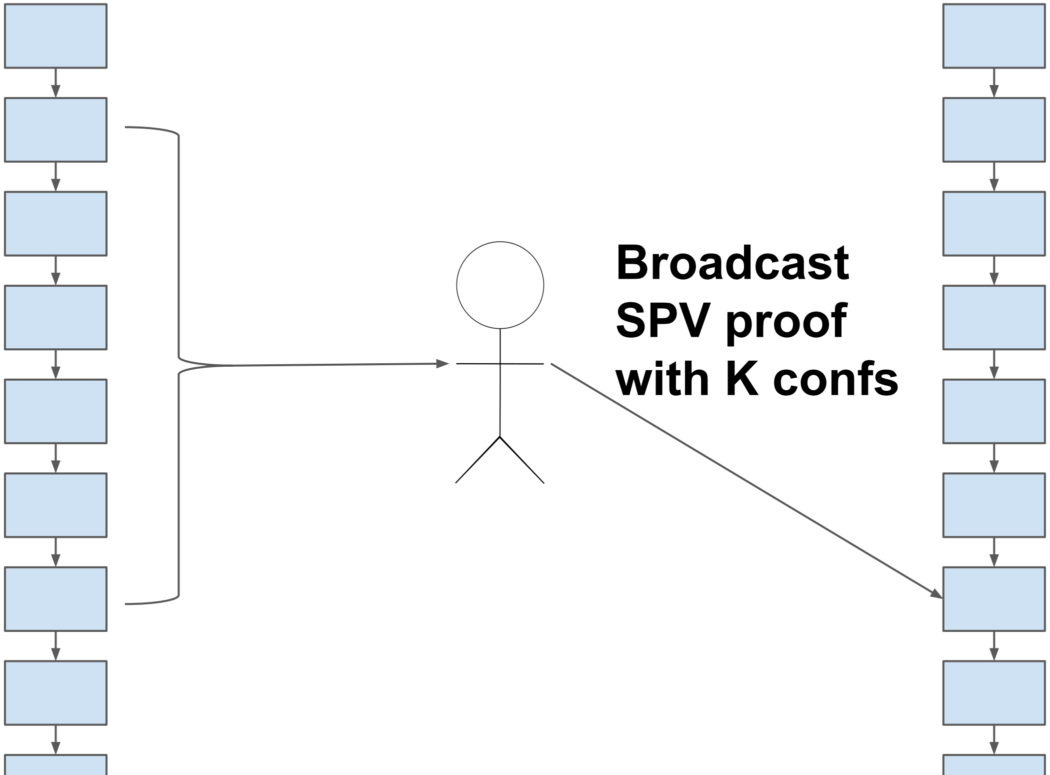
PoW <> PoS sidechains: Deposit



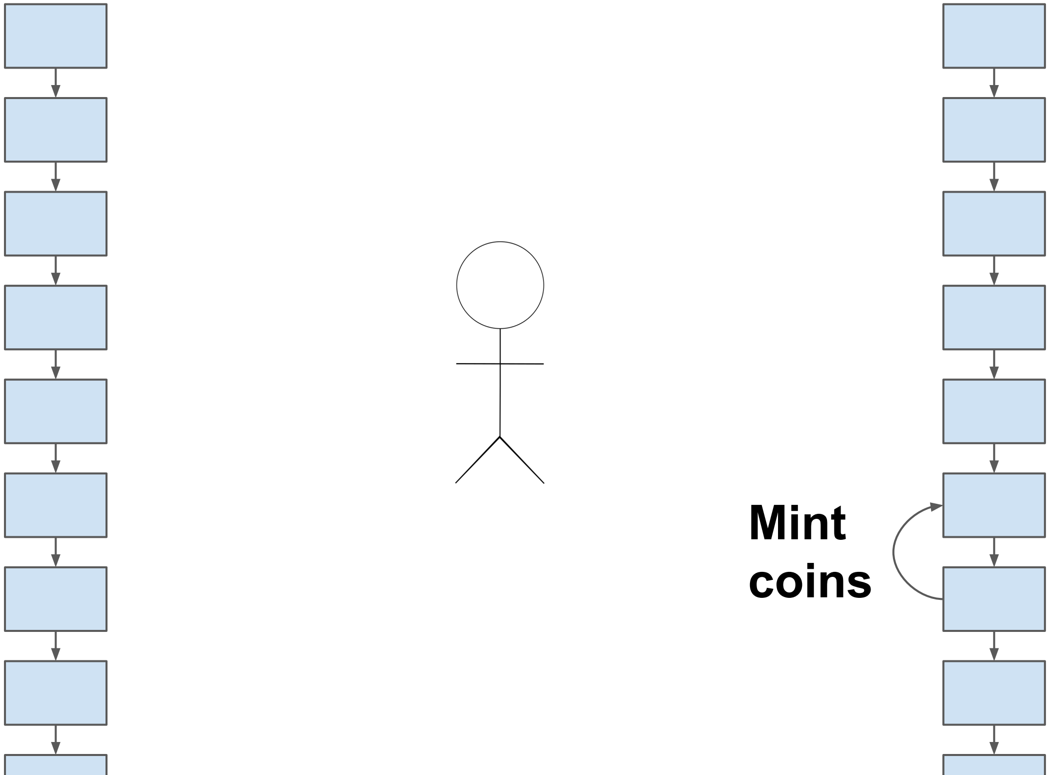
PoW <> PoS sidechains: Deposit



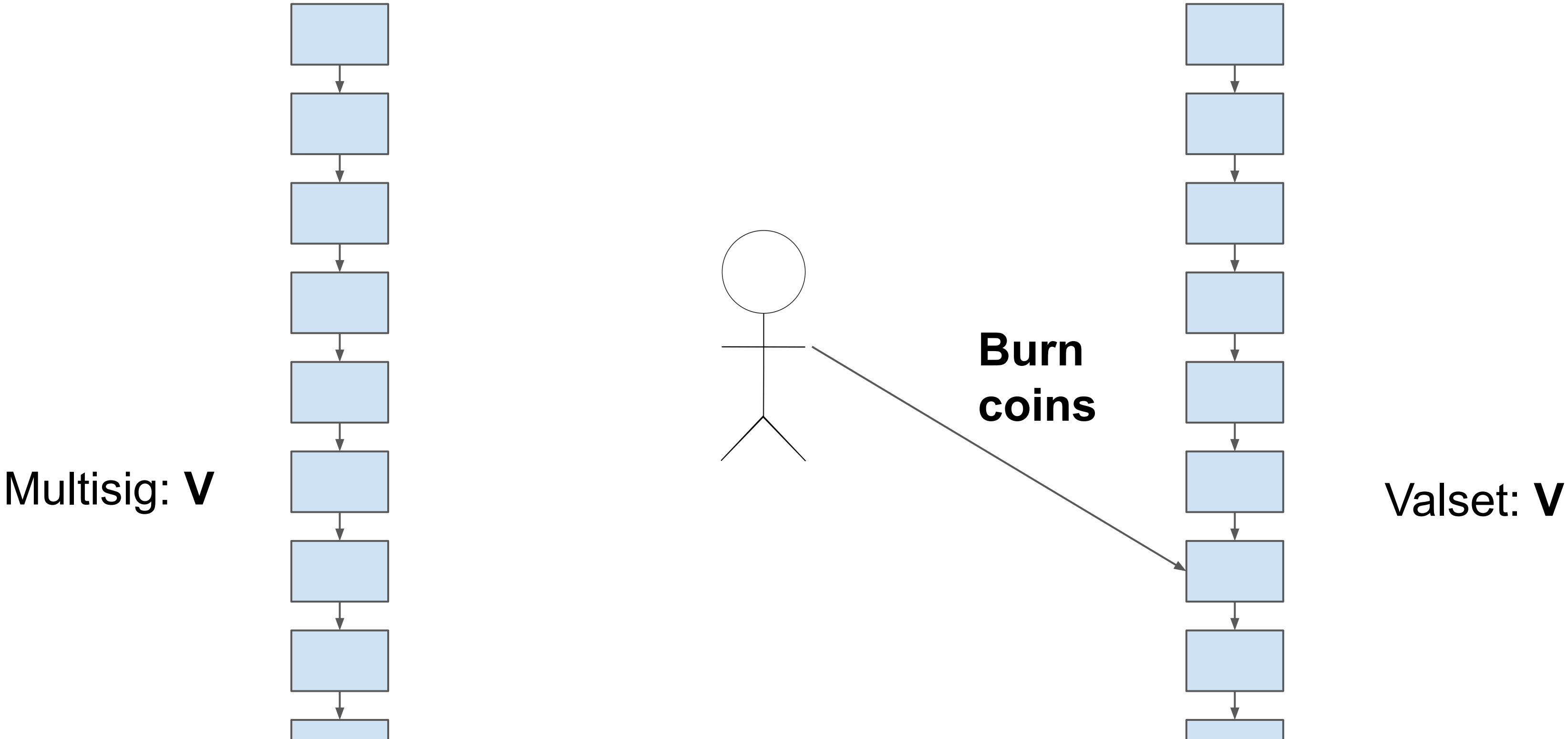
