Bitcoin Assessment

Gallersdörfer, U., Holl, P., & Matthes, F. (2019). "Blockchain-based Systems Engineering“. Lecture Slides. TU Munich.

Chair of Software Engineering for Business Information Systems (sebis)
Faculty of Informatics
Technische Universität München
wwwmatthes.in.tum.de

Outline

1. Evolution of Blockchain networks

- Protocol update
- Design
- Signaling
- Potential results

2. Attacks on Blockchain networks

- 51% attack
- Selfish mining attack

3. Limitations of the Bitcoin Blockchain

- Transaction throughput
- Energy consumption
- Comparison with centralized systems

4. Outlook

Evolution of Blockchain networks

As any other software, Blockchains also require **updates**.

These updates affect two parts of the network:

- The **software relying on full nodes** (wallets, etc.)
- The **Blockchain network** (the full node implementations)

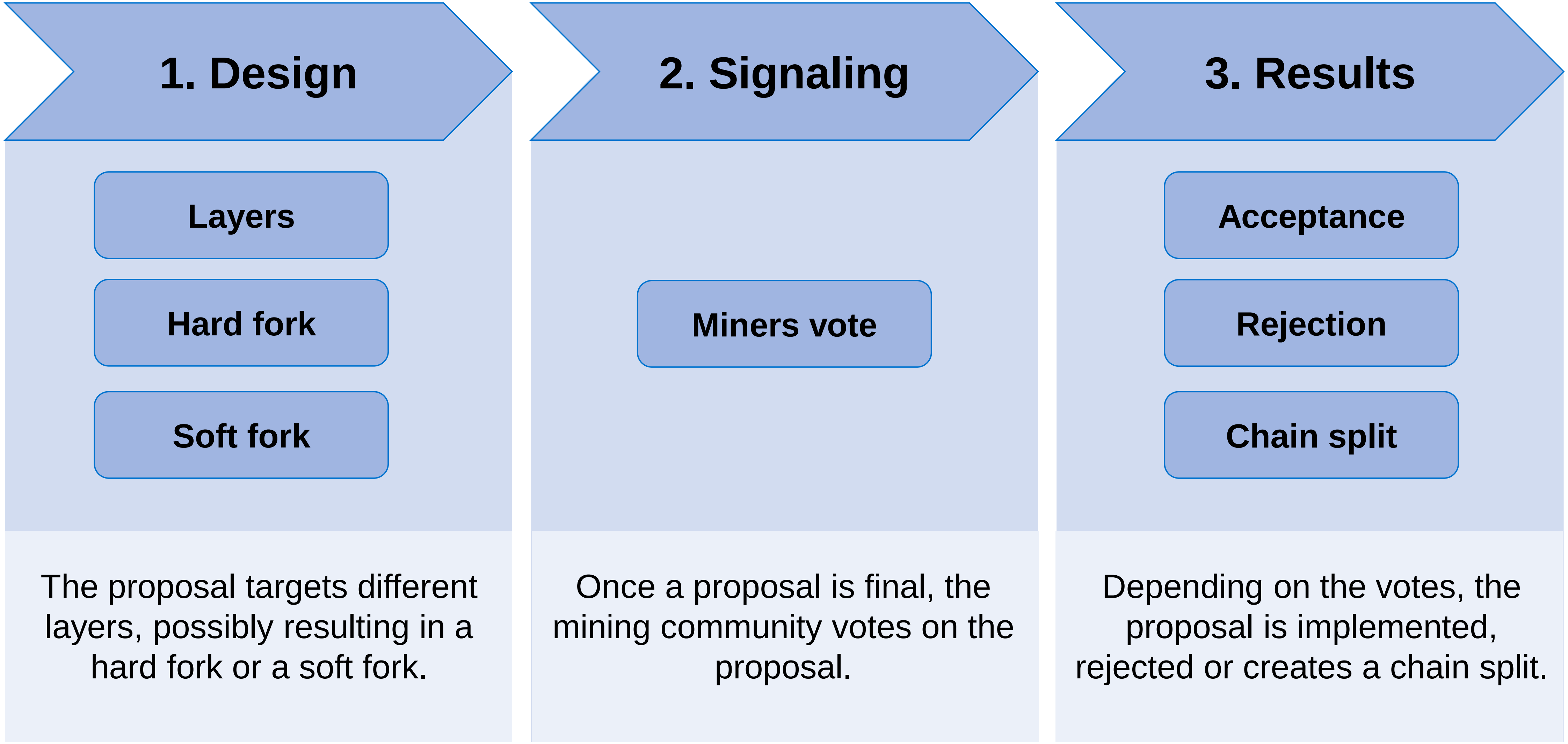
Considering wallets and other software, updates have well-known issues.

