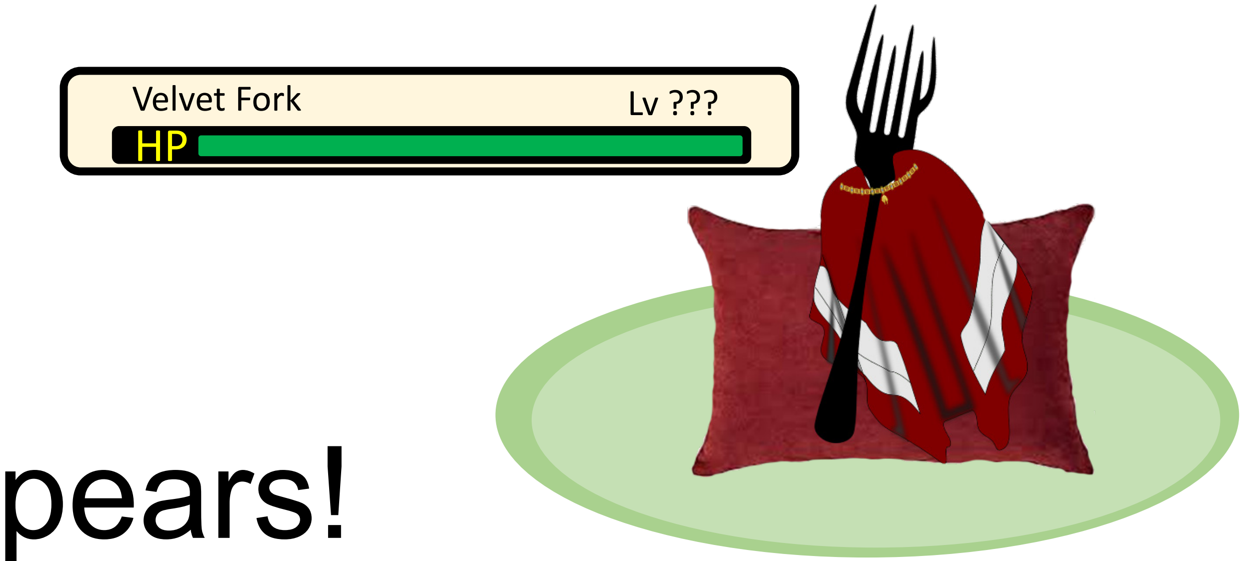
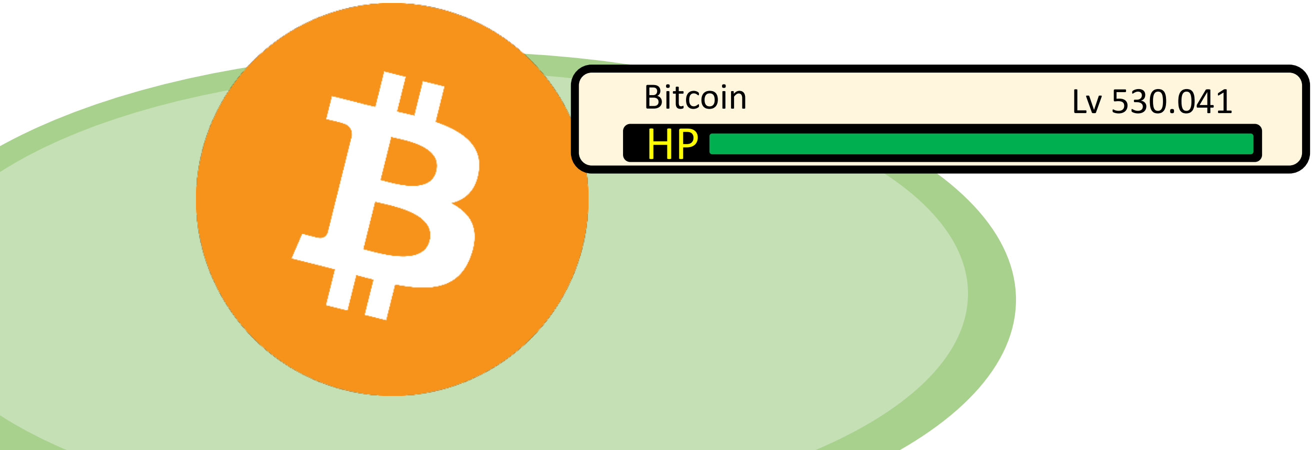
Imperial College London

SBA Research



A Wild Velvet Fork Appears!

Inclusive Blockchain Protocol Changes in Practice



Alexei Zamyatin

2F5F E92D CDAC 15B0 84A6 9FE9 9018 A958 5485 B999

Building on Bitcoin 2018

Motivation

- Bitcoin: dynamically changing set of pseudonymous participants
  - Random leader election process via *Nakamoto consensus*

- Consensus rule changes require majority vote

  →Ongoing debates on consensus changes in permissionless blockchains

- **BUT**: Soft and hard forks are not the only way to add new features!

Soft vs. Hard Forks

- Hard fork
  - Descriptor for changes incurring a **permanent split** of the blockchain

- Soft Fork
  - Some level of **compatibility** preserved towards clients adhering to previous rules

Soft vs. Hard Forks

- Hard fork
  - Descriptor for changes incurring a **permanent split** of the blockchain
  - **However:**
    No majority → No chain split (assuming econom. rational actors)
  - E.g., a failed 2Mb blocks fork: upgraded miners consider old rules valid and follow the longer „legacy" chain. New blocks continously discarded by legacy miners.
- Soft Fork
  - Some level of **compatibility** preserved towards clients adhering to previous rules
  - **However:**
    If majority of consensus participants is not upgraded → Permanent split

Notation

- Pre-agreed set of protocol rules $\boldsymbol{P}$

- Validity set $(\boldsymbol{V})$
  - Set of all blocks valid under rules $\boldsymbol{P}$
  - Block $\boldsymbol{b}$ is valid under $\boldsymbol{P}$ iff $\boldsymbol{b} \in \boldsymbol{V}$

- Question: how does a protocol change $\boldsymbol{P} \rightarrow \boldsymbol{P}'$ affect consensus?
  - Changes to validity set denoted as $\boldsymbol{N}$