PoW <> PoS sidechains: Deposit



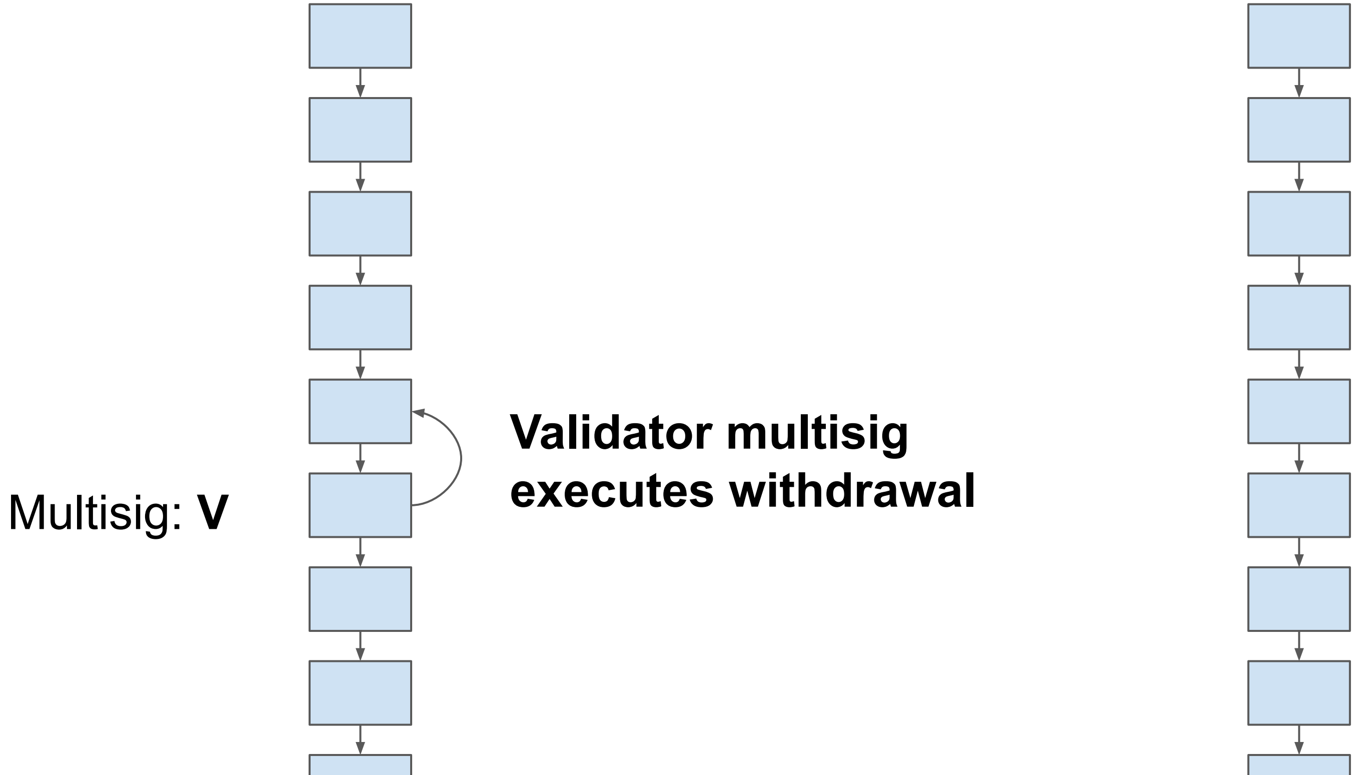
PoW <> PoS sidechains: Deposit



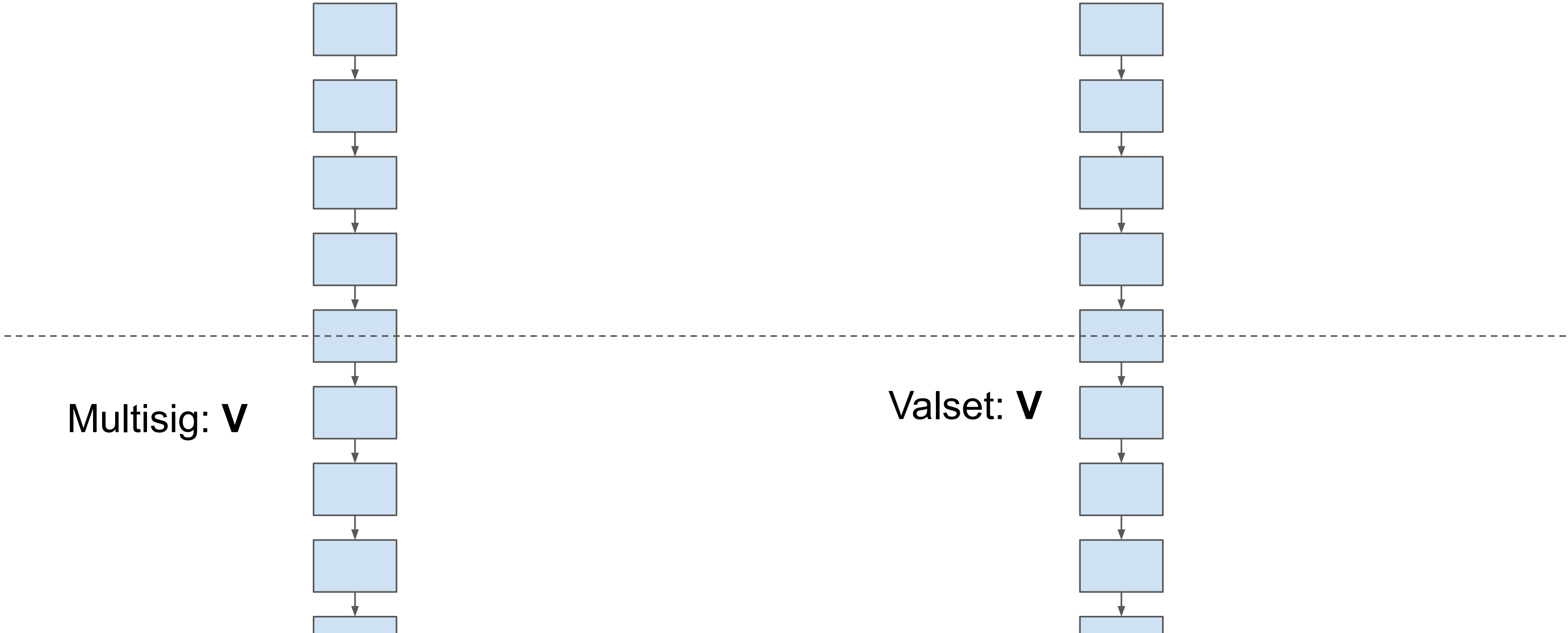
