Deconstructing Stellar Consensus

Álvaro García-Pérez and Maria A. Schett

IMDEA Software Institute and University College London

The Byzantine consensus challenge

Permissionless blockchains (decentralised protocols like PoW and PoS):

- do not require to know participants *a priori*
- but have asymptotic guarantees and big latency and energy consumption

Permissioned blockchains (classical quorum-based protocols like PBFT):

- hard guarantees and small latency and energy consumption
- but require to know participants *a priori*

The Byzantine consensus challenge

Permissionless blockchains (decentralised protocols like PoW and PoS):

- do not require to know participants *a priori*
- but have asymptotic guarantees and big latency and energy consumption

Permissioned blockchains (classical quorum-based protocols like PBFT):

- hard guarantees and small latency and energy consumption
- but require to know participants *a priori*

The recent **federated Byzantine agreement** by Stellar and Ripple combines features of the two worlds above:

- each participant individually chooses who to trust, no central authority
- quorums arise from individual choices, and participants operate with local information
- hard guarantees and small latency and energy consumption

Motivation and contribution

However, federated Byzantine agreement is poorly understood, and its correctness has not been investigated as thoroughly as other classical solutions for BFT based on quorums.

Motivation and contribution

However, federated Byzantine agreement is poorly understood, and its correctness has not been investigated as thoroughly as other classical solutions for BFT based on quorums.

Our contribution:

- We focus on the **Stellar consensus protocol (SCP)**, as described in Stellar's whitepaper at www.stellar.org and in David Mazières's blog at www.scs.stanford.edu.
- Prove SCP correct by reusing proof of core component **federated voting**, previously investigated.

Byzantine Consensus

What is Byzantine consensus?

Abstraction for distributed systems in which *honest* nodes can only fail by stopping, and *malicious* nodes fail by deviating arbitrarily from the protocol specification.

Each *correct* node *proposes* some value x, and eventually all correct nodes *decide* one and the same value y.

What is Byzantine consensus?

Abstraction for distributed systems in which *honest* nodes can only fail by stopping, and *malicious* nodes fail by deviating arbitrarily from the protocol specification.

Each *correct* node *proposes* some value x, and eventually all correct nodes *decide* one and the same value y.

Formally, Byzantine consensus enjoys the following properties:

Safety

(*Agreement*) No two correct nodes decide differently.

(*Validity*) If every node is correct, then a node can only decide a value that was proposed by some node.

Liveness

(*Termination*) Every correct node eventually decides a value.

What do we prove about SCP?

Assume a *partially synchronous* system in which a reliable network delivers messages in bounded time after *global stabilisation time* (GST).

Properties are relative to disjoint fragments of the system that are internally consistent and contain only correct nodes, called *intact sets*:

Given any maximal intact set I:

Safety

(*Agreement*) No two nodes in I decide differently.

(*Validity*) If every node is honest, then a node in I can only decide a value that was proposed by some node.

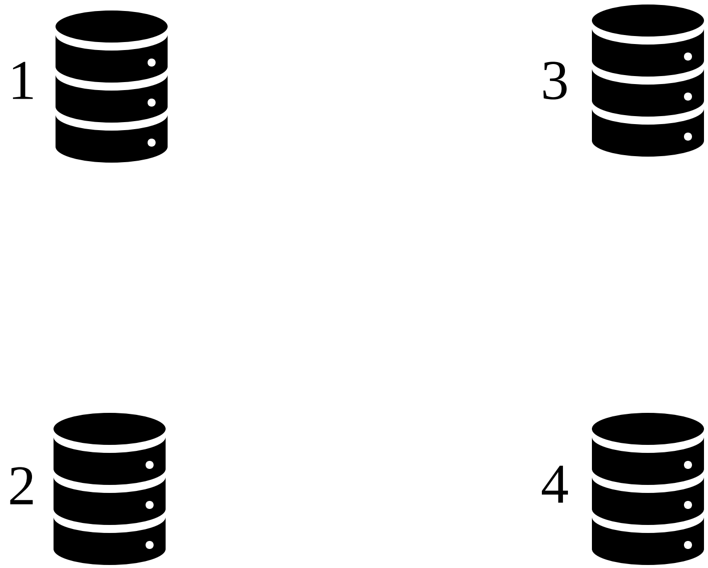
Liveness

(*Non-blocking*) If a node v in I has not decided a value yet, then in every continuation of the run in which malicious nodes stop, the node v eventually decides some value.

Federated Byzantine Quorum Systems (FBQS)

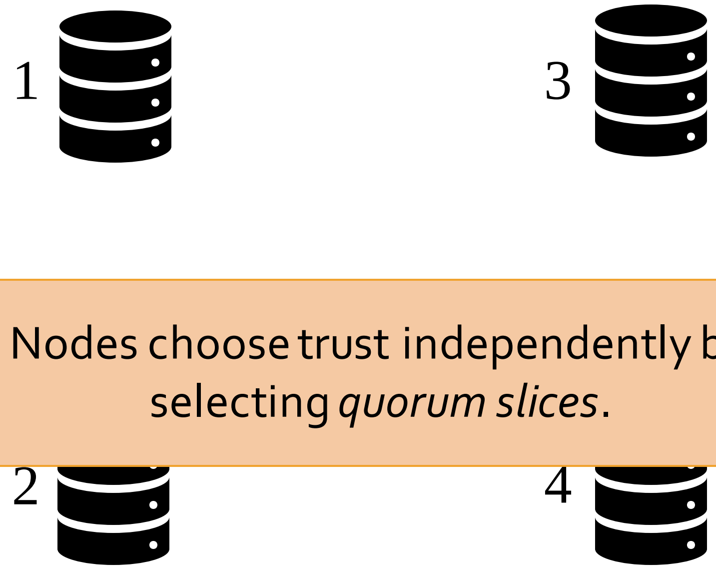
Federated Byzantine quorum systems (FBQS)

$$\mathbb{V} = \{1,2,3,4\}$$



Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

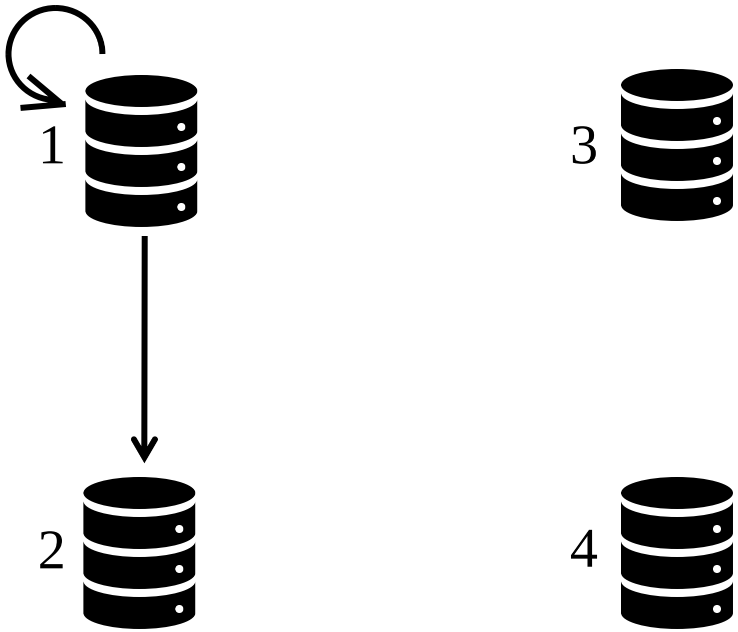


Nodes choose trust independently by selecting *quorum slices*.