Mechanisms for Consensus Rule Changes

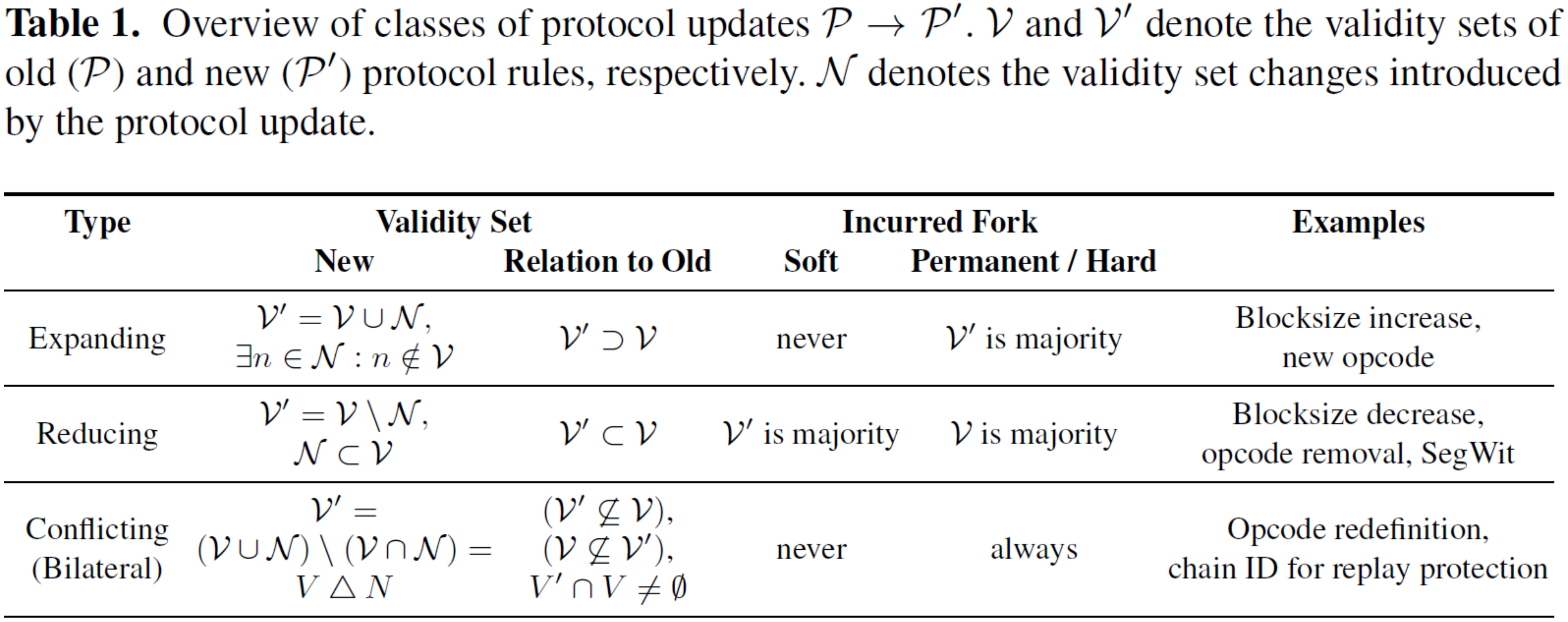


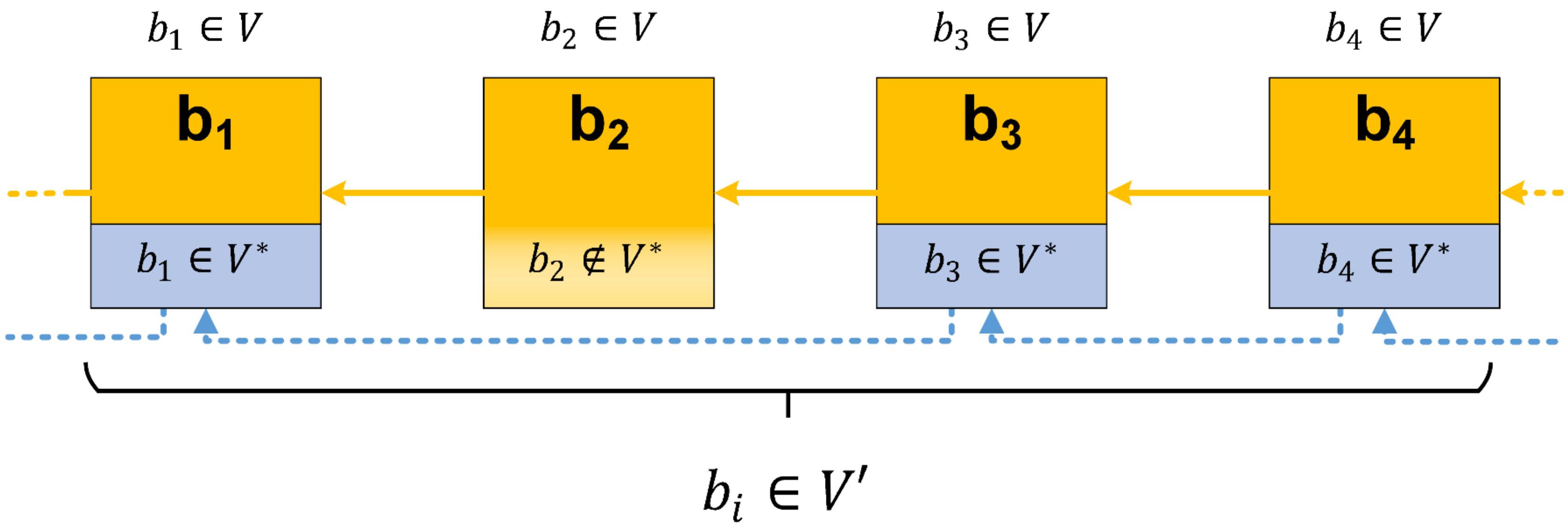
Table 1. Overview of classes of protocol updates $\mathcal{P} \rightarrow \mathcal{P}'$. $\mathcal{V}$ and $\mathcal{V}'$ denote the validity sets of old ($\mathcal{P}$) and new ($\mathcal{P}'$) protocol rules, respectively. $\mathcal{N}$ denotes the validity set changes introduced by the protocol update.

Type	**Validity Set**		**Incurred Fork**		**Examples**
	New	**Relation to Old**	**Soft**	**Permanent / Hard**	
Expanding	$\mathcal{V}' = \mathcal{V} \cup \mathcal{N}$, $\exists n \in \mathcal{N} : n \notin \mathcal{V}$	$\mathcal{V}' \supset \mathcal{V}$	never	$\mathcal{V}'$ is majority	Blocksize increase, new opcode
Reducing	$\mathcal{V}' = \mathcal{V} \setminus \mathcal{N}$, $\mathcal{N} \subset \mathcal{V}$	$\mathcal{V}' \subset \mathcal{V}$	$\mathcal{V}'$ is majority	$\mathcal{V}$ is majority	Blocksize decrease, opcode removal, SegWit
Conflicting (Bilateral)	$\mathcal{V}' =$ $(\mathcal{V} \cup \mathcal{N}) \setminus (\mathcal{V} \cap \mathcal{N}) =$ $V \triangle N$	$(\mathcal{V}' \not\subseteq \mathcal{V})$, $(\mathcal{V} \not\subseteq \mathcal{V}')$, $V' \cap V \neq \emptyset$	never	always	Opcode redefinition, chain ID for replay protection

See:
(Short Paper) A Wild Velvet Fork Appears! Inclusive Blockchain Protocol Changes in Practice
Alexei Zamyatin, Nicholas Stifter, Aljosha Judmayer, Philipp Schindler, Edgar Weippl and William J. Knottenbelt
5th Workshop on Bitcoin and Blockchain Research, Financial Cryptography and Data Security 18