PoW <> PoS sidechains: Withdraw



PoW <> PoS sidechains: Withdraw



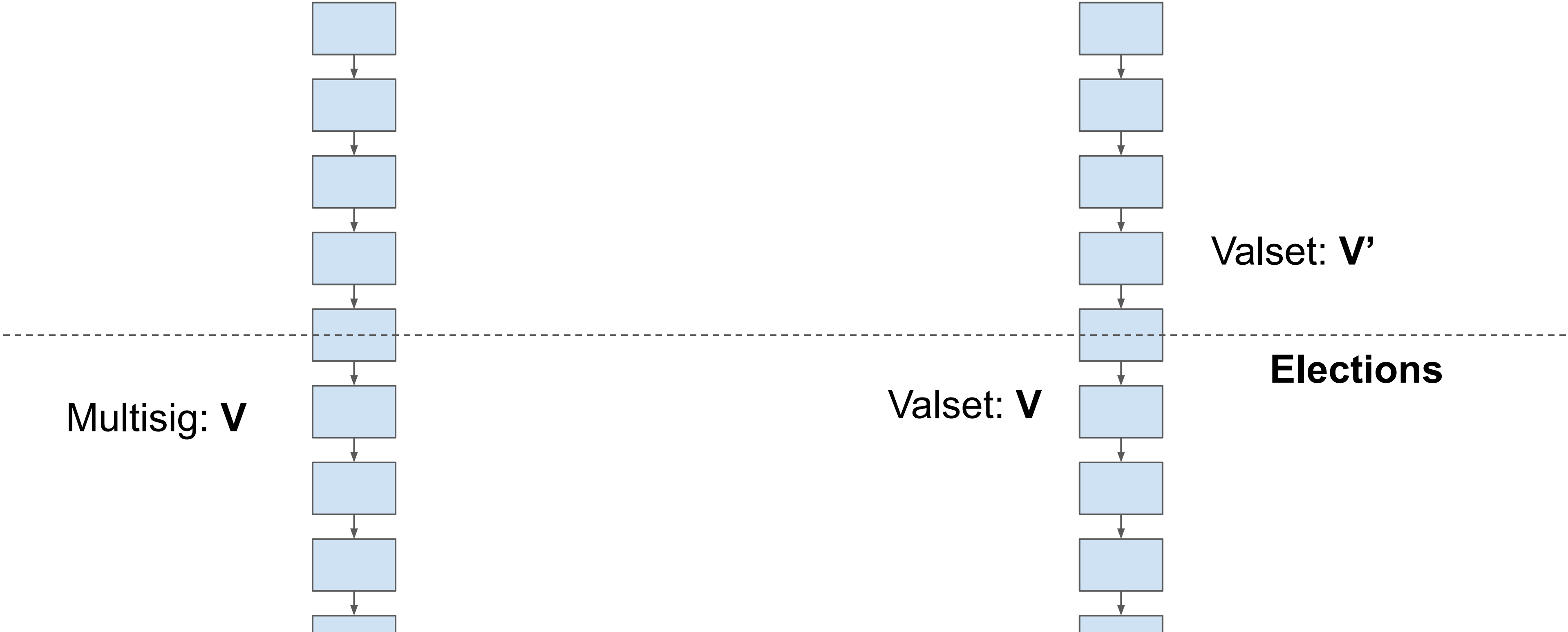
PoW <> PoS sidechains: Elections



Multisig: **V**

Valset: **V**