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

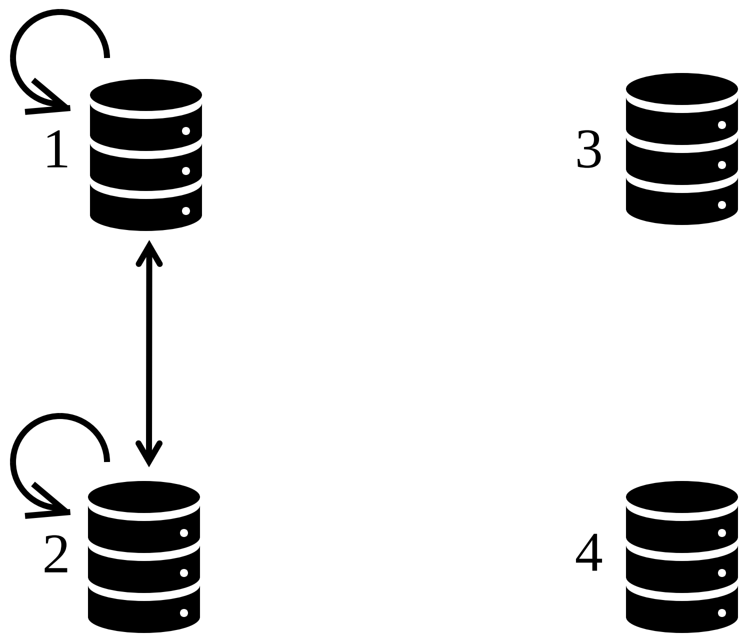
$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$



Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

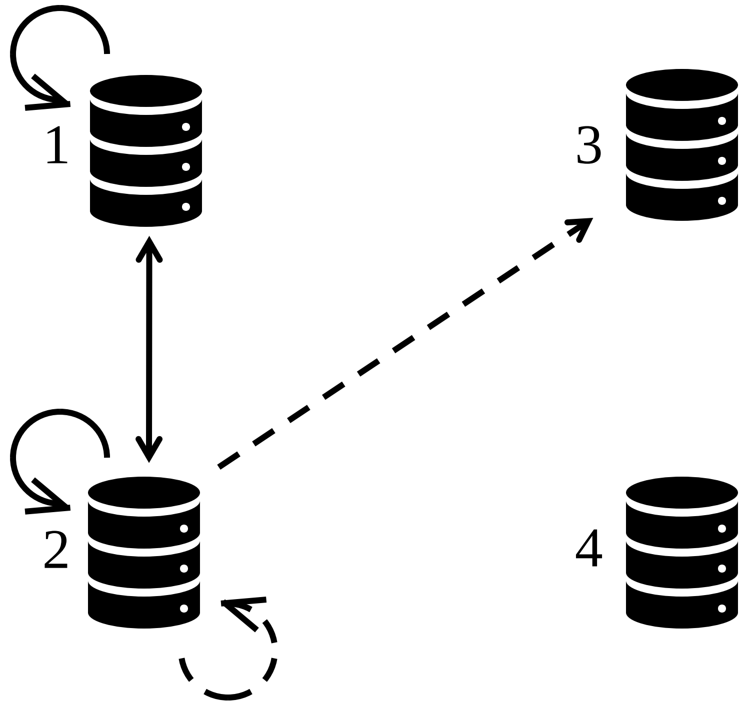
$\mathbb{S} : \mathbb{V} \to 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\}\}$



Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \to 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$



Federated Byzantine quorum systems (FBQS)

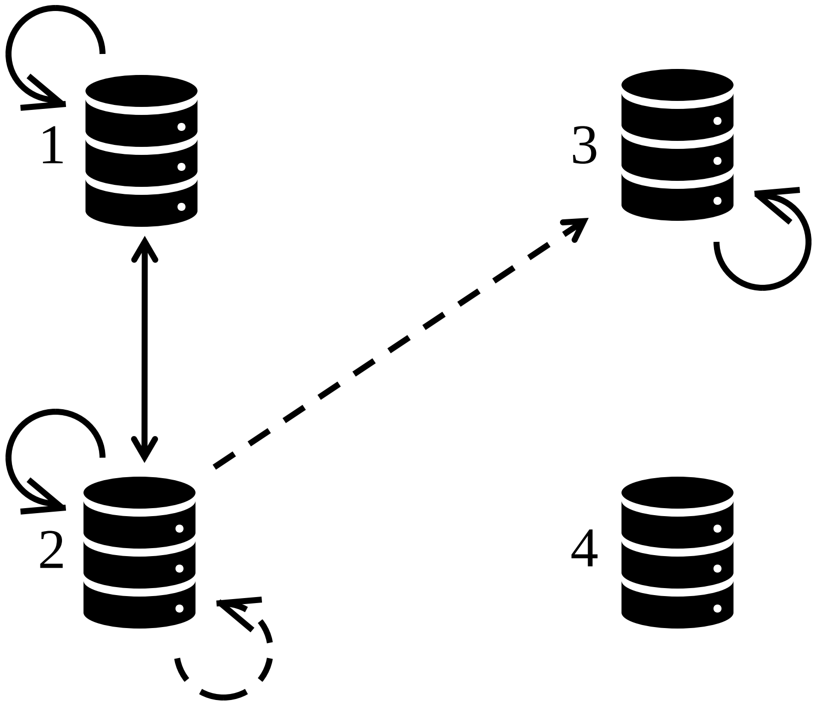
$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$

$\mathbb{S}(1) = \{\{1,2\}\}$

$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$

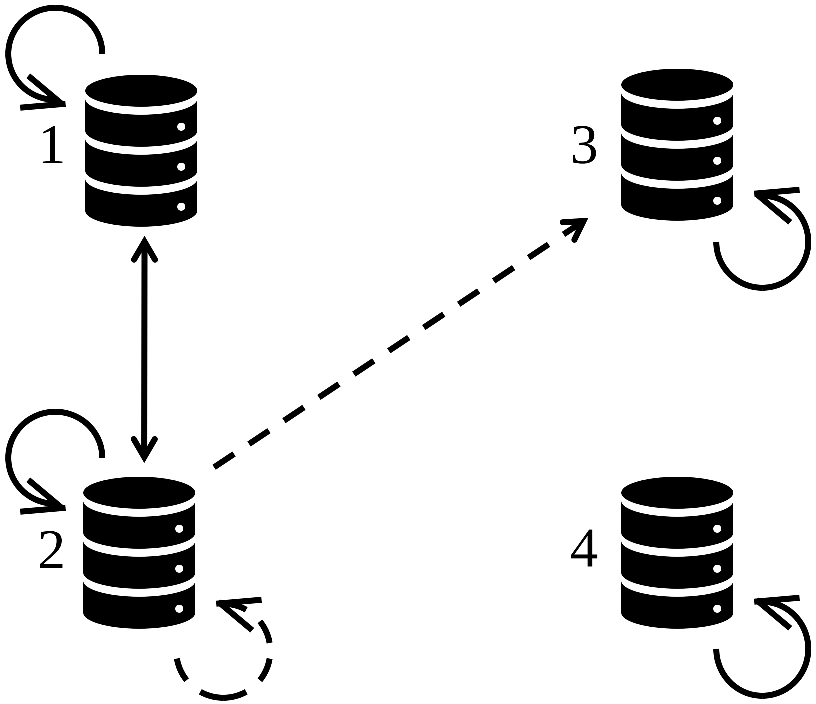
$\mathbb{S}(3) = \{\{3\}\}$



Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

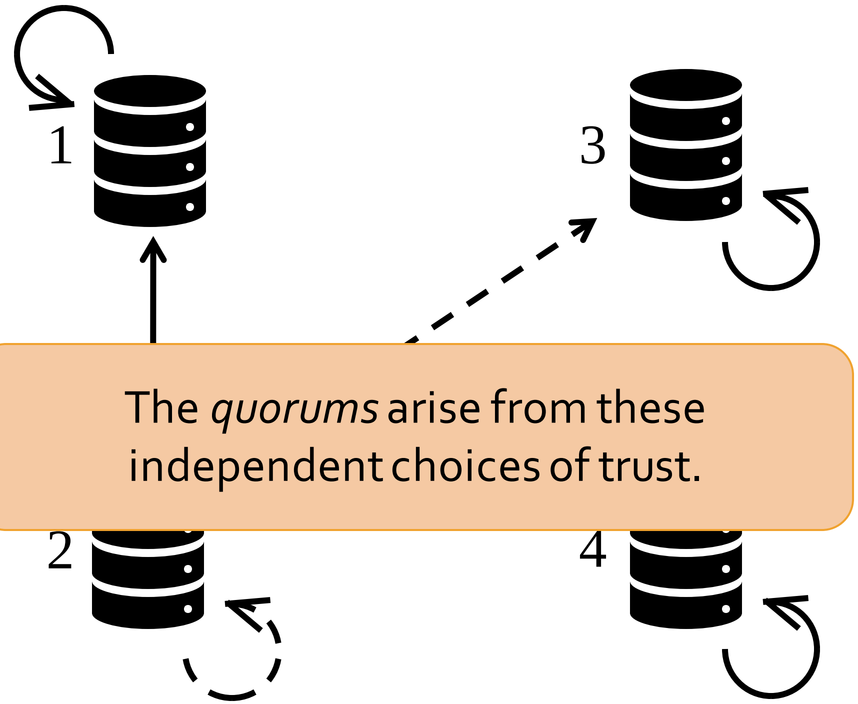
$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$



Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$



The *quorums* arise from these independent choices of trust.