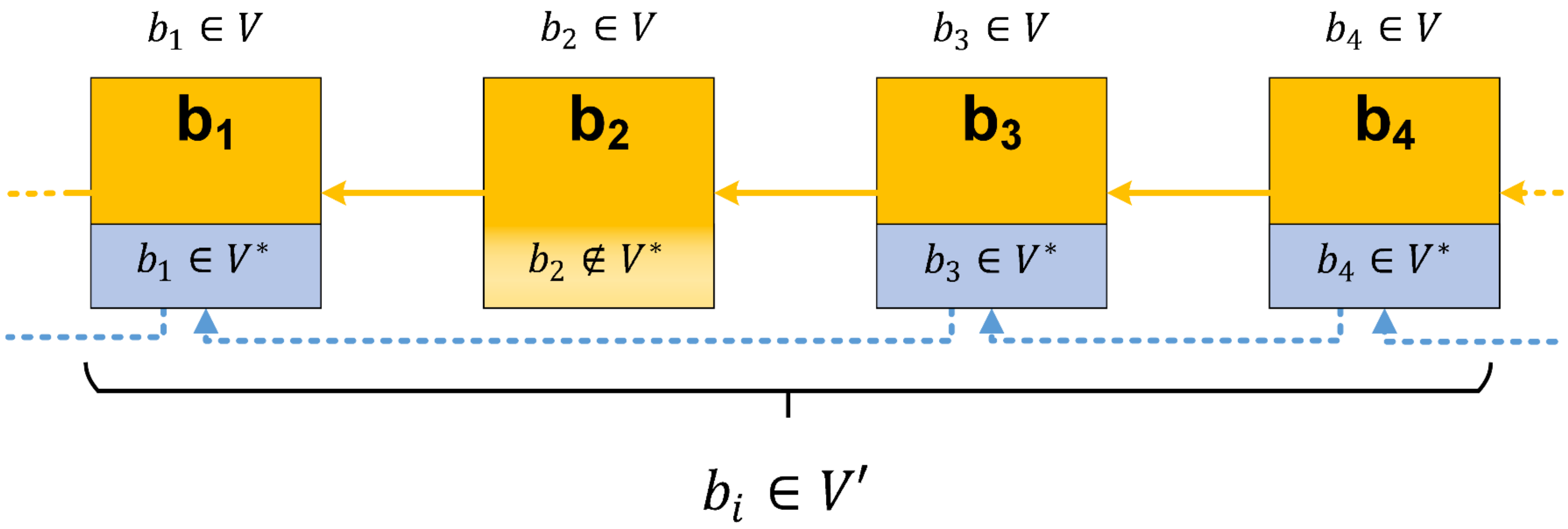
Velvet Forks

- Rules applied **conditionally**
- No majority agreement required



Velvet Forks

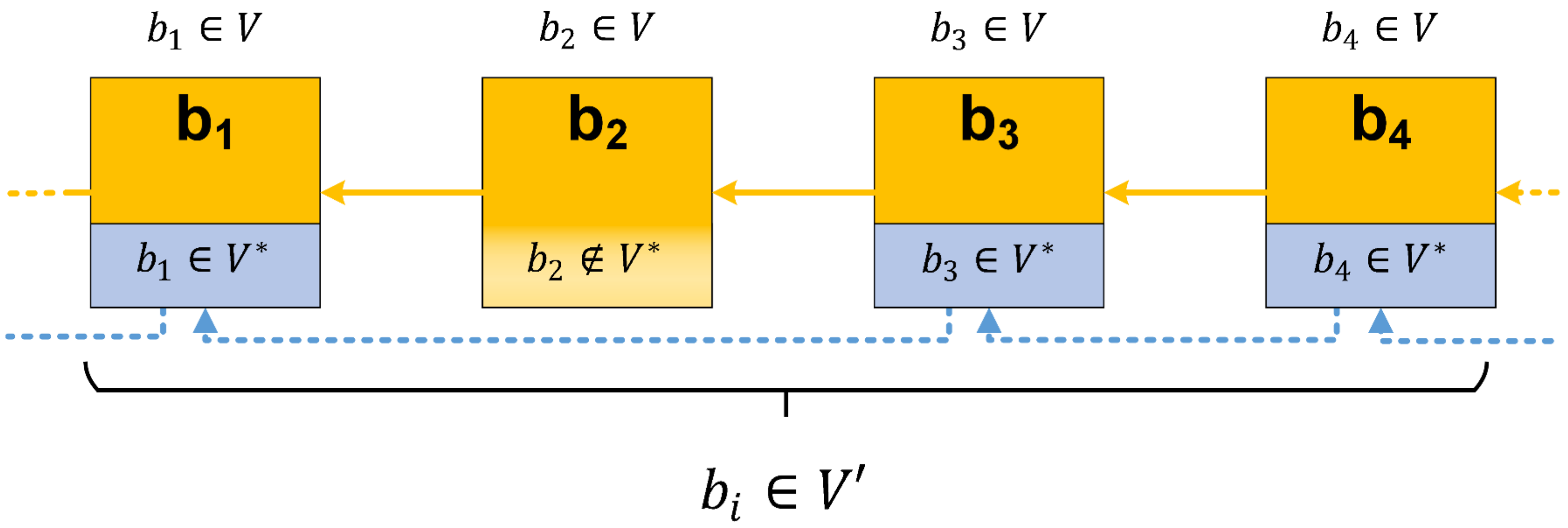
- Rules applied **conditionally**
- No majority agreement required



- Never cause a permanent chain split*

Velvet-NON-Forks

- Rules applied **conditionally**
- No majority agreement required



- Never cause a permanent chain split*

Mechanisms for Consensus Rule Changes (Cont'd)

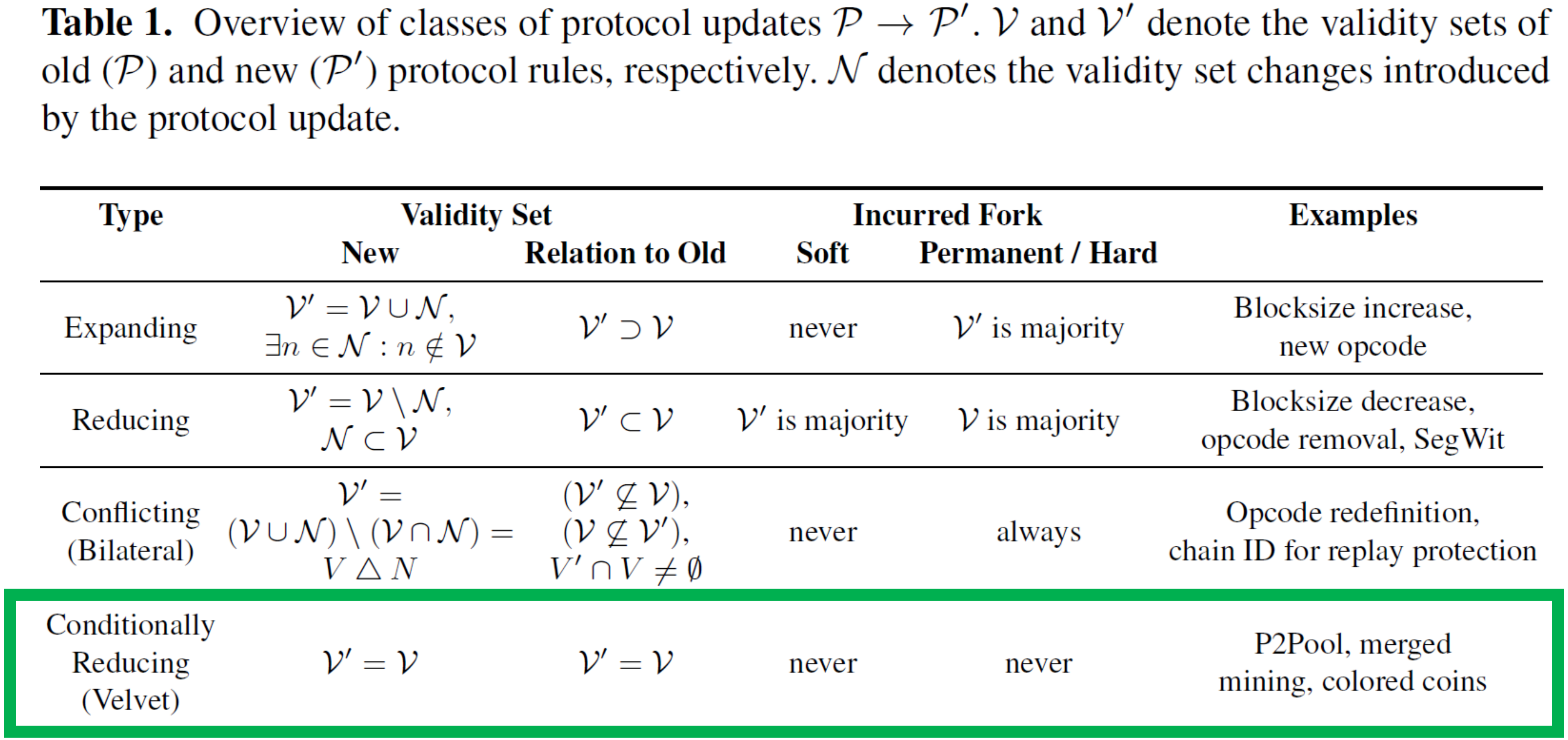


Table 1. Overview of classes of protocol updates $\mathcal{P} \rightarrow \mathcal{P}'$. $\mathcal{V}$ and $\mathcal{V}'$ denote the validity sets of old ($\mathcal{P}$) and new ($\mathcal{P}'$) protocol rules, respectively. $\mathcal{N}$ denotes the validity set changes introduced by the protocol update.