1. Incompatibility between old and new software components
   ➔ Old and new software components have to check the version available at runtime
2. Incompatibility between historic data and current data schema expected by the software components
   ➔ Database schema changes and data migration

➔ Therefore, the Bitcoin network inherits these standard problems.

Additionally, the immutable Blockchain data structure and the decentralized P2P-network lead to evolutionary issues. ➔ Process for protocol update

Process of Blockchain protocol update



1. Design

- Layers
- Hard fork
- Soft fork

The proposal targets different layers, possibly resulting in a hard fork or a soft fork.

2. Signaling

- Miners vote

Once a proposal is final, the mining community votes on the proposal.

3. Results

- Acceptance
- Rejection
- Chain split

Depending on the votes, the proposal is implemented, rejected or creates a chain split.

1. Design

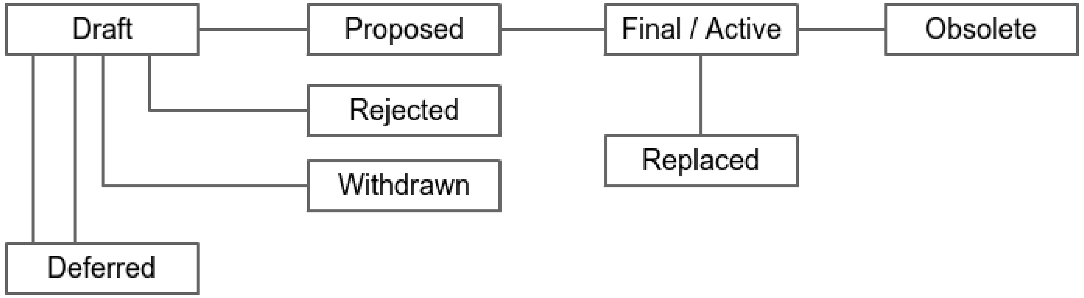
- Proposals can target different layers:
  - Consensus Layer (how to validate states and history)
  - Peer Services Layer (propagation of messages)
  - API/RPC Layer (high-level calls accessible by apps)
  - Applications Layer (high-level structures)

- All proposals in Bitcoin are referred to as **Bitcoin improvement proposals (BIP)**.

- A Github-repository maintained by the core-developers contains all BIPs.

- A BIP contained in the repository is not automatically accepted. Furthermore, the miner community decides whether an BIP is implemented. (See signaling)

- A final BIP contains a detailed description as well as a **reference implementation**. Developers of other clients should be able to also adopt the BIP.



Possible BIP status paths.

Find out more: https://github.com/bitcoin/bips
Process of creating BIPS and status changes https://github.com/bitcoin/bips/blob/master/bip-0002.mediawiki
Different layers in BIPS https://github.com/bitcoin/bips/blob/master/bip-0123.mediawiki

1. Design (cont.)

The terms hard fork and soft fork describe changes within the **consensus layer**.

- **Hard fork**[1]: Structures, that are invalid under old rules become valid under new rules.
- **Soft fork**: Some structures, that were valid under the old rules are no longer valid under the new rules.
- Other changes applying to other layers are not classified as a hard fork or a soft fork. E.g., if a new RPC/API-call is introduced, the consensus layer is not affected.

Examples

The Bitcoin core client specifies a maximum block size of 1MB.

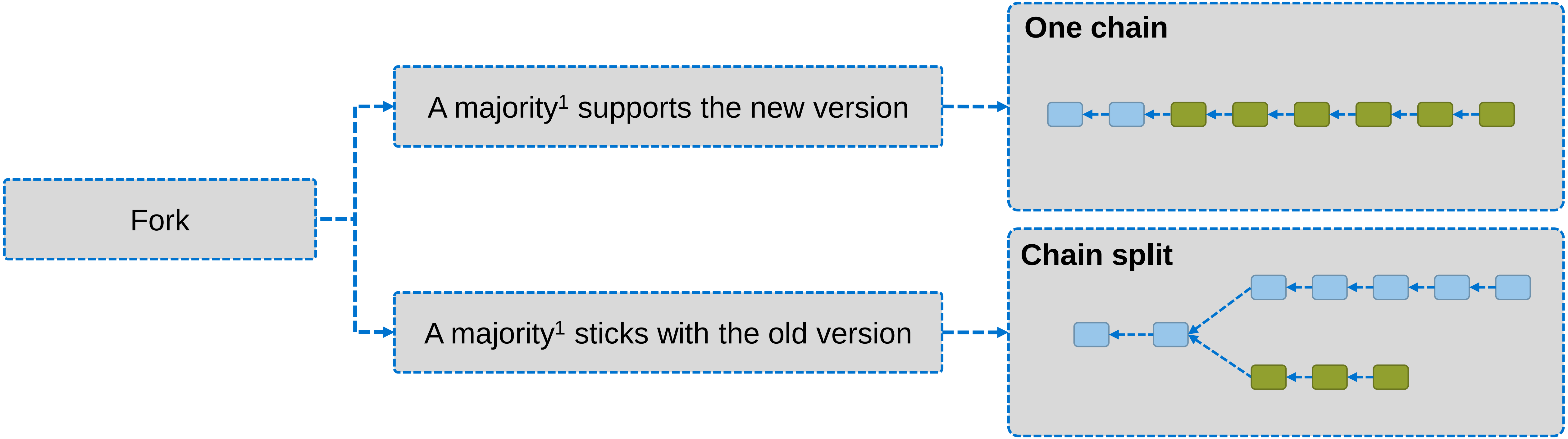
- An update enabling block sizes up to 8MB is considered a **hard fork**.
- An update restricting block sizes up to 0,5MB is considered a **soft fork**.