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

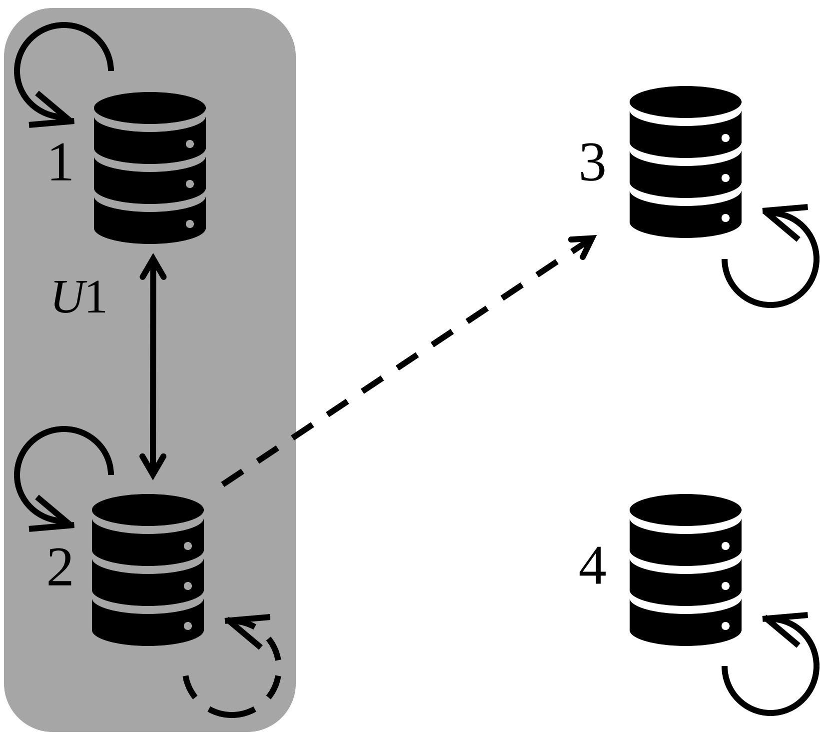
$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$