Type	**Validity Set**		**Incurred Fork**		**Examples**
	New	**Relation to Old**	**Soft**	**Permanent / Hard**	
Expanding	$\mathcal{V}' = \mathcal{V} \cup \mathcal{N}$, $\exists n \in \mathcal{N} : n \notin \mathcal{V}$	$\mathcal{V}' \supset \mathcal{V}$	never	$\mathcal{V}'$ is majority	Blocksize increase, new opcode
Reducing	$\mathcal{V}' = \mathcal{V} \setminus \mathcal{N}$, $\mathcal{N} \subset \mathcal{V}$	$\mathcal{V}' \subset \mathcal{V}$	$\mathcal{V}'$ is majority	$\mathcal{V}$ is majority	Blocksize decrease, opcode removal, SegWit
Conflicting (Bilateral)	$\mathcal{V}' =$ $(\mathcal{V} \cup \mathcal{N}) \setminus (\mathcal{V} \cap \mathcal{N}) =$ $V \triangle N$	$(\mathcal{V}' \not\subseteq \mathcal{V})$, $(\mathcal{V} \not\subseteq \mathcal{V}')$, $V' \cap V \neq \emptyset$	never	always	Opcode redefinition, chain ID for replay protection
Conditionally Reducing (Velvet)	$\mathcal{V}' = \mathcal{V}$	$\mathcal{V}' = \mathcal{V}$	never	never	P2Pool, merged mining, colored coins

Mechanisms for Consensus Rule Changes (Cont'd)

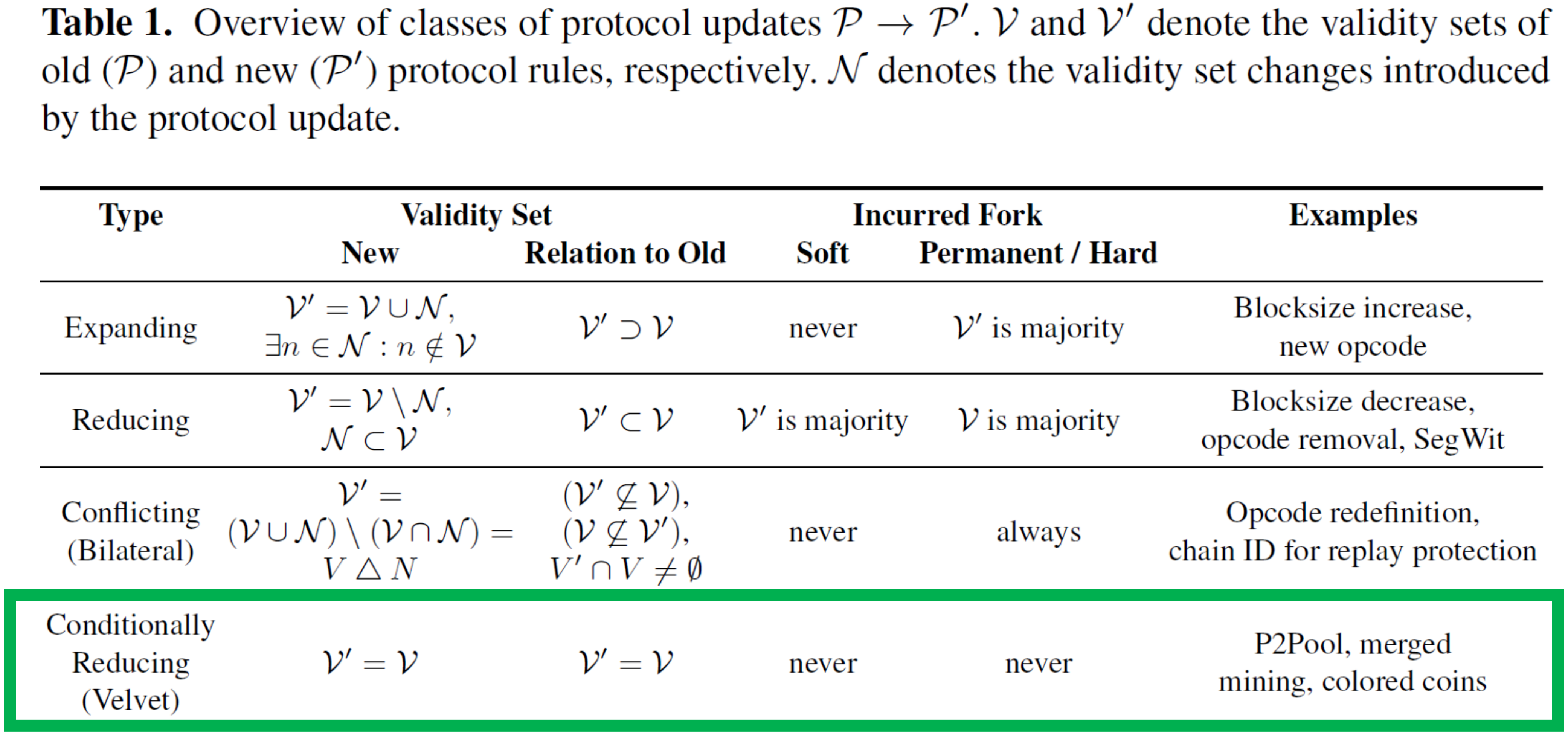


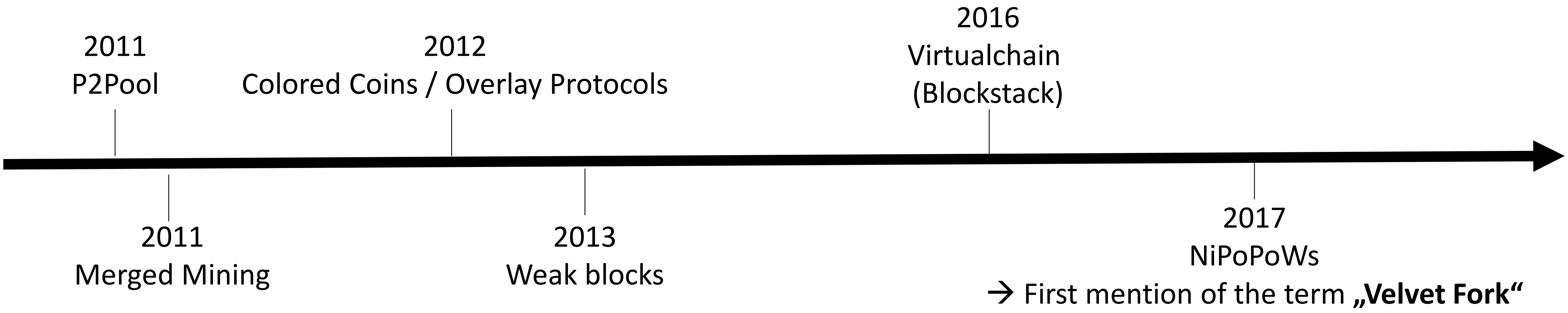
Table 1. Overview of classes of protocol updates $\mathcal{P} \rightarrow \mathcal{P}'$. $\mathcal{V}$ and $\mathcal{V}'$ denote the validity sets of old ($\mathcal{P}$) and new ($\mathcal{P}'$) protocol rules, respectively. $\mathcal{N}$ denotes the validity set changes introduced by the protocol update.

Type	Validity Set		Incurred Fork		Examples
	New	**Relation to Old**	**Soft**	**Permanent / Hard**	
Expanding	$\mathcal{V}' = \mathcal{V} \cup \mathcal{N}$, $\exists n \in \mathcal{N} : n \notin \mathcal{V}$	$\mathcal{V}' \supset \mathcal{V}$	never	$\mathcal{V}'$ is majority	Blocksize increase, new opcode
Reducing	$\mathcal{V}' = \mathcal{V} \setminus \mathcal{N}$, $\mathcal{N} \subset \mathcal{V}$	$\mathcal{V}' \subset \mathcal{V}$	$\mathcal{V}'$ is majority	$\mathcal{V}$ is majority	Blocksize decrease, opcode removal, SegWit
Conflicting (Bilateral)	$\mathcal{V}' =$ $(\mathcal{V} \cup \mathcal{N}) \setminus (\mathcal{V} \cap \mathcal{N}) =$ $V \triangle N$	$(\mathcal{V}' \not\subseteq \mathcal{V})$, $(\mathcal{V} \not\subseteq \mathcal{V}')$, $V' \cap V \neq \emptyset$	never	always	Opcode redefinition, chain ID for replay protection
Conditionally Reducing (Velvet)	$\mathcal{V}' = \mathcal{V}$	$\mathcal{V}' = \mathcal{V}$	never	never	P2Pool, merged mining, colored coins

*Except if conflicting rules are introduced by legacy miners

→ Essentially, ban adding data to transactions/blocks….