PoW <> PoS sidechains: Elections



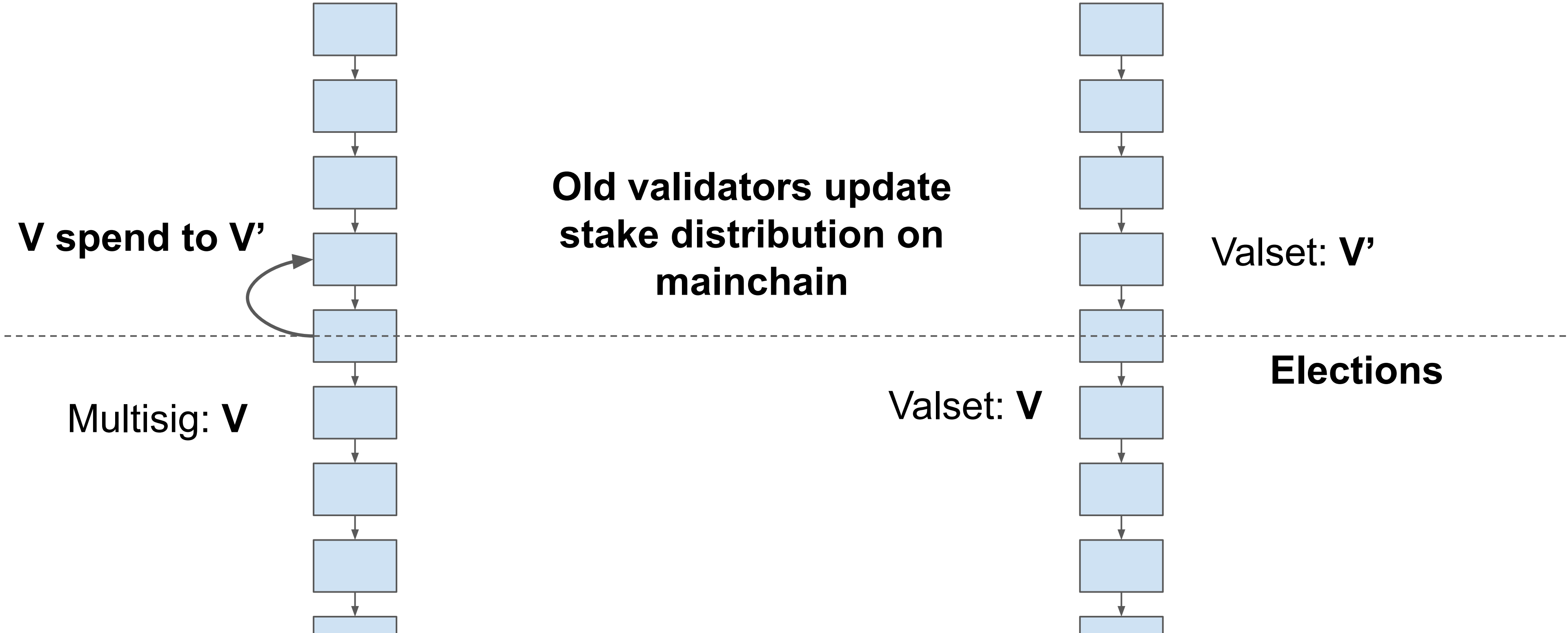
Valset: **V’**

Elections

Multisig: **V**

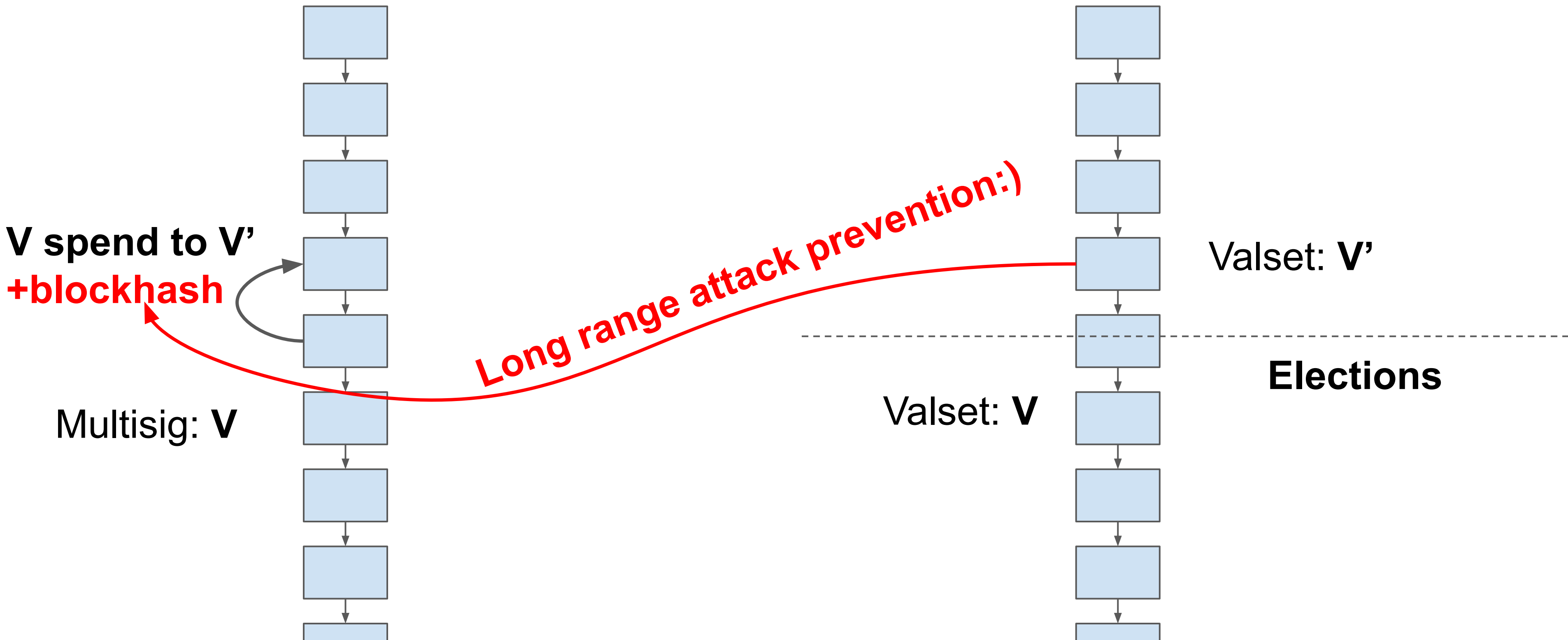
Valset: **V**