$\mathbb{S}(1) = \{\{1,2\}\}$

$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$

$\mathbb{S}(3) = \{\{3\}\}$

$\mathbb{S}(4) = \{\{4\}\}$

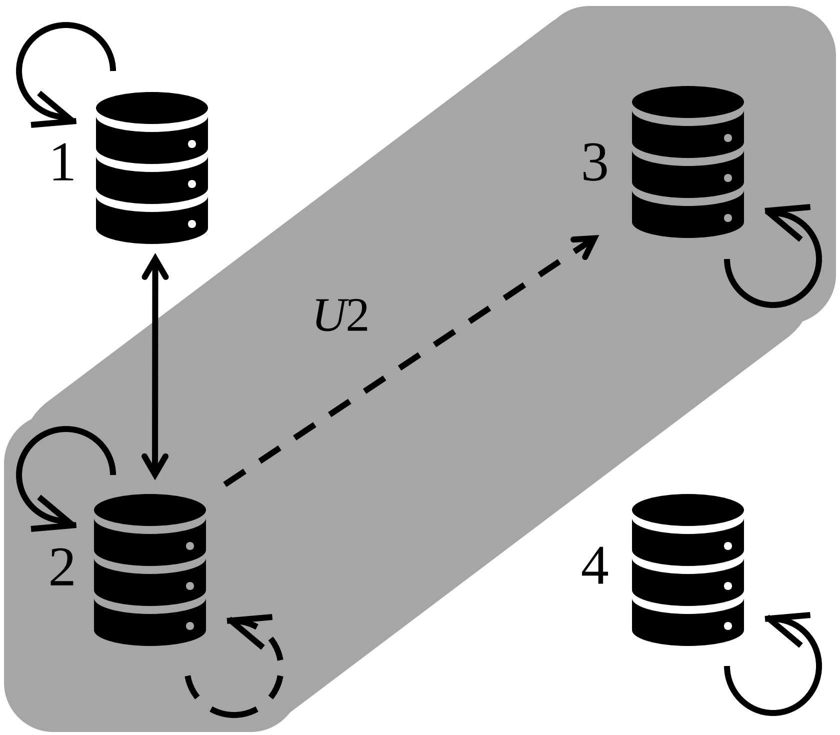


$U1 = \{1,2\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$

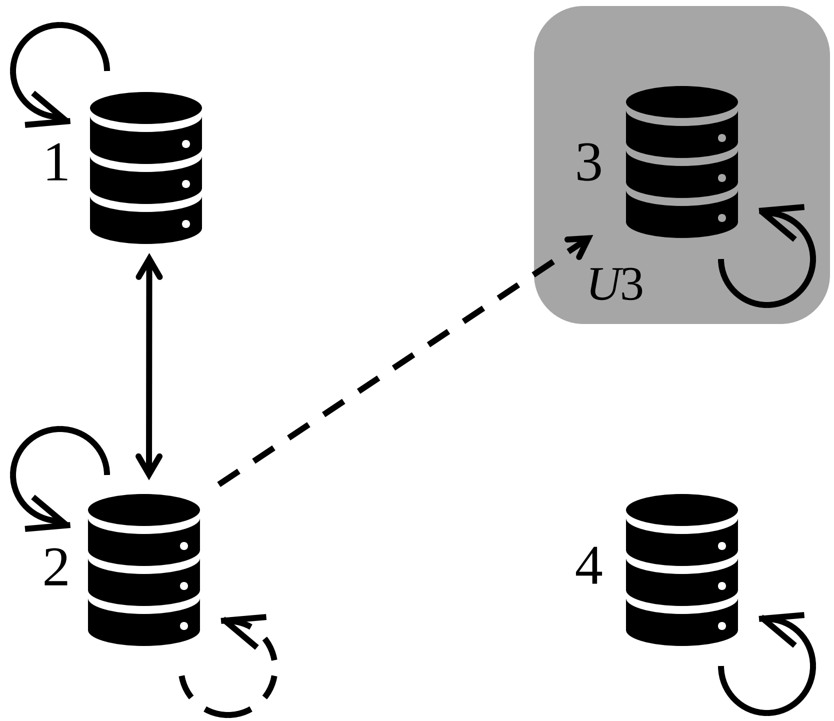


$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$

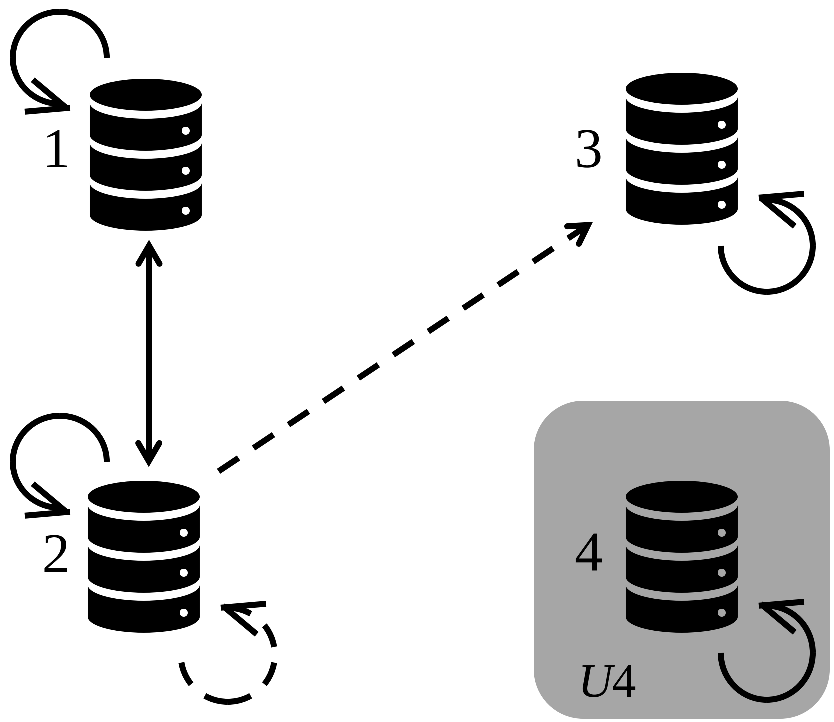


$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$

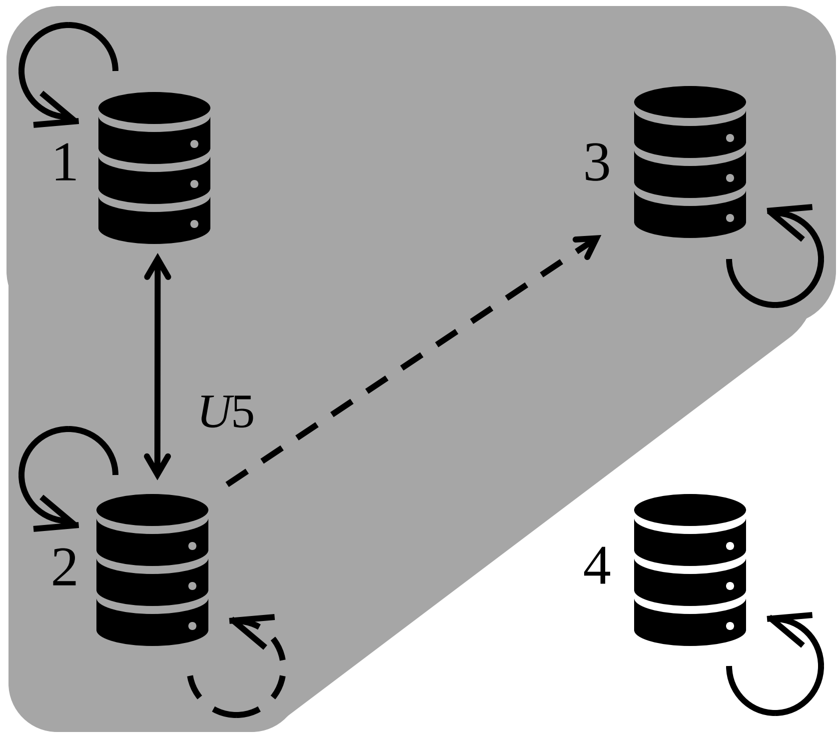


$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$



$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

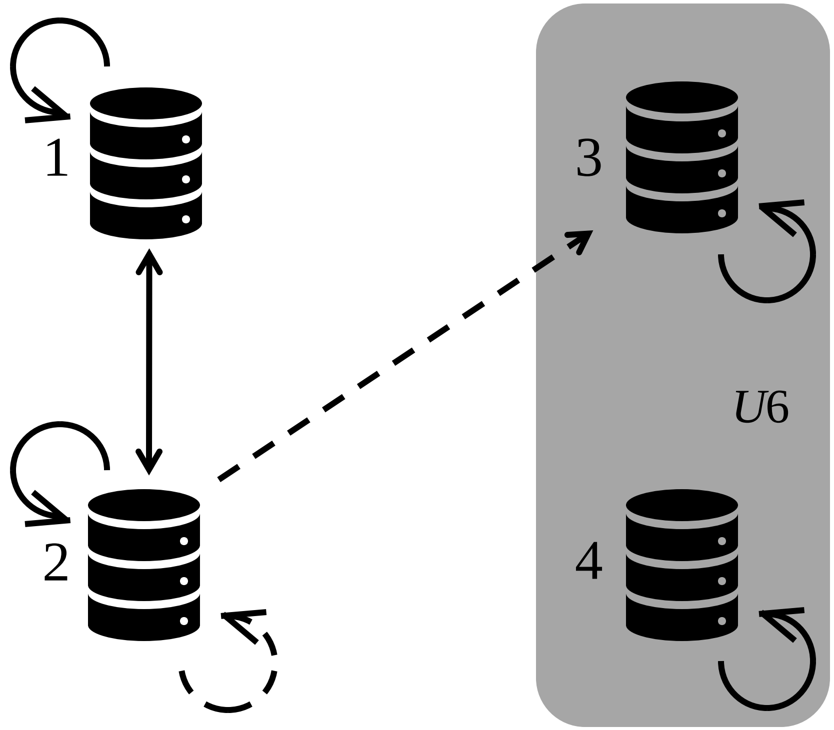
$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$

$\mathbb{S}(1) = \{\{1,2\}\}$

$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$

$\mathbb{S}(3) = \{\{3\}\}$

$\mathbb{S}(4) = \{\{4\}\}$



$U1 = \{1,2\} \in \mathbb{Q}$

$U2 = \{2,3\} \in \mathbb{Q}$

$U3 = \{3\} \in \mathbb{Q}$

$U4 = \{4\} \in \mathbb{Q}$

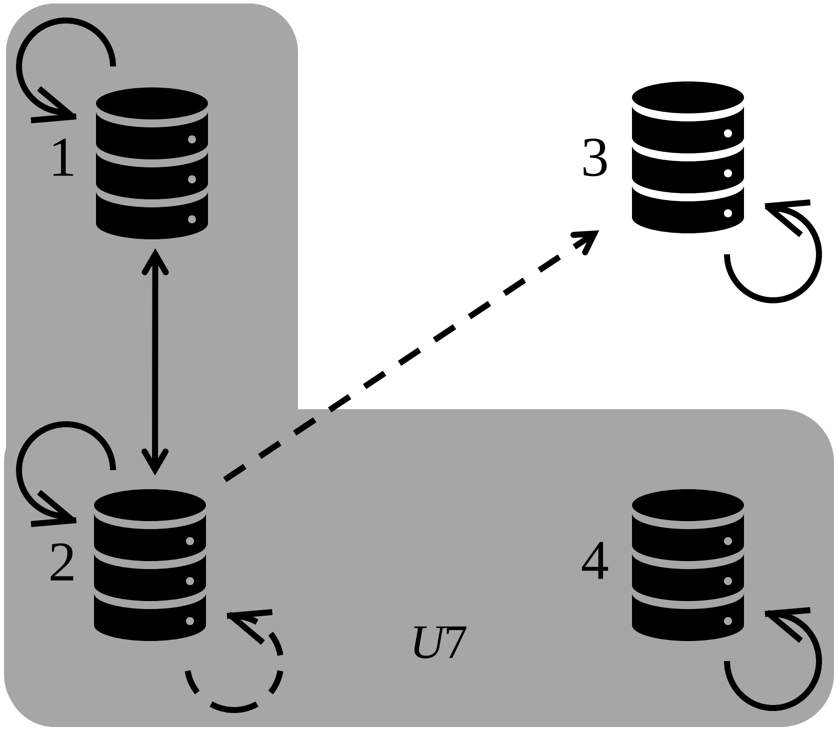
$U5 = \{1,2,3\} \in \mathbb{Q}$

$U6 = \{3,4\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \to 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$