Velvet Forks in the Wild



2011
P2Pool

2011
Merged Mining

2012
Colored Coins / Overlay Protocols

2013
Weak blocks

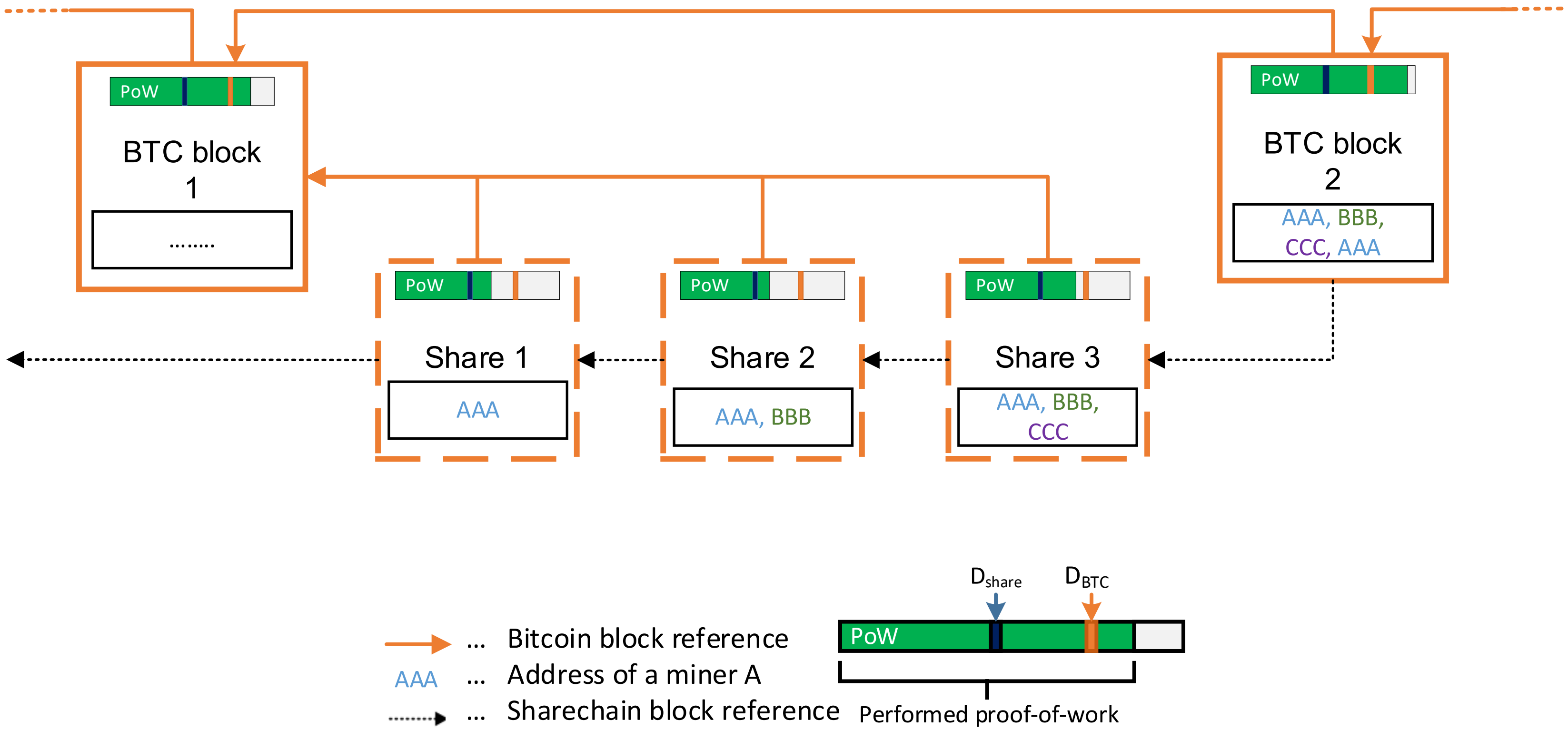
2016
Virtualchain
(Blockstack)

2017
NiPoPoWs
→ First mention of the term **„Velvet Fork“**

P2Pool

- Decentralized Mining pool
- Weak/Near blocks used as pool „shares“
- Additional structure: “Sharechain”
- New rules:
  - Payout scheme (coinbase TX outputs)
  - Prev. Share reference

As seen by P2Pool miners:

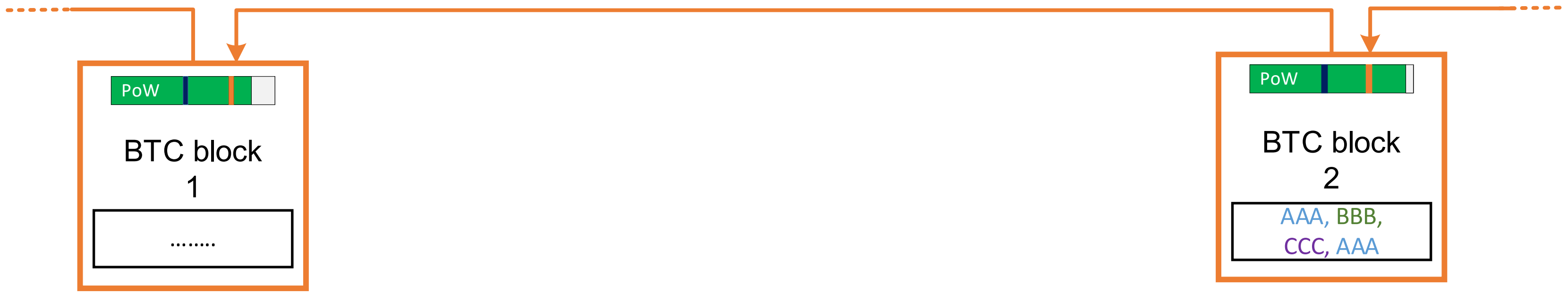


Source: A. Zamyatin, „Merged Mining: Analysis of Effects and Implications“, *MSc Thesis, Vienna University of Technology*, 2017

P2Pool

- Decentralized Mining pool
- Weak/Near blocks used as pool „shares“
- Additional structure: “Sharechain”
- New rules:
  - Payout scheme (coinbase TX outputs)
  - Prev. Share reference

As seen by other miners:



PoW

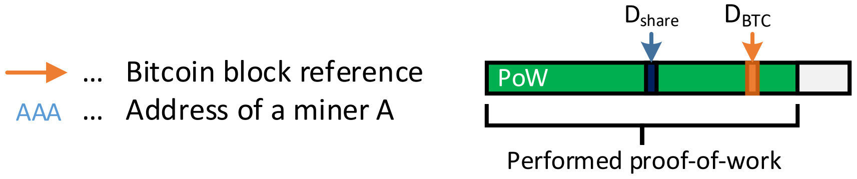
BTC block 1

……..

PoW

BTC block 2

AAA, BBB, CCC, AAA



→ … Bitcoin block reference

AAA … Address of a miner A

D_{share} D_{BTC}

PoW

Performed proof-of-work

Source: A. Zamyatin, „Merged Mining: Analysis of Effects and Implications“, *MSc Thesis, Vienna University of Technology*, 2017

When Velvet Forks Don‘t Work

- When rules need to be enforced across all participants

When Velvet Forks Don‘t Work

- When rules need to be enforced across all participants

SegWit, Bitcoin-NG, … → “Anyone-can-spend“ in the eyes of old clients

When Velvet Forks Can Lead to Problems

- When honest majority is required for safety

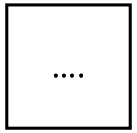
When Velvet Forks Can Lead to Problems

- When honest majority is required for safety

→ Security assumptions of Bitcoin don‘t neccessarily hold!