PoW <> PoS sidechains: Elections

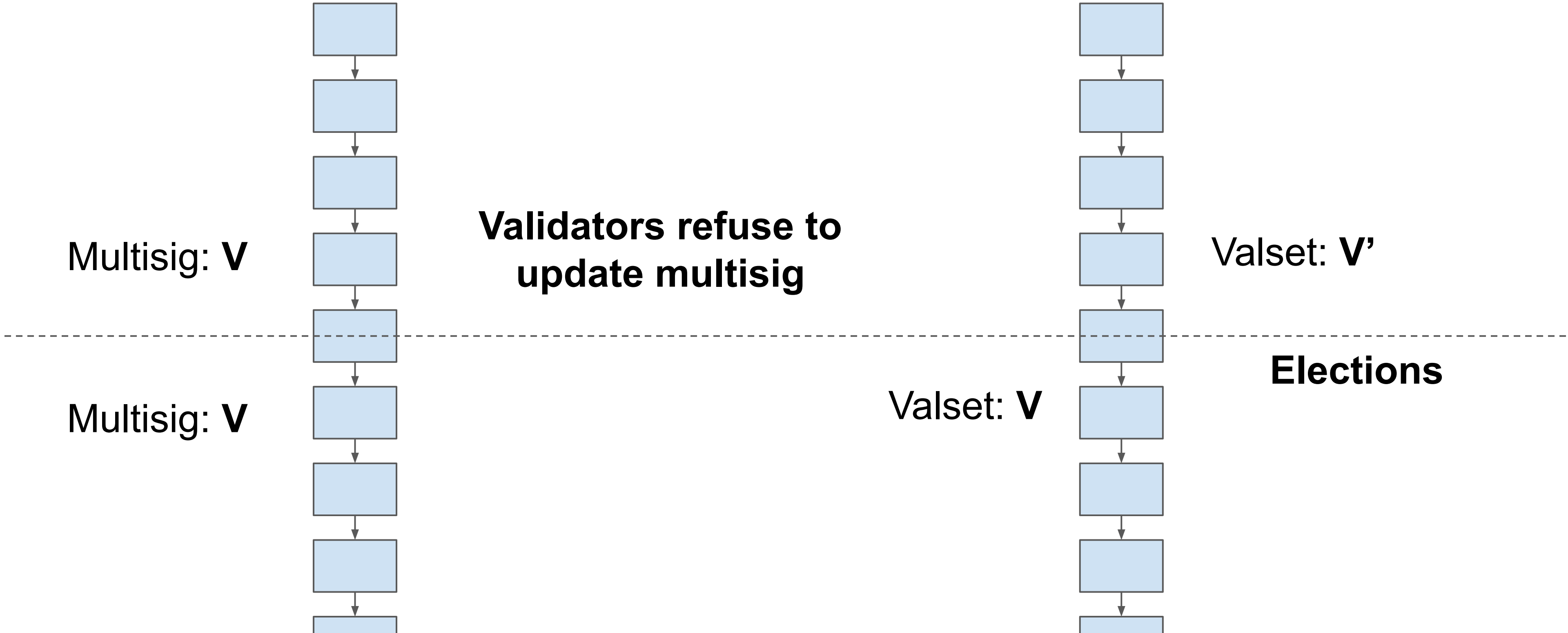


Old validators update stake distribution on mainchain

PoW <> PoS sidechains: Elections



Malicious Validators?