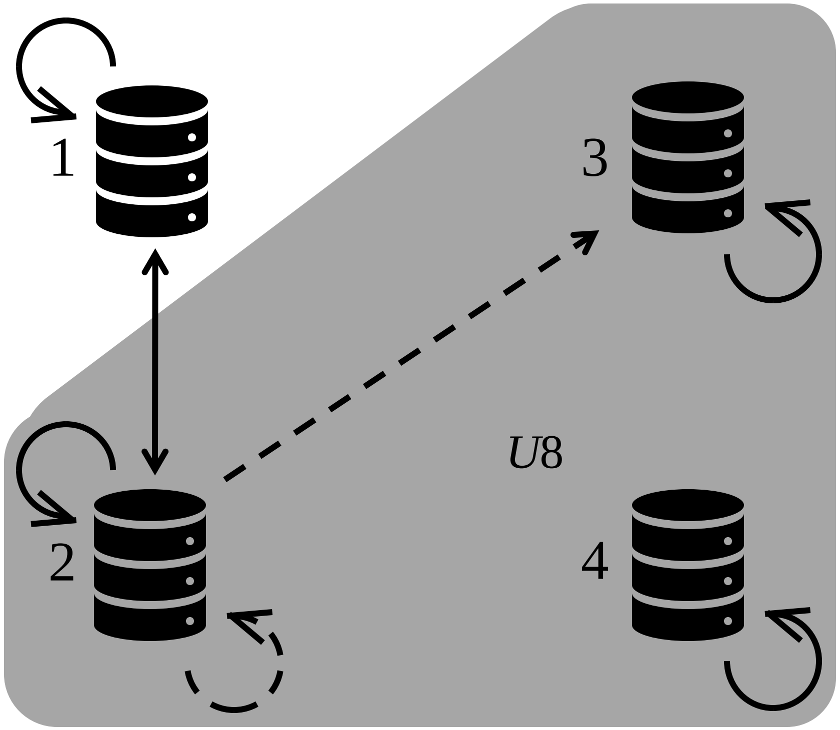


$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$

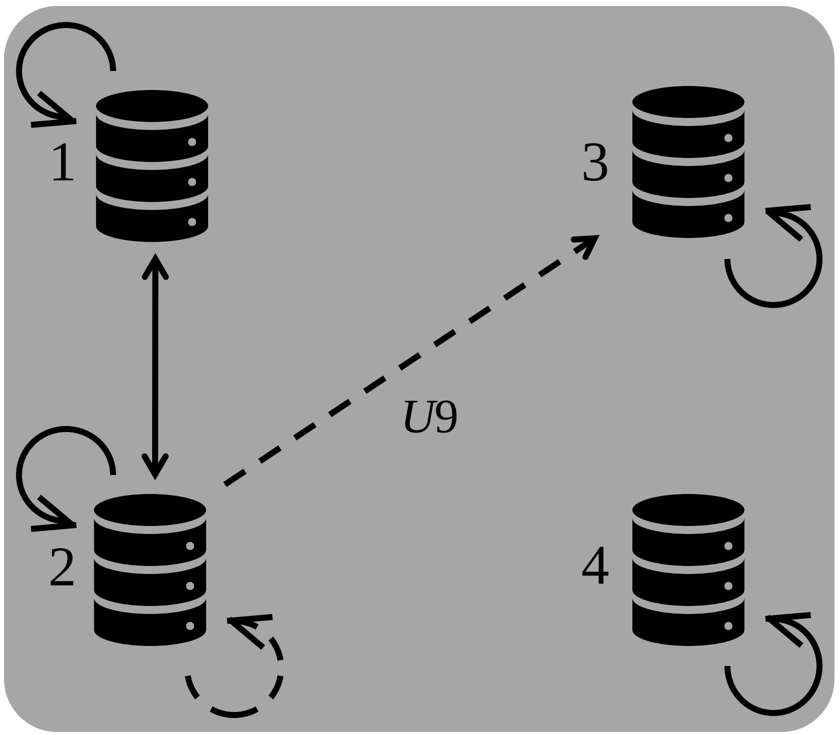


$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$
$U8 = \{2,3,4\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$

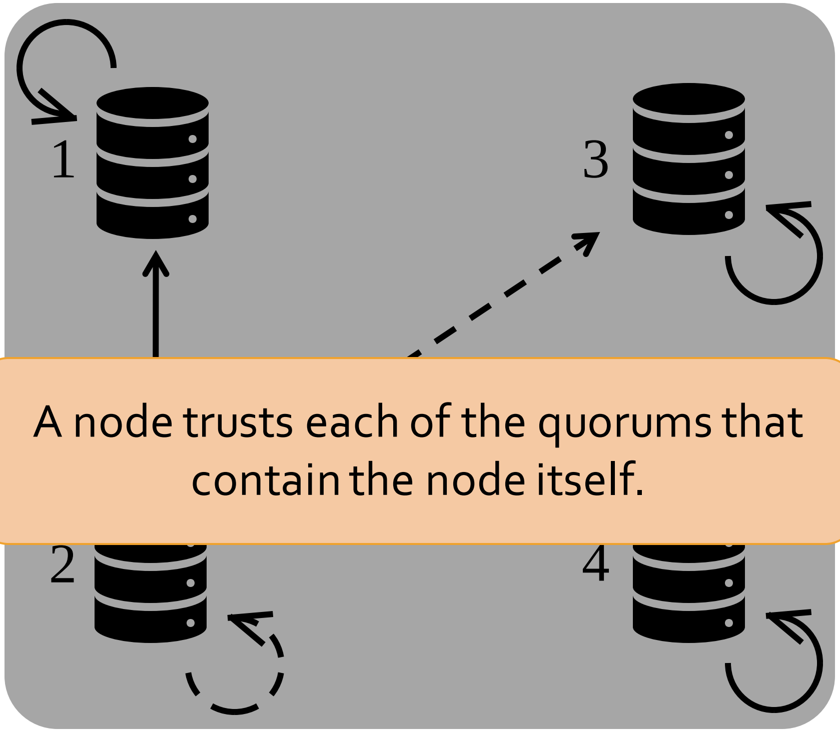


$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$
$U8 = \{2,3,4\} \in \mathbb{Q}$
$U9 = \{1,2,3,4\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$



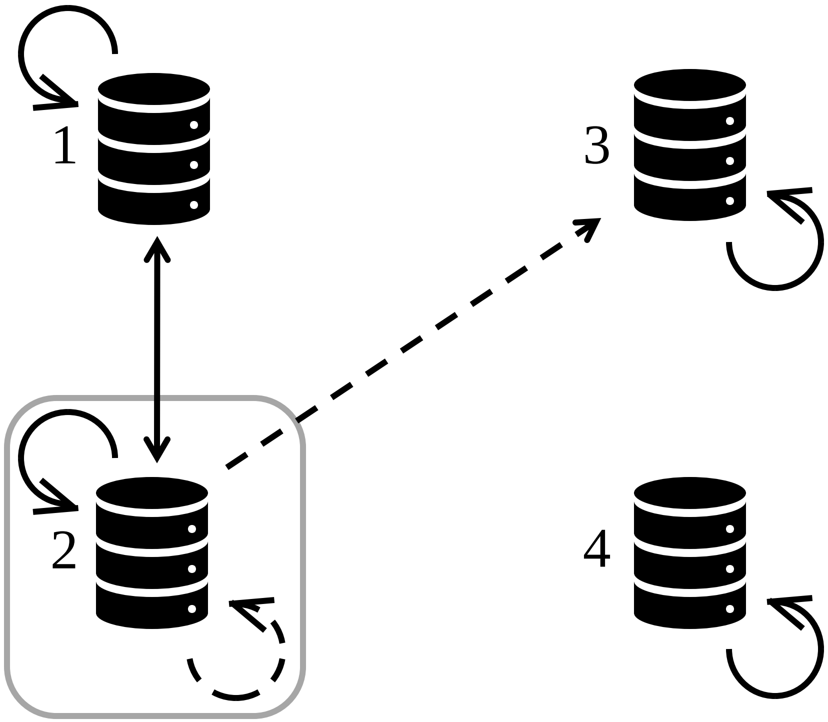
A node trusts each of the quorums that contain the node itself.

$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$
$U8 = \{2,3,4\} \in \mathbb{Q}$
$U9 = \{1,2,3,4\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$

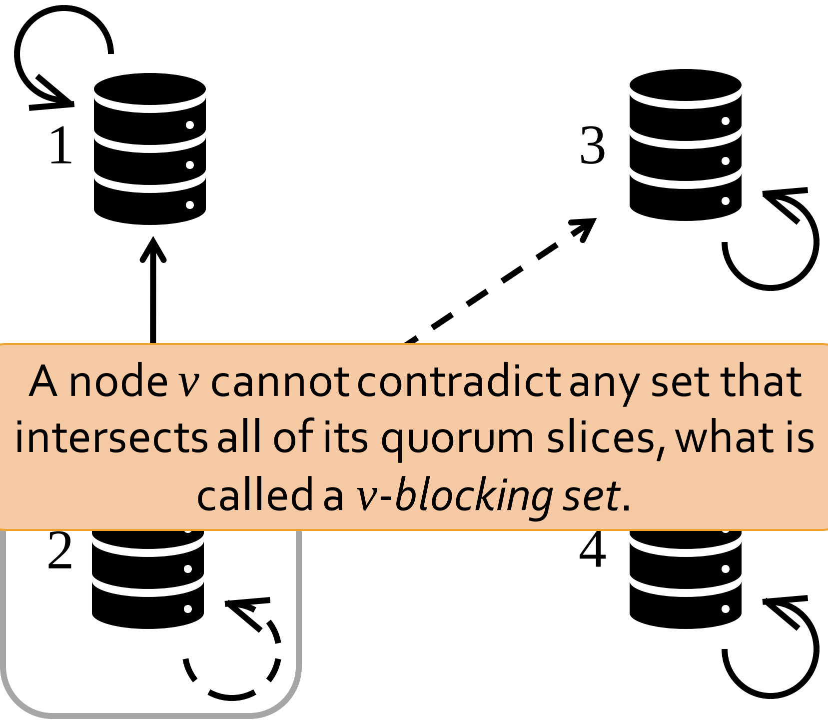


$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$
$U8 = \{2,3,4\} \in \mathbb{Q}$
$U9 = \{1,2,3,4\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$