Why do we need signaling?

[1] *Please note, that we use the definition of a hard fork from BIP123. The creation of new chains sharing history is considered as chain split. https://github.com/bitcoin/bips/blob/master/bip-0123.mediawiki*

Why do we need signaling?

- The following diagram shows that a fork that is not accepted by the majority of miners leads to two chains.



Fork
A majority[1] supports the new version
A majority[1] sticks with the old version
One chain
Chain split

- The disadvantage of a chain split is that both chains compete for users.
  - A split could be the goal of the designer or not.
- To find out whether a chain split would occur, a signaling phase is used.

[1] The majority of the hash power in the overall network.

2. Signaling

Signaling is the process which miners use to vote in favor or against a proposal.

Only miners can express their support. They include special values in the header of the blocks they mine.

How does it work?

- The version-field in the block header contains 32 bit.
- Each author of a proposal (reference implementation) selects a bit in this version-field, a start time and an end time.
- If miners update, their node software uses the bit to signal the support of this miner for this proposal between start time and end time.

- **Option A**: The overall support for this proposal is higher than a certain threshold (usually 1916 out of 2016 blocks ➔ 95%) in one difficulty period. The proposal is **accepted (locked in)** and the rules apply after further 2016 blocks to allow the remaining miners to update as well.

- **Option B**: The overall support for this proposal is lower than the threshold until the end time. The proposal is **rejected**.

- As of the signaling, miners can vote up to 29 different proposals at the same time. Bits are reused after the end time.

3. Results

- If the signaling leads to acceptance, the proposal is automatically implemented.
- Upon a rejection, the community can split (separate the development and go into different directions).

➔ This is called a “Chain split”

Obviously, both communities want to keep the history.

Two main problems arise with the creation of a second chain:

1. If the community behind the hard fork has less than 50% of the computational power, the new chain will not work, as the new software will switch back to the old chain, as the old chain will probably have the higher weight.
2. One transaction can be executed on both chains ➔ Replay attacks

To prevent both problems, the new community has to make the new chain/software incompatible with the old chain. This is done via the creation and adaption of new parameters, such that the old chain is not accepted by the new software and the other way round. Another possibility is to define a second “genesis” block which has to be contained in the longest chain. If the second block contains the new rules (rejected by the old software), the chains are split indefinitely and can not merge together.

3. Results (cont.)

- There have been numerous successful Bitcoin soft forks like *Pay to Script Hash* (P2SH).

- We are not aware of a successful Bitcoin hard fork.

- Some chain splits were executed with *varying* success, e.g.,
  - Bitcoin Cash[1] (8MB blocks instead of 1MB blocks)
  - Bitcoin Gold[2] (changes in the PoW-algorithm for ASIC-resistance)

[1] Bitcoin Cash blocks are on average smaller than the blocks of Bitcoin, see https://thenextweb.com/hardfork/2019/01/17/bitcoin-cash-block-size/
[2] Bitcoin Gold was subject to a 51%-attack, see https://bitcoinist.com/51-percent-attack-hackers-steals-18-million-bitcoin-gold-btg-tokens/

Outline

1. Evolution of Blockchain networks
- Protocol update
- Design
- Signaling
- Potential results

2. Attacks on Blockchain networks
- 51% attack
- Selfish mining attack

3. Limitations of the Bitcoin Blockchain
- Transaction throughput
- Energy consumption
- Comparison with centralized systems

4. Outlook

51% attack

A 51% attack is the worst possible scenario in a Blockchain (based on PoW as consensus mechanism). This means that more than 50 % of the hash power belongs to one entity and this entity uses this power maliciously.