An Attack on P2Pool

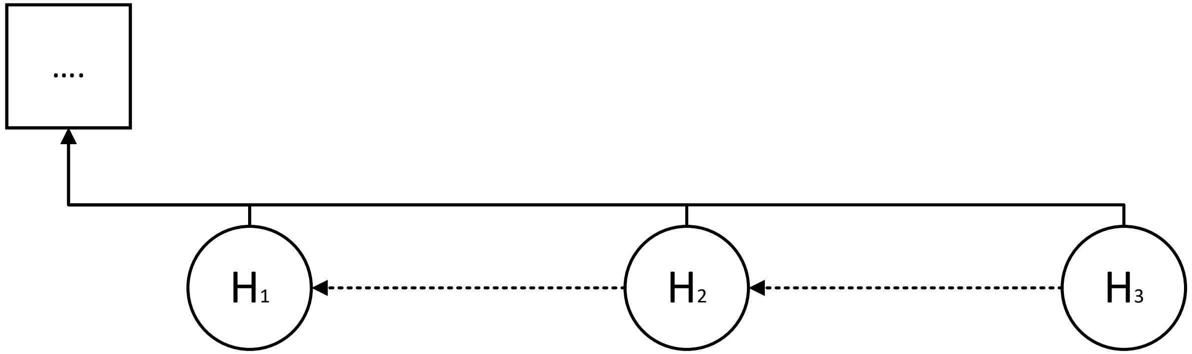
Assume: 5% of the hash rate in P2Pool



....