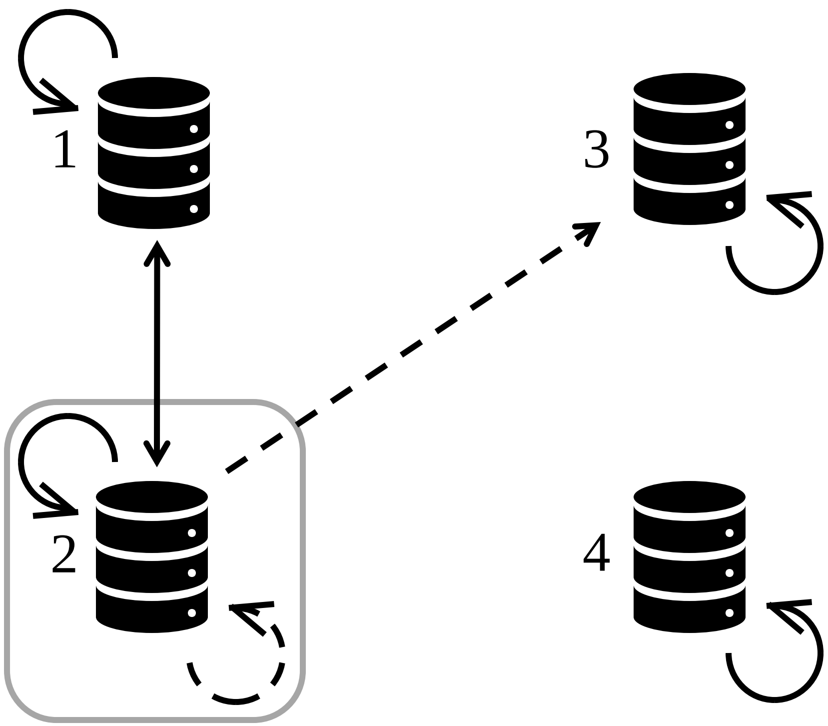
A node *v* cannot contradict any set that intersects all of its quorum slices, what is called a *v-blocking set*.

$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$
$U8 = \{2,3,4\} \in \mathbb{Q}$
$U9 = \{1,2,3,4\} \in \mathbb{Q}$

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$



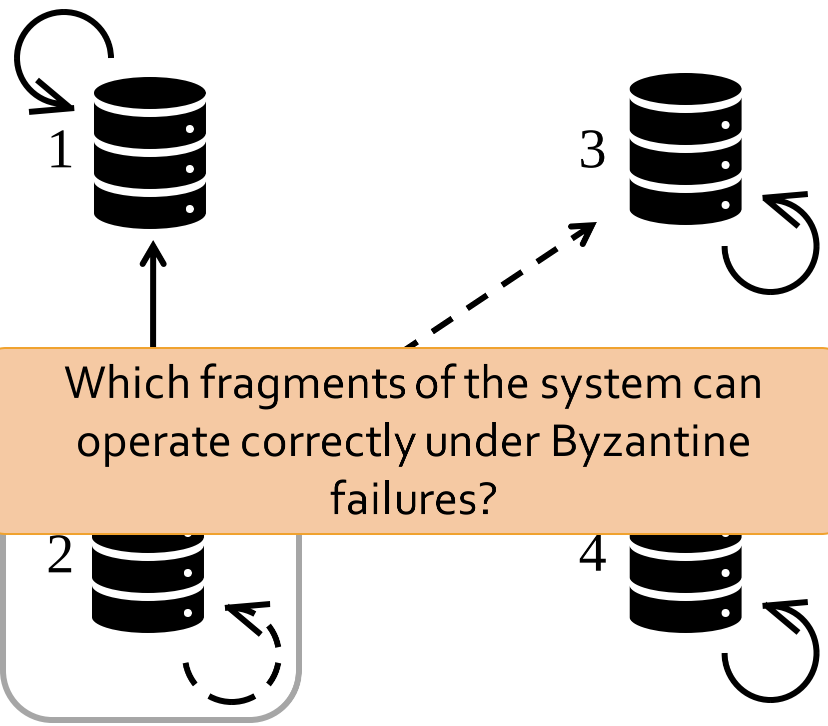
$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$
$U8 = \{2,3,4\} \in \mathbb{Q}$
$U9 = \{1,2,3,4\} \in \mathbb{Q}$

$\{1,3\}$ is 2-blocking

Federated Byzantine quorum systems (FBQS)

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$



Which fragments of the system can operate correctly under Byzantine failures?

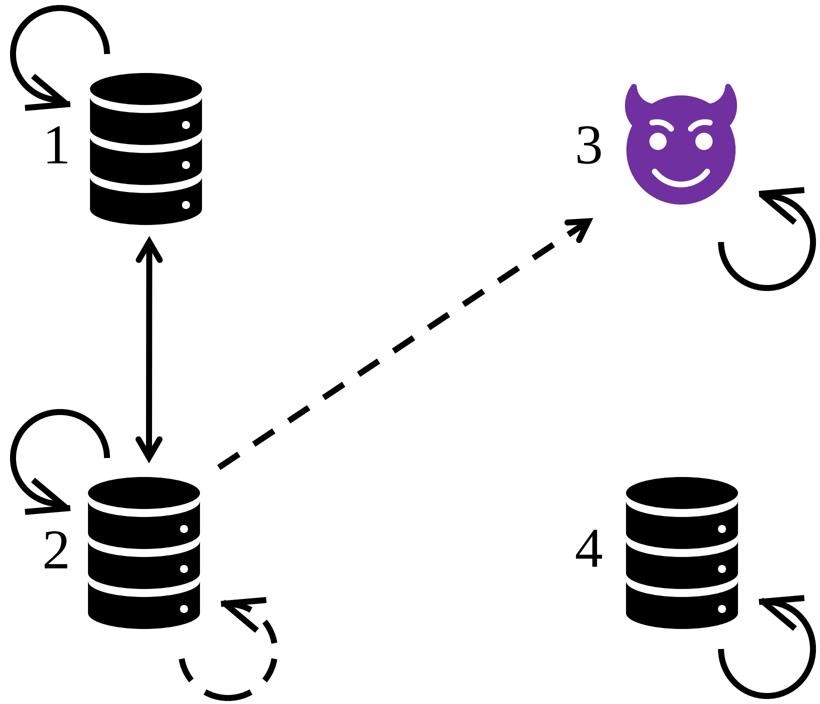
$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$
$U8 = \{2,3,4\} \in \mathbb{Q}$
$U9 = \{1,2,3,4\} \in \mathbb{Q}$

$\{1,3\}$ is 2-blocking

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S} : \mathbb{V} \to 2^{2^{\mathbb{V}}}$
$\mathbb{S}(1) = \{\{1,2\}\}$
$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$
$\mathbb{S}(3) = \{\{3\}\}$
$\mathbb{S}(4) = \{\{4\}\}$



$U1 = \{1,2\} \in \mathbb{Q}$
$U2 = \{2,3\} \in \mathbb{Q}$
$U3 = \{3\} \in \mathbb{Q}$
$U4 = \{4\} \in \mathbb{Q}$
$U5 = \{1,2,3\} \in \mathbb{Q}$
$U6 = \{3,4\} \in \mathbb{Q}$
$U7 = \{1,2,4\} \in \mathbb{Q}$
$U8 = \{2,3,4\} \in \mathbb{Q}$
$U9 = \{1,2,3,4\} \in \mathbb{Q}$