Multisig: **V**

Validators refuse to update multisig

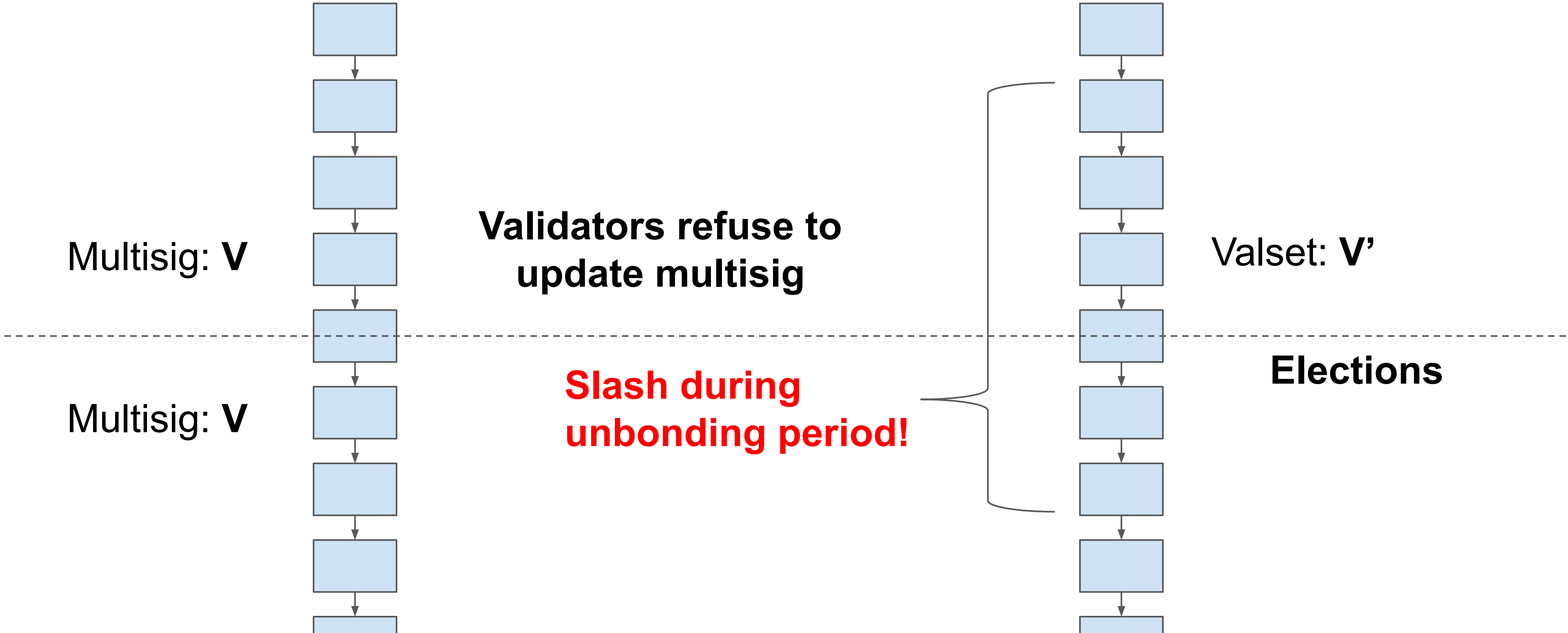
Valset: **V’**

Elections

Multisig: **V**

Valset: **V**

Malicious Validators?



Multisig: **V**

Validators refuse to update multisig

Valset: **V’**

Elections

Multisig: **V**

Slash during unbonding period!

Peg-in / Peg-out taxonomy

	Peg-in / Peg-out
Federated	Multisig
PoW Sidechain	NiPoPoWs + reorg proofs
PoS Sidechain	Rotating multisig weighted by stake + equivocation slashing (+ checkpoint to PoW chain)

“Collateral” value > BTC value for security 🤔

SoK: Communication Across Distributed Ledgers
https://eprint.iacr.org/2019/1128.pdf

(references on described technique)

https://github.com/nomic-io/bitcoin-peg/blob/master/bitcoinPeg.md
https://lists.linuxfoundation.org/pipermail/bitcoin-dev/2019-February/016642.html
https://zmnscpxj.github.io/sidechain/mainstake/index.html

Similar ideas applied to tBTC

PoS <> PoS chains

PoS <> PoS chains

???

https://eprint.iacr.org/2018/1239.pdf

Caveats:

- Rational, not byzantine adversaries
- Collateral aligns incentives,
  but is expensive

Layer 2!

L2 **safety goal**:
- honest users can withdraw their funds even if all other **non-miner** parties collude.

L2 **assumptions**:

- honest users can include a dispute transaction before a timeout
- L1: hard-to-51% attack PoW chain

L2 **primitives**:

- State machines
- Merkle Trees
- Signatures
- (Zero Knowledge Proofs)

On Bitcoin?

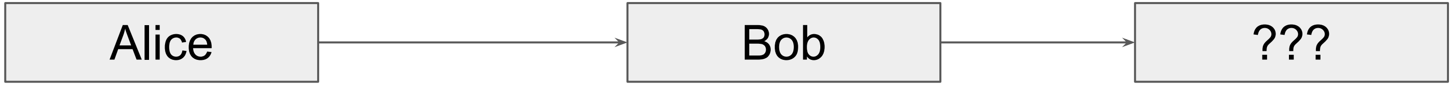
Covenants → State machines

What is a covenant?

Restriction on the outputs spending a UTXO.

O'Connor @ Bitcoin Workshop 2017:

- Digital signatures: **WHO** can spend Bitcoin
- Timelocks: **WHEN** Bitcoin can be spent

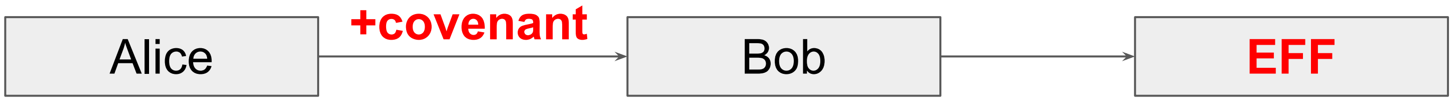


What is a covenant?

Restriction on the outputs spending a UTXO.