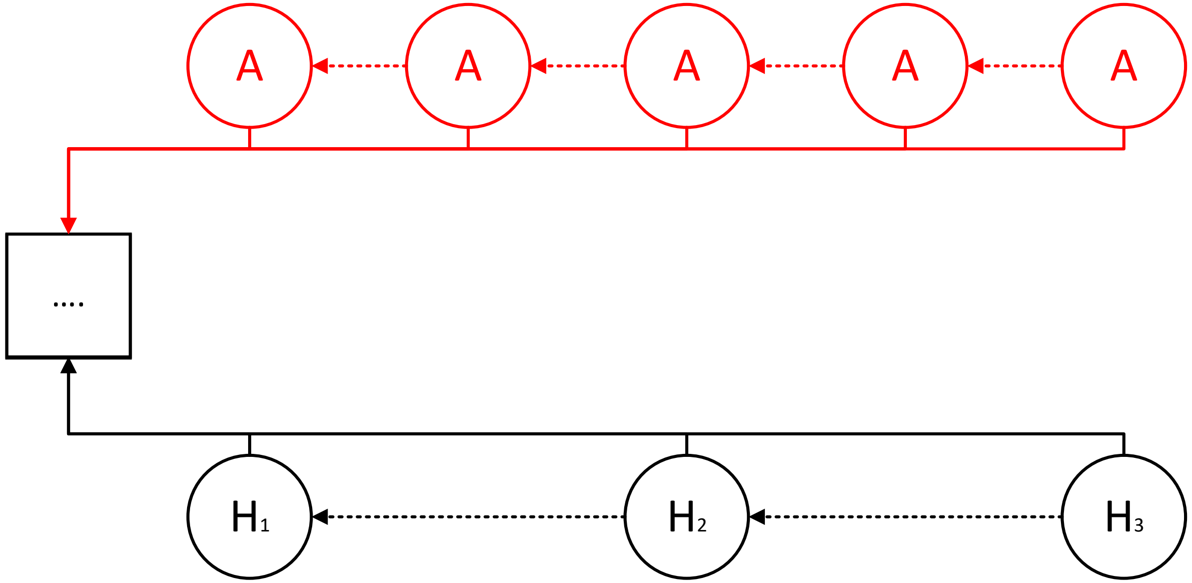
An Attack on P2Pool

Assume: 5% of the hash rate in P2Pool



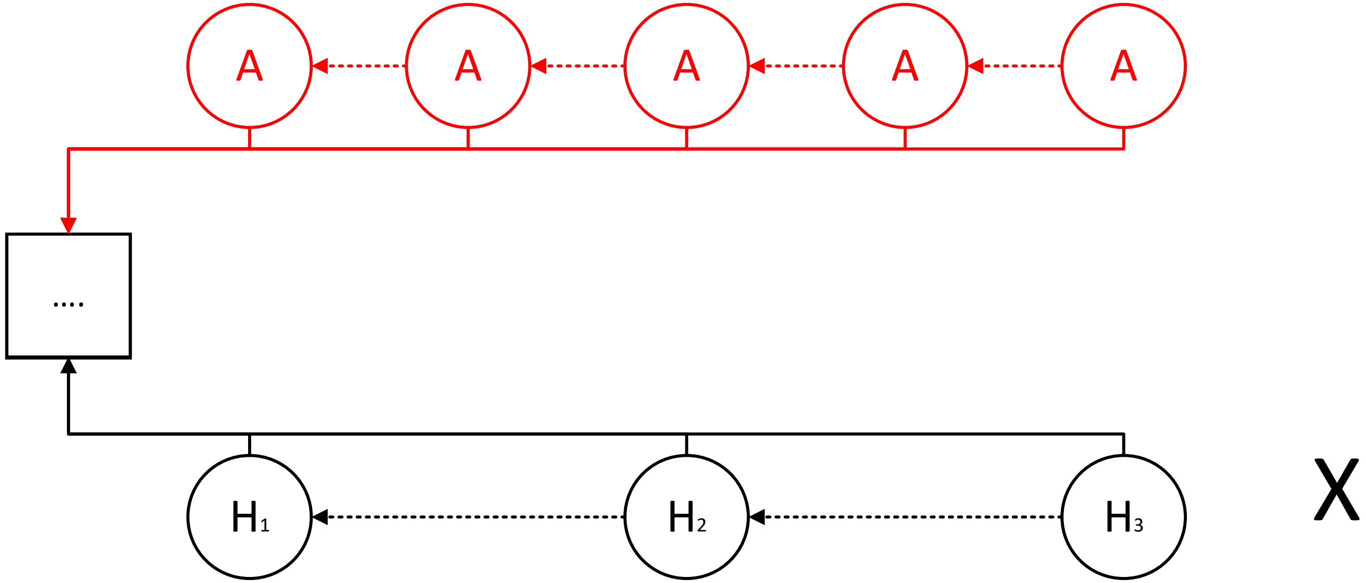
An Attack on P2Pool

Assume: Attacker with 10% of overall hash rate



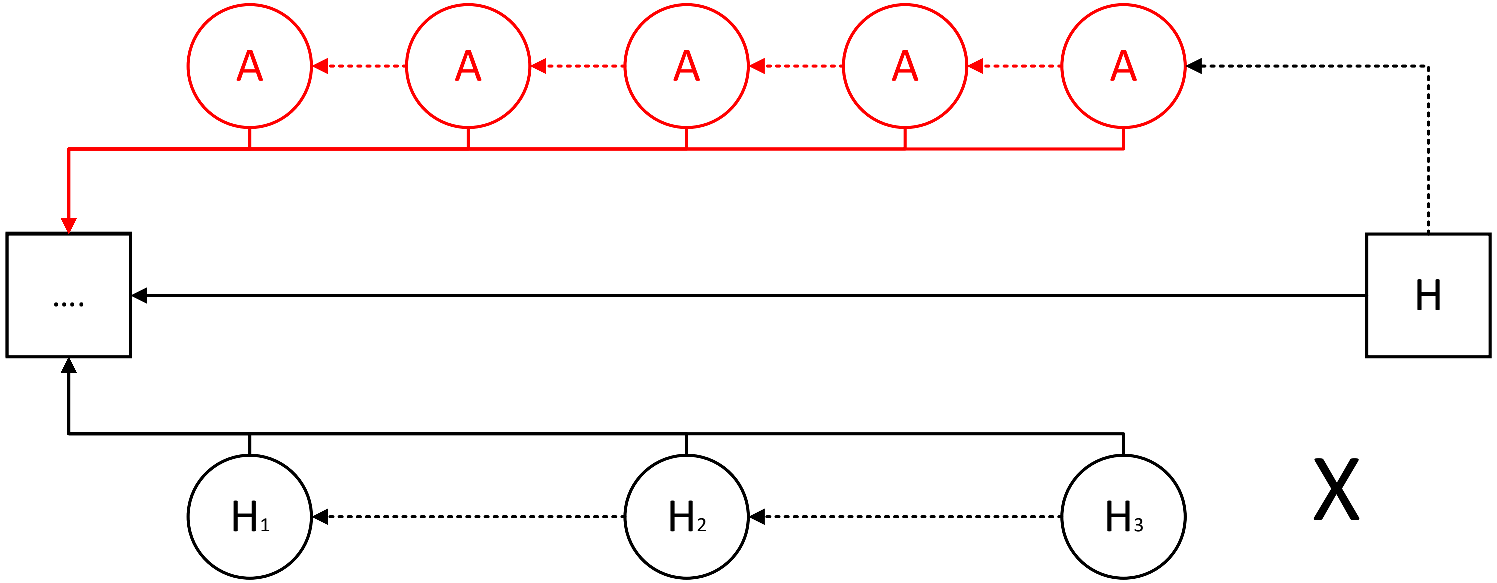
An Attack on P2Pool

Assume: Attacker with 10% of overall hash rate



An Attack on P2Pool

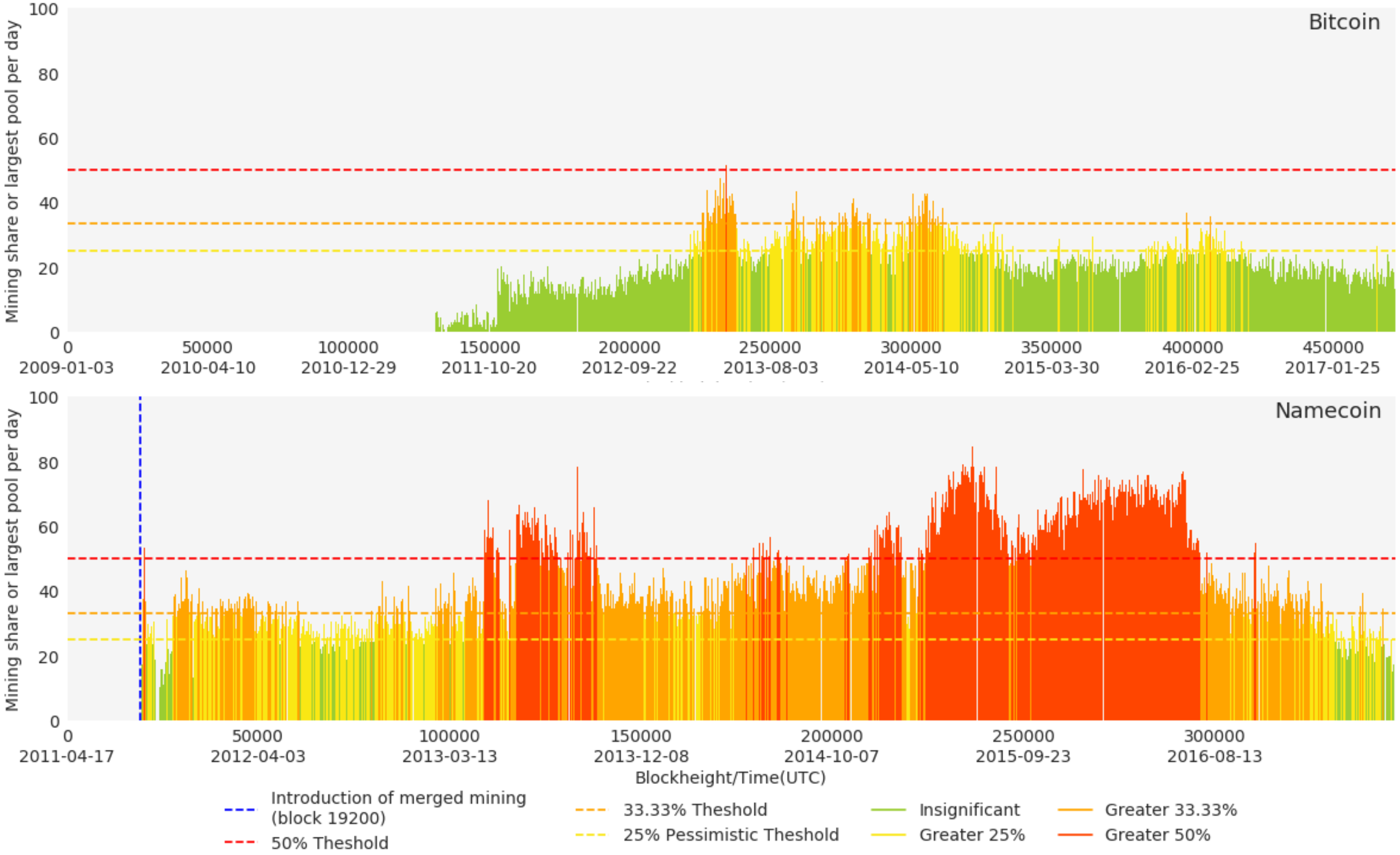
Assume: Attacker with 10% of overall hash rate



Work of honest P2Pool miners H1,H2,H3 is lost!

Merged Mining: Effects of partial adoption

% of Blocks mined by a single miner/pool per day



Merged Mining: Curse of Cure?
Judmayer, A. Zamyatin, N. Stifter, A.G. Voyiatzis and E. Weippl
Data Privacy Management, Cryptocurrencies and Blockchain Technology. Springer, Cham, 2017. 316-333.

When is it Safe to Use Velvet Forks?

- Build upon security **properties** of underlying chain
- No majority required
- Rules don‘t need to be enforced

- Examples:
  - Virtualchain (Blockstack)
  - Colored Coins / Overlay Protocols (e.g., Counterparty)
  - Non-interactive Proofs of Proof-of-Work (NiPoPoWs)

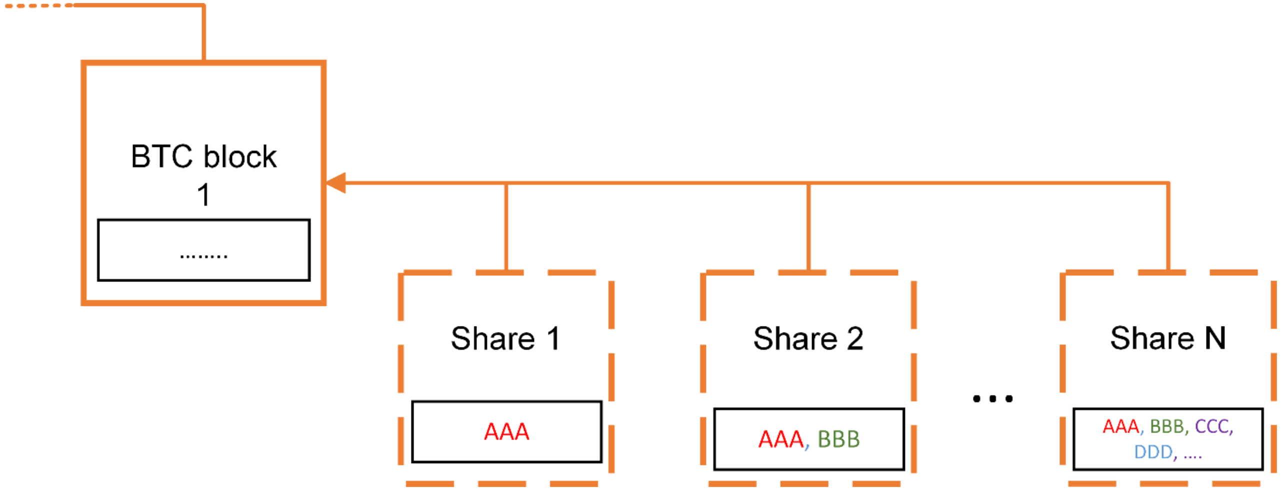
Imperial College London SBA Research

Can Velvet Forks Impact the Security of Legacy Miners?