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

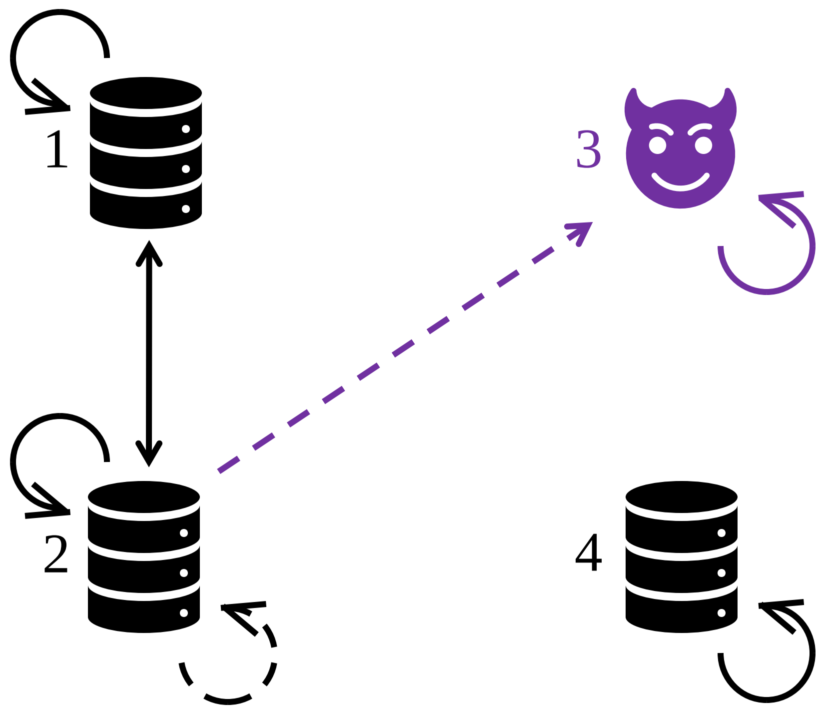
$\mathbb{S} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$

$\mathbb{S}(1) = \{\{1,2\}\}$

$\mathbb{S}(2) = \{\{1,2\},\{2,3\}\}$

$\mathbb{S}(3) = \{\{3\}\}$

$\mathbb{S}(4) = \{\{4\}\}$



$U1 = \{1,2\} \in \mathbb{Q}$

$U2 = \{2,3\} \in \mathbb{Q}$

$U3 = \{3\} \in \mathbb{Q}$

$U4 = \{4\} \in \mathbb{Q}$

$U5 = \{1,2,3\} \in \mathbb{Q}$

$U6 = \{3,4\} \in \mathbb{Q}$

$U7 = \{1,2,4\} \in \mathbb{Q}$

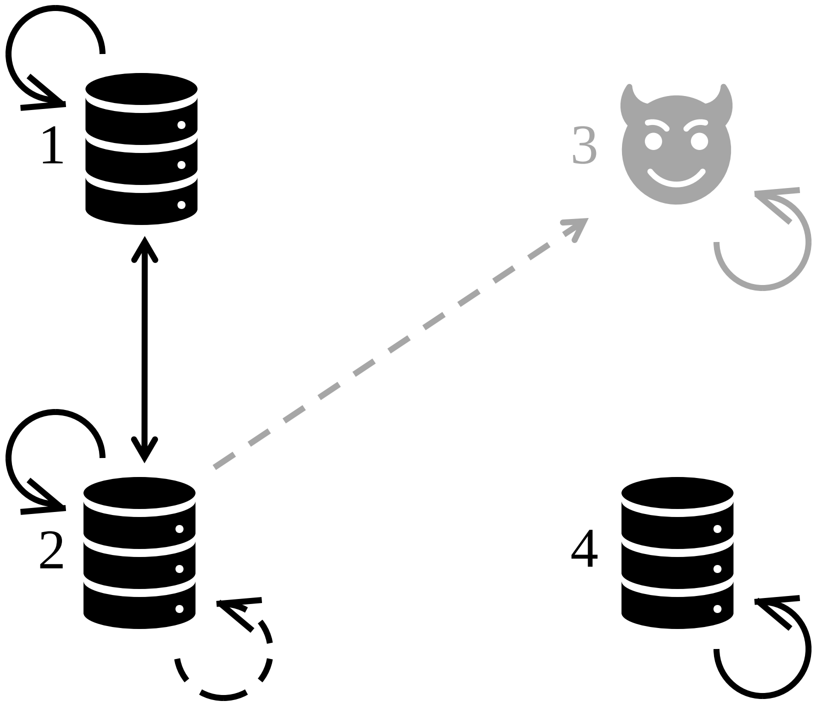
$U8 = \{2,3,4\} \in \mathbb{Q}$

$U9 = \{1,2,3,4\} \in \mathbb{Q}$

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

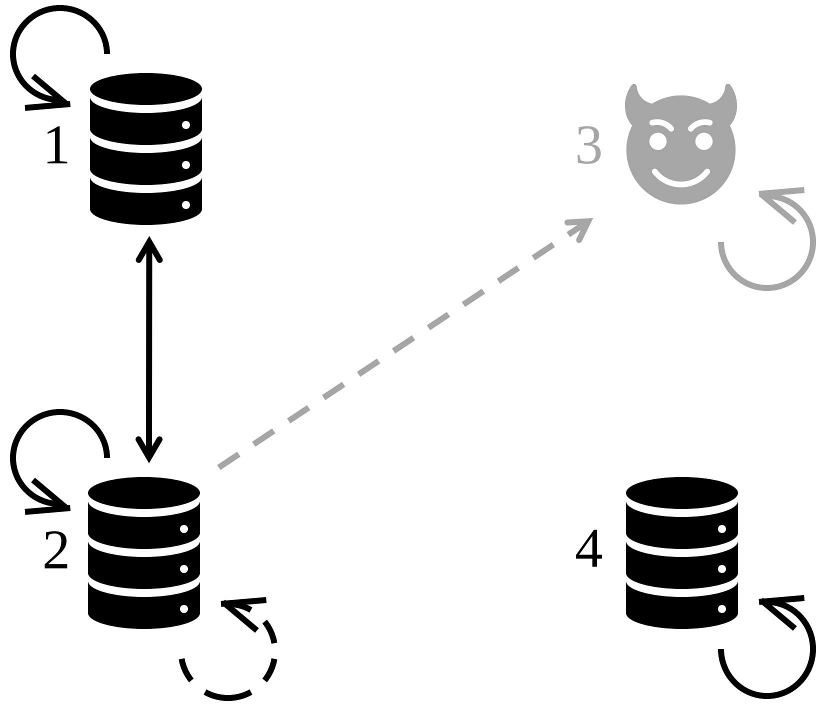
$\mathbb{S}|_{\{1,2,4\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{1,2,4\}}(1) = \{\{1,2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(2) = \{\{1,2\},\{2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(4) = \{\{4\}\}$



Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S}|_{\{1,2,4\}} : \mathbb{V} \to 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{1,2,4\}}(1) = \{\{1,2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(2) = \{\{1,2\},\{2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(4) = \{\{4\}\}$