O'Connor @ Bitcoin Workshop 2017:

- Digital signatures: **WHO** can spend Bitcoin
- Timelocks: **WHEN** Bitcoin can be spent
- Covenants: **HOW** and **WHERE** Bitcoin can be spent



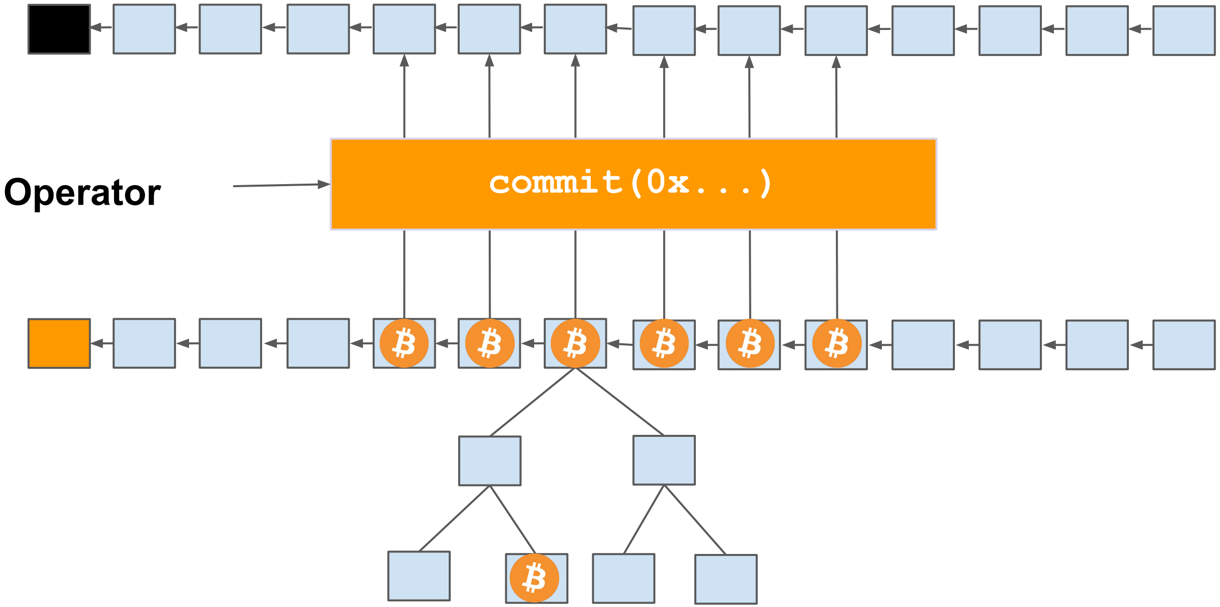
Use Cases

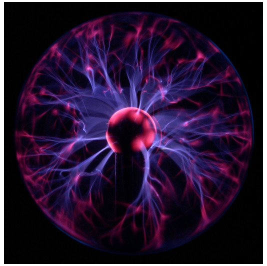
- Vaults
- Paralysis Proofs
- Colored Coins (non-fungible tokens)
- Congestion Control
- **Fraud proofs → Sidechains with trust-minimized reverse peg**
- ...more in the mailing list

Covenant Designs

- OP_CHECKOUTPUT (MES’16)
- OP_CAT + OP_CHECKSIGFROMSTACK (O’Connor, Piekarska ‘17)
- OP_CHECKOUTPUTSHASHVERIFY / OP_SECURETHEBAG (Rubin ‘19)
- OP_PUSHTXDATA (Lau ‘17)
- Presigned Transactions (..? mailing list spec)

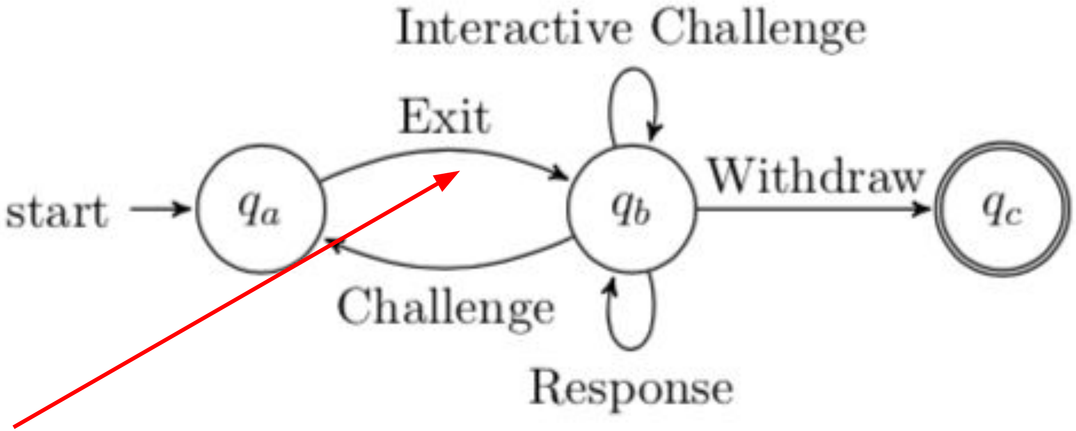
Case Study: Plasma Cash on Bitcoin





uses accumulator that supports non-membership proofs e.g. ordered merkle tree

Plasma UTXO state machine



Enforce state transitions
with a covenant!

Enforce UTXO is spent to next state

```
EnforceSpentTo(ARGS, NEXT_STATE_PATTERN):
    ARGS
    NEXT_STATE_PATTERN
    CHECKOUTPUTVERIFY
```

(use `PICK` to dynamically construct the covenant with scriptSig args)

This allows for loops which are probably unwanted in Bitcoin.
OP_SECURETHEBAG maybe?