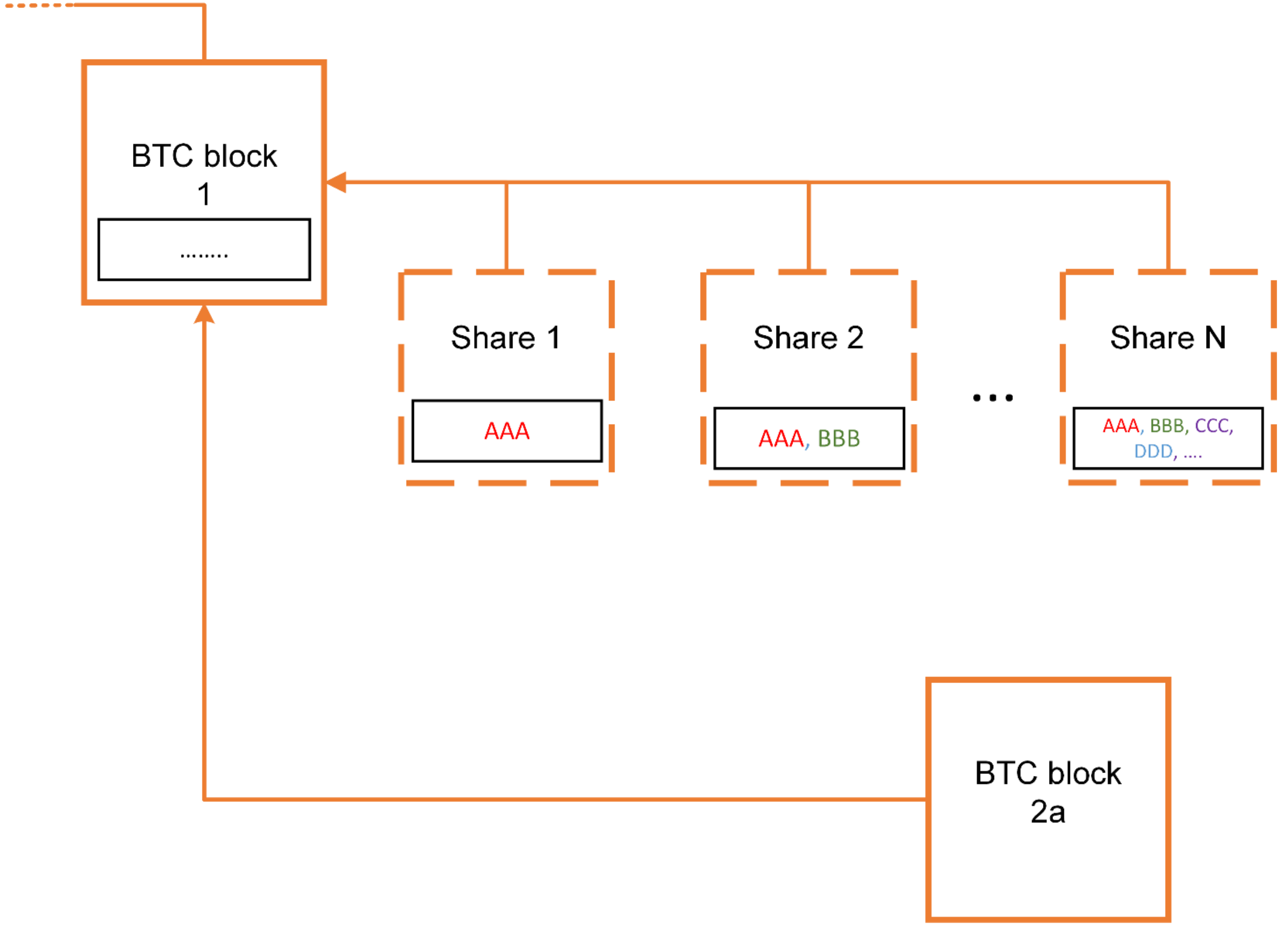
Security Implications

- Blocks may no longer have the same (economic) value to upgraded (velvet) and legacy miners.
  - Possible effects on double spending and selfish mining
  - [Carlsten et al.,‘16] – Petty compliant miners and better timing of selfish mining attacks in a block reward free model

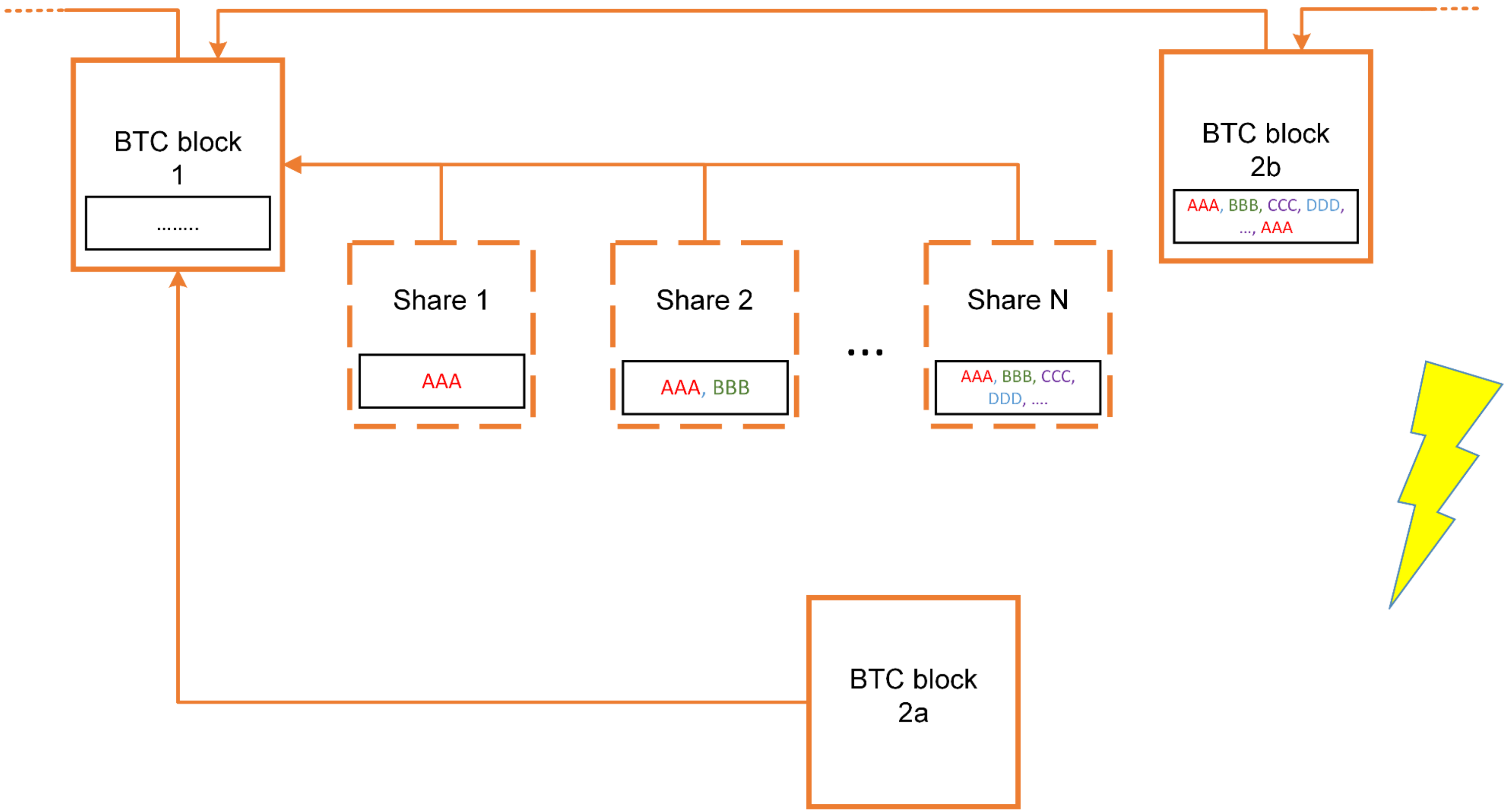
What if …P2Pool were used by the majority of miners?



What if …P2Pool were used by the majority of miners?



What if …P2Pool were used by the majority of miners?



Sufficient incentive to discard the block / fork?

Questions?

Alexei Zamyatin

@alexeiZamyatin

PGP: 2F5F E92D CDAC 15B0 84A6 9FE9 9018 A958 5485 B999