This attack enables

- History rewriting: The attacker can build a Blockchain with the highest accumulated value, defining all contents:
  - Blocking / DoS-ing addresses / users
  - Collecting all mining rewards
  - Creating successful double-spending patterns (orphaning many blocks)

- However:
  - Cannot invent money, cannot propose invalid blocks or transactions, as they would simply be rejected.
  - As of the high hash power, the attacker is highly invested.
  - Entities highly invested have no interest in destroying the network, as they profit the most from it.

➔ A successful executed 51 % attack would destroy the trust in the system and with that, the value of the currency in the system.

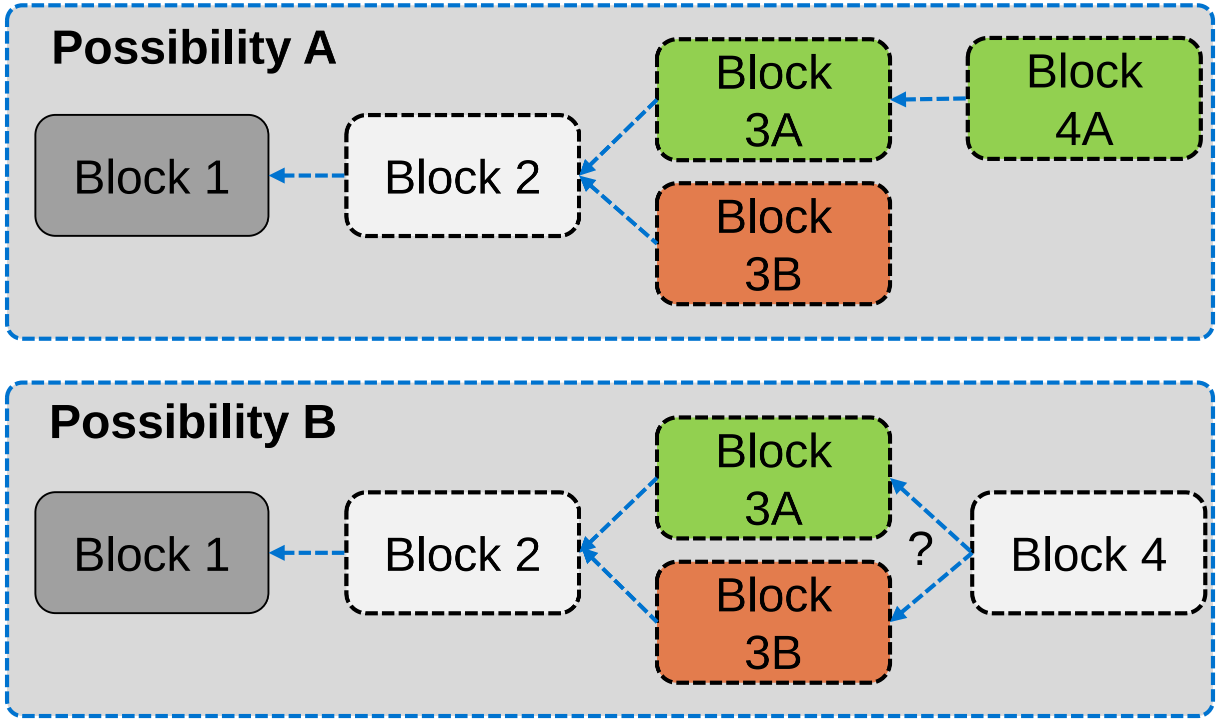
Selfish-mining attack

A selfish-mining attack exploits the probability to be able to propose two blocks one after another.

It works as follows:

- The attacking node finds a new block 3A, but does not propose it to the network
  - **Possibility A**: The node finds a second block 4A building on its block 3A. The network is still at block 2. When the network finds another block 3B, the attacker publishes both block 3A and 4A, making the new 3B an orphan block. The network has worked on an old chain, practically wasting its power.
  - **Possibility B**: When the network proposes a block 3B before the attacker finds a block 4A, the attacker publishes 3A, hoping the network will select block 3A with probability α.

➔ Attack is possible for hashing power minimum of 25% with α = 50% and 33% with α = 0%.



Can only be used to increase profits. Has not been observed in practice.

If $\alpha = 100\ \%$, what is the minimum hash rate needed to do this attack?