Merkle Proof Verification

```
VerifyIncluded(UTXO_ID, ROOT, TX_HASH, PROOF):

    ROOT
    TX_HASH
    PROOF
    UTXO_ID
    MERKLEBRANCHVERIFY
```

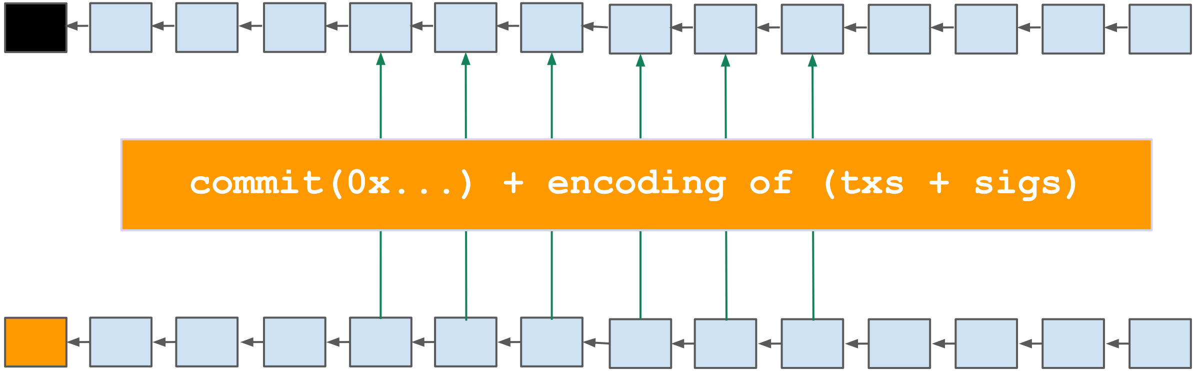
Verify block root was signed by Operator

```
VerifySignedByOperator(BLOCK_NUM, ROOT, SIG):

    BLOCK_NUM
    ROOT
    CAT
    SIG
    <OPERATOR_ADDRESS>
    CHECKSIGFROMSTACKVERIFY
```

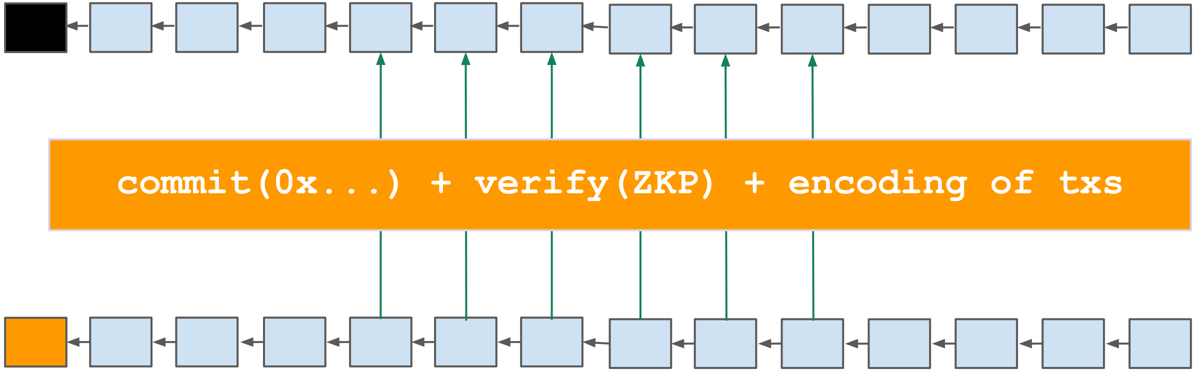
any newer schemes?

“Optimistic Rollup” - Put all the data on-chain



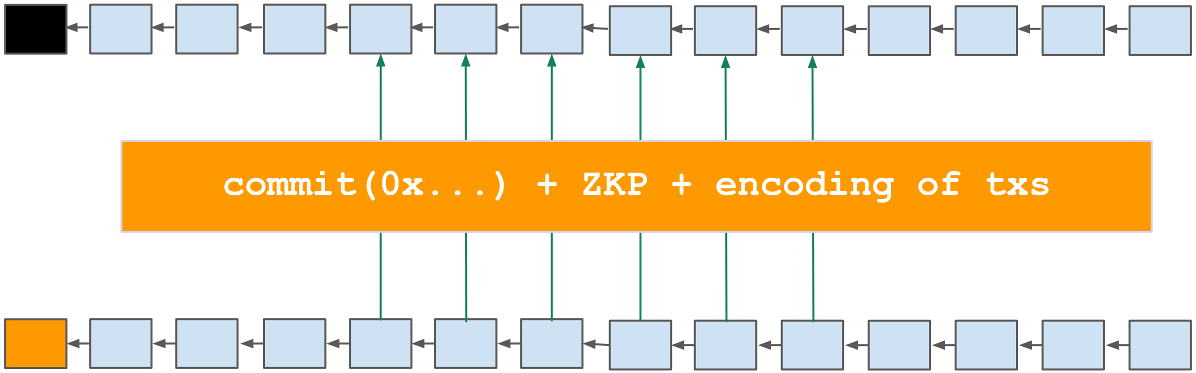
Use the Layer 1 as a data availability and dispute layer. Do not perform any computations on the txs themselves.

“ZK Rollup” - Verify ZKP & put all the data on-chain



ZKP enforces state transition correctness.
Sig verification in ZKP.

“Optimistic ZK Rollup” - Post ZKP & put all the data on-chain



Post ZKP. Verify off-chain. Verify on-chain if invalid.

Takeaways

- “Non-custodial”: either via collateral (“expensive”) or via synchrony assumption (“trust the miners”)
- State machines on Bitcoin are hard (on Ethereum too!)
- Next generation of “L2”:
    - Rollup: the L1 of L2s
    - On-chain data
    - Off-chain execution
- ZKPs w/o setup & efficient prover/verifier → HUGE.

Thank you for your attention

Q & A ?

@gakonst / me@gakonst.com

gakonst.com/cesc2019.pdf

Appendix

Security & Incentive Compatibility of Layer 2 games requirements*:

- **liveness (somebody must challenge)**
- **expected reward of attacker <=0**

*L2 games are implemented as deferred optimists: *https://medium.com/@decanus/optimistic-contracts-fb75efa7ca84*

What if the attacker is a miner?

- **violates our initial assumption**
- **Did our assumption make sense in the first place?**