Outline

1. Evolution of Blockchain networks
- Protocol update
- Design
- Signaling
- Potential results

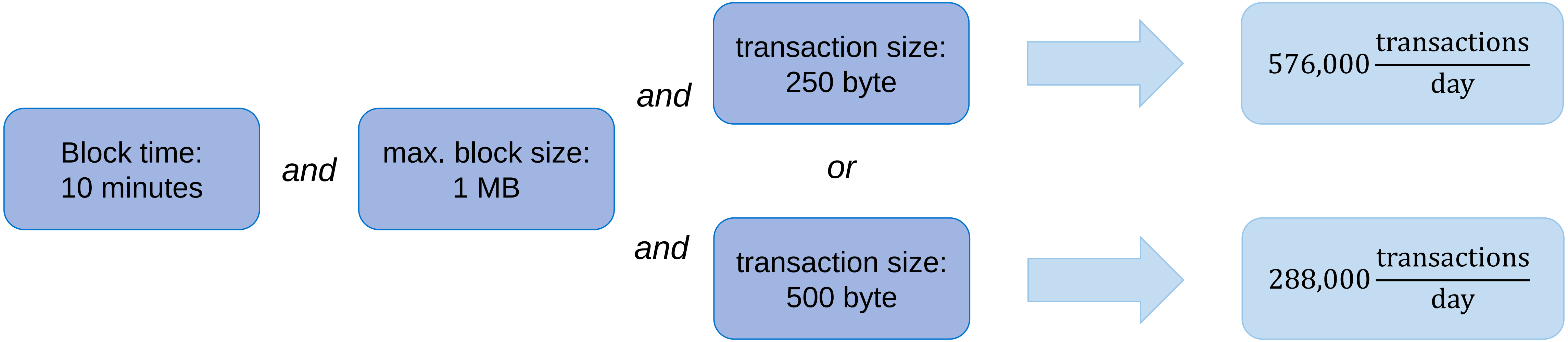
2. Attacks on Blockchain networks
- 51% attack
- Selfish mining attack

3. Limitations of the Bitcoin Blockchain
- Transaction throughput
- Energy consumption
- Comparison with centralized systems

4. Outlook

Bitcoin transaction throughput

In the case of **Bitcoin**, the current technology and protocol introduces a theoretical maximum transaction throughput, determined by three factors:



Block time:
10 minutes

and

max. block size:
1 MB

and

transaction size:
250 byte

$576{,}000\,\frac{\text{transactions}}{\text{day}}$

or

and

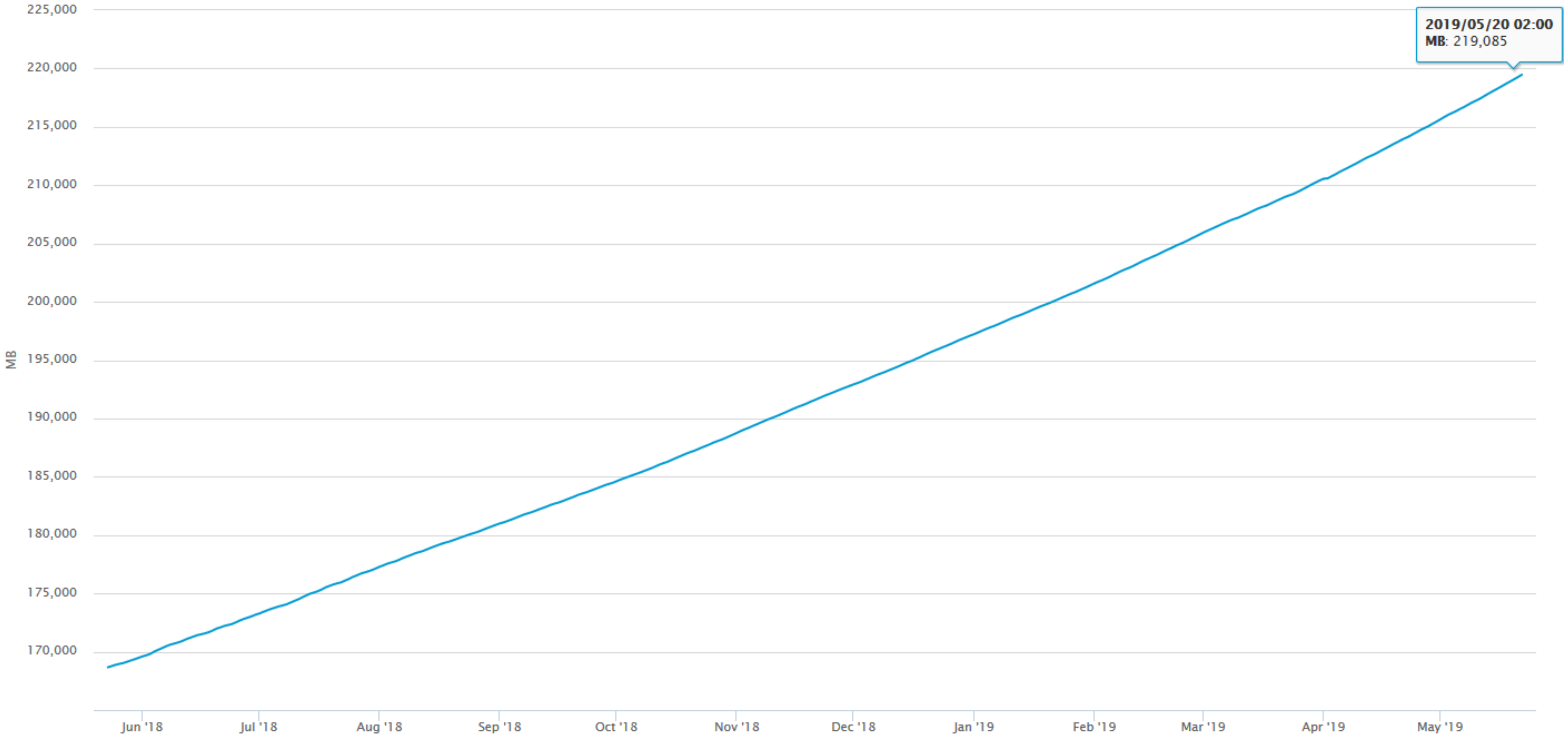
transaction size:
500 byte

$288{,}000\,\frac{\text{transactions}}{\text{day}}$

The recent throughput (May 2019) of 379,574 transactions per day is between the upper and lower bound.

Croman, K. et al. (2016). On Scaling Decentralized Blockchains. In International Conference on Financial Cryptography and Data Security (pp. 106-125). Springer Berlin Heidelberg.

The Blockchain size increases by at most 1MB every 10 minutes



If we would assume that the transaction throughput of Visa (150.000.000 per day), the Bitcoin blockchain would increase daily by:

$$150.000.000 \ * 250 \ byte = \mathbf{37,5\ GB}$$

The graph of the Blockchain size: *https://blockchain.info/de/charts/blocks-size*

Energy consumption in Bitcoin – A naive approach

Difficulties:

- Miners do not disclose their energy consumption and the hardware used by them.

➔ We calculate with the hash rate and the most efficient mining hardware.

Current speed of the network:
4,2 * 10^{19} hashes/s

Average time for a block generation:
10 minutes = 600 s

Average hash tries per block:
600 * 4,2 * 10^{19} = 2,5 * 10^{22}

Energy consumption of Antminer S9:
1372Wh with 14*10^{12} hashes/s

Number of Antminers S9 to provide speed:
4,2 * 10^{19} / 14*10^{12} = 3.000.000 S9

Energy consumed by these Antminers:
3.000.000 * 1,372kWh = 3900 MWh

More information on https://shop.bitmain.com/antminer_s9_asic_bitcoin_miner.htm?flag=overview, picture taken from the website

Energy consumption in Bitcoin – A naive approach (cont.)

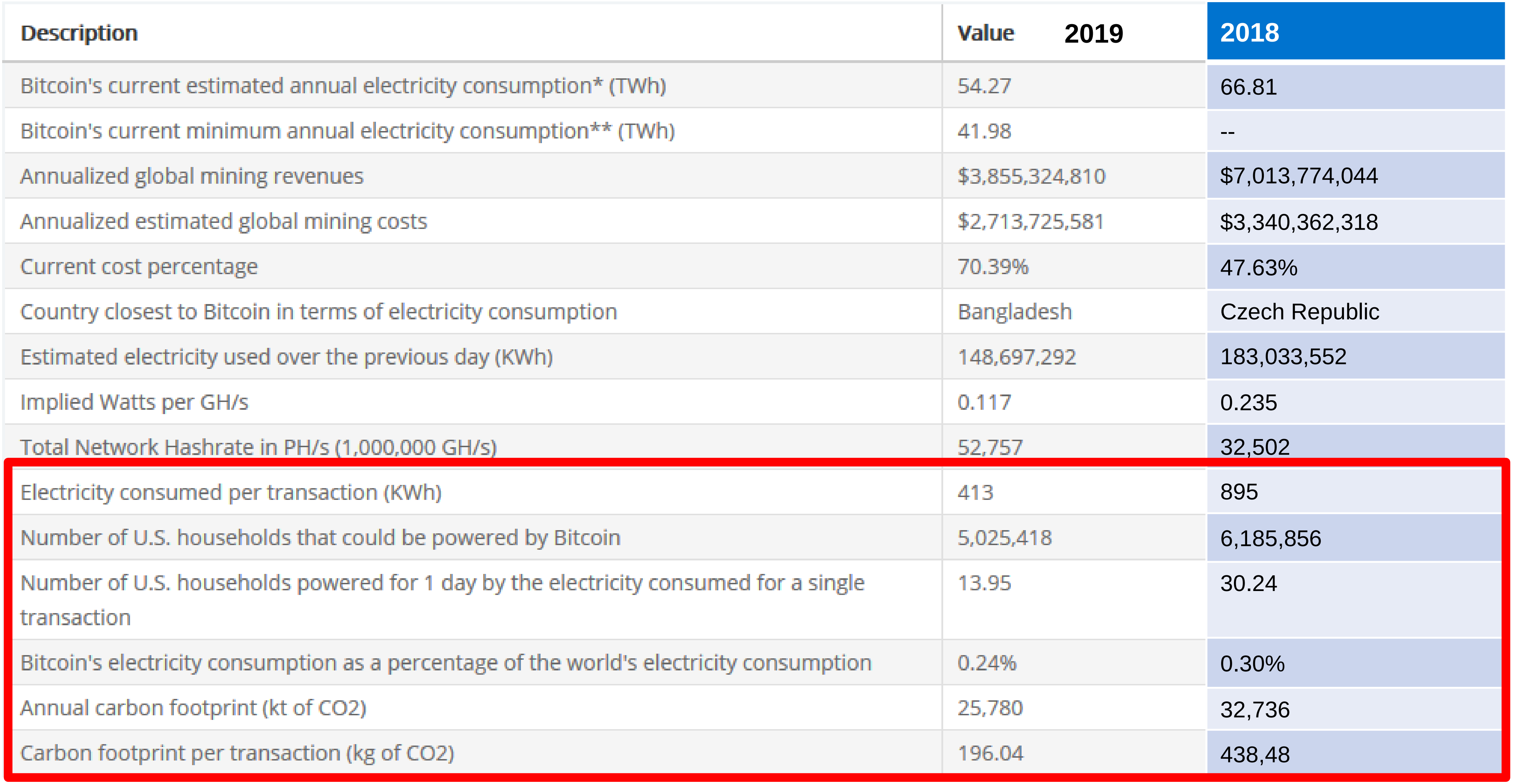
AKW Isar 2

3900 MWh
Consumption

1400 MWh
Production

Image E.ON Kernkraft GmbH - E.ON Kernkraft GmbH taken from https://de.wikipedia.org/wiki/Kernkraftwerk_Isar#/media/File:Gel%C3%A4nde_des_Kernkraftwerk_Isar_(KKI).jpg, not modified

Estimated Bitcoin energy consumption



Description	Value 2019	2018
Bitcoin's current estimated annual electricity consumption* (TWh)	54.27	66.81
Bitcoin's current minimum annual electricity consumption** (TWh)	41.98	--
Annualized global mining revenues	$3,855,324,810	$7,013,774,044
Annualized estimated global mining costs	$2,713,725,581	$3,340,362,318
Current cost percentage	70.39%	47.63%
Country closest to Bitcoin in terms of electricity consumption	Bangladesh	Czech Republic
Estimated electricity used over the previous day (KWh)	148,697,292	183,033,552
Implied Watts per GH/s	0.117	0.235
Total Network Hashrate in PH/s (1,000,000 GH/s)	52,757	32,502
Electricity consumed per transaction (KWh)	413	895
Number of U.S. households that could be powered by Bitcoin	5,025,418	6,185,856
Number of U.S. households powered for 1 day by the electricity consumed for a single transaction	13.95	30.24
Bitcoin's electricity consumption as a percentage of the world's electricity consumption	0.24%	0.30%
Annual carbon footprint (kt of CO2)	25,780	32,736
Carbon footprint per transaction (kg of CO2)	196.04	438,48

As of April 2019

Source: https://digiconomist.net/bitcoin-energy-consumption

Comparison with established centralized solutions

Advantages

- Decentralized management (trust in one party ➔ trust in a system of multiple parties)
- Transparency of all transactions
- Traceability of the complete transaction history
- Pseudonymity of the wallet owners
- *Built-in* financial incentives for early adopters and network growth (➔ business model)

Opportunities

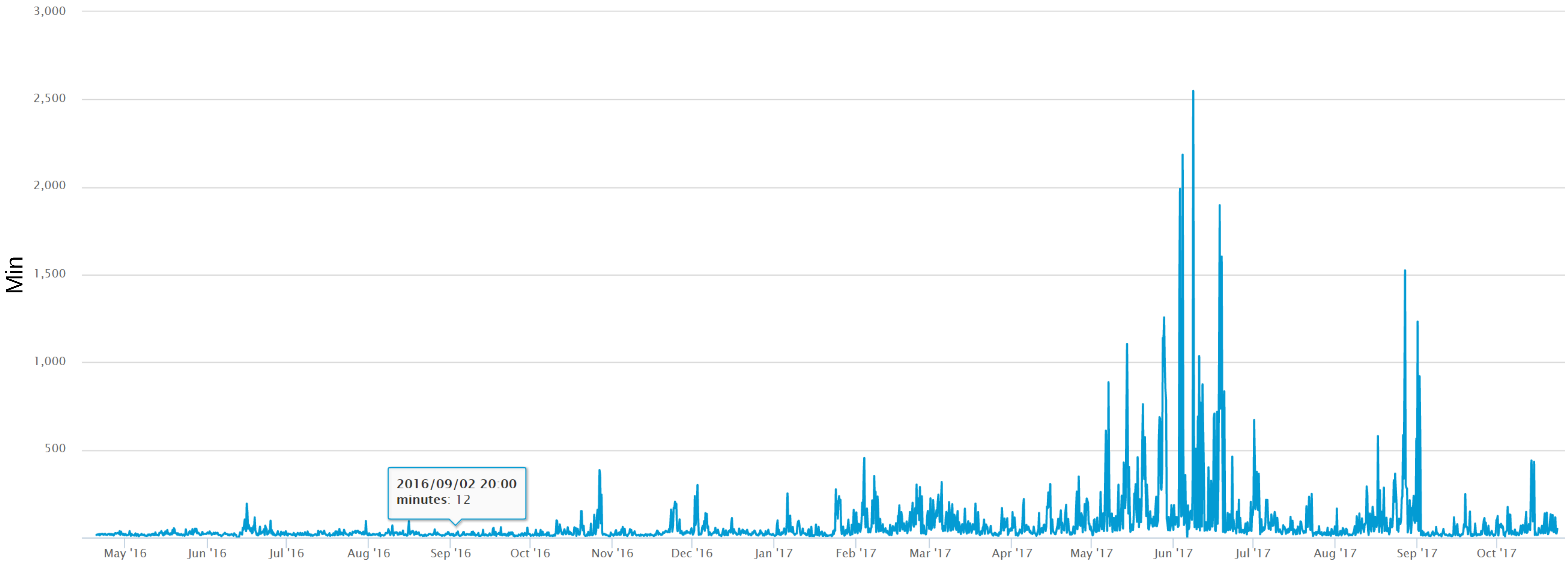
- Innovation thrust for IT solutions in the global finance system
- Lowered entry barriers for IT-savvy players with limited financial resources
- Impetus to re-evaluate established business models and economic mechanisms

Comparison with established centralized solutions

Transaction rate 2019	Average rate	Maximum rate
Bitcoin	~ 3,3	~ 6,6
VISA	2000	>24.000
PayPal	250	?

Sources: https://usa.visa.com/run-your-business/small-business-tools/retail.html
https://www.statista.com/statistics/218495/paypals-net-number-of-payments-per-quarter/

Bitcoin average transaction confirmation time



Outline

1. Evolution of Blockchain networks
- Protocol update
- Design
- Signaling
- Potential results

2. Attacks on Blockchain networks
- 51% attack
- Selfish mining attack

3. Limitations of the Bitcoin Blockchain
- Transaction throughput
- Energy consumption
- Comparison with centralized systems

4. Outlook

Bitcoin faces several challenges

- Bitcoin Script is **limited in its expressive power**!

  ➔ **Ethereum** and other solutions provide a Turing complete[1] Smart Contract language!

- Bitcoin does **not scale** / is too slow!

  ➔ Second layer solutions like **Lightning Network**[2] enable higher transaction throughput with lower fees.

- Bitcoin is **not private**!

  ➔ Cryptocurrencies like **Zcash** or **Monero** enable anonymous (sender, receiver and amount) transactions.

- Bitcoin is **too volatile** to be used as cryptocurrency!

  ➔ Stable coins either pegged by fiat currencies (**Tether**) or by collateral[3] (**Dai**) provide more stable prices.

[1] Turing complete means that any system can be replicated in the respective programming language.

[2] Lightning Network Technical paper: https://lightning.network/lightning-network-paper.pdf

[3] Collateral are other assets in Blockchains, e.g., tokens or cryptocurrencies. Dai is backed by Ether, the cryptocurrency in Ethereum.

DAI Image taken from Introduction to Dai, 2017. Cropped.