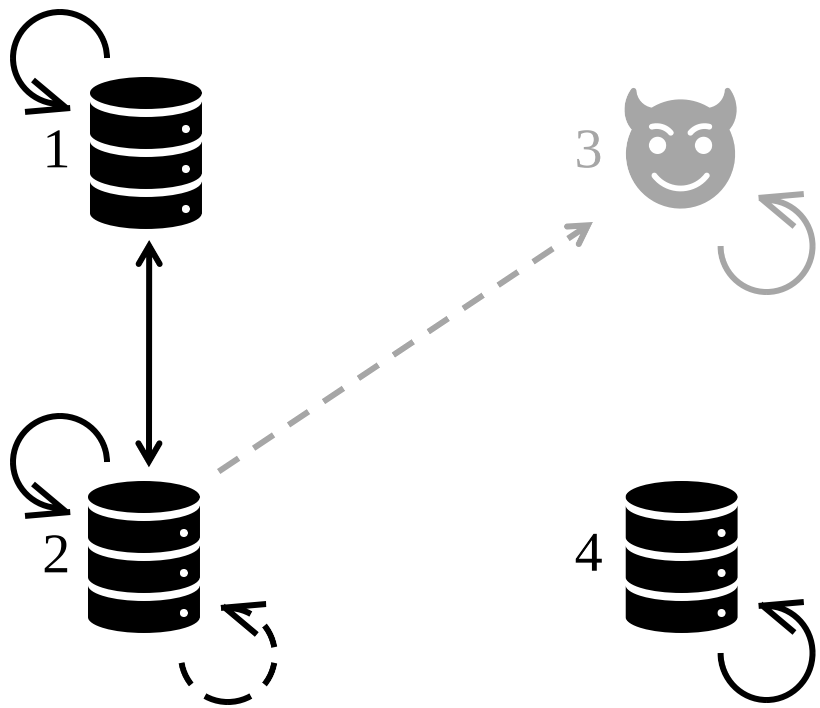


$U1 = \{1,2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U2' = \{2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U4 = \{4\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U7 = \{1,2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S}|_{\{1,2,4\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{1,2,4\}}(1) = \{\{1,2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(2) = \{\{1,2\},\{2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(4) = \{\{4\}\}$



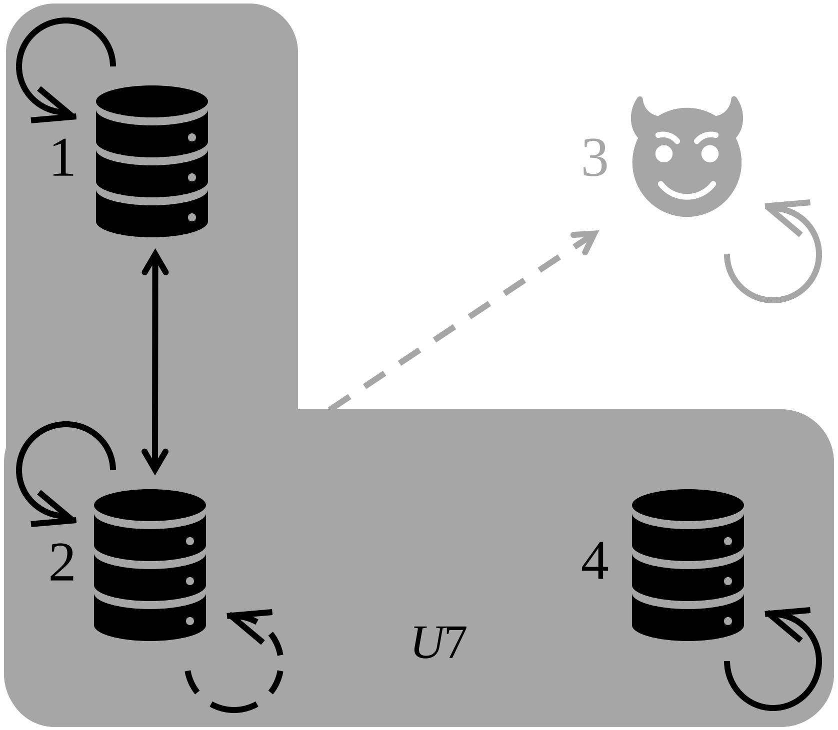
$U1 = \{1,2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U2' = \{2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U4 = \{4\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U7 = \{1,2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$

Liveness requires $\mathbb{V}|_{\{1,2,4\}}$ to enjoy *quorum availability*.

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S}|_{\{1,2,4\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{1,2,4\}}(1) = \{\{1,2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(2) = \{\{1,2\},\{2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(4) = \{\{4\}\}$



$U1 = \{1,2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U2' = \{2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U4 = \{4\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U7 = \{1,2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$

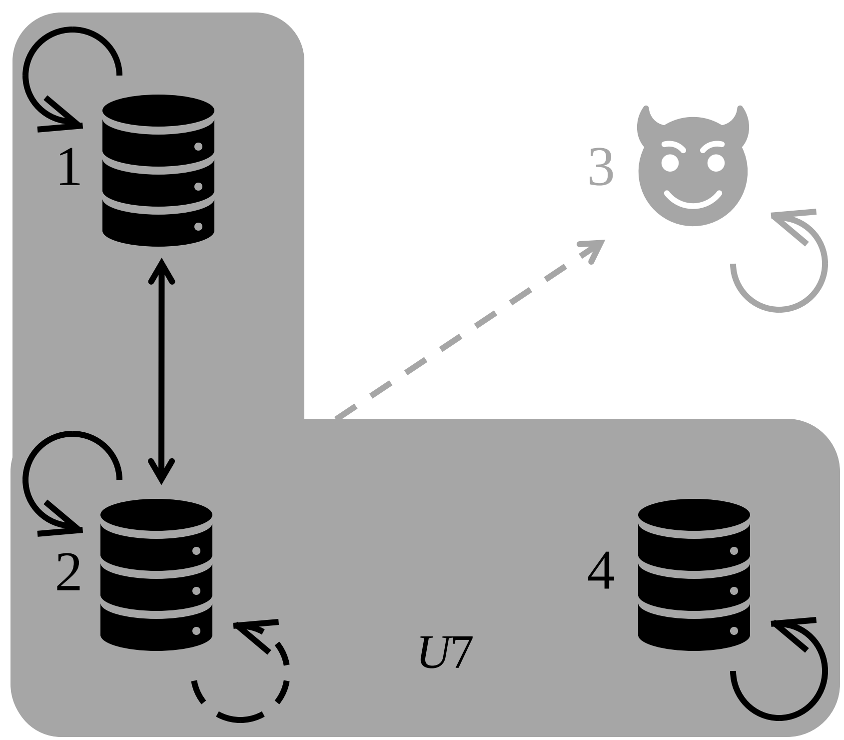
$\mathbb{V}|_{\{1,2,4\}} = U7 \in \mathbb{Q}$ ✓

Liveness requires $\mathbb{V}|_{\{1,2,4\}}$ to enjoy *quorum availability*.

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S}|_{\{1,2,4\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{1,2,4\}}(1) = \{\{1,2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(2) = \{\{1,2\},\{2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(4) = \{\{4\}\}$



$U1 = \{1,2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U2' = \{2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U4 = \{4\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U7 = \{1,2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$

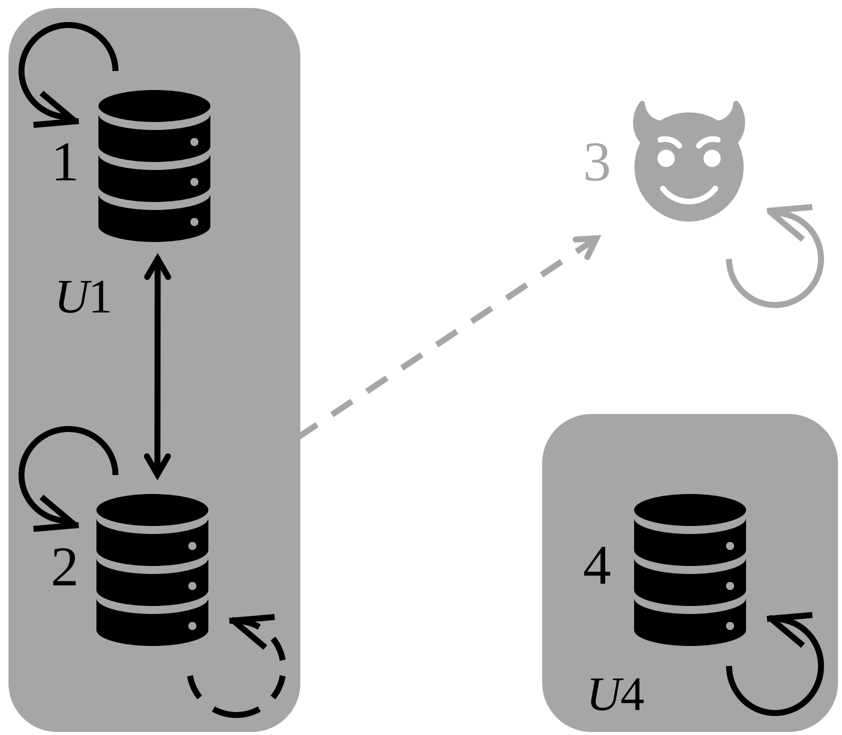
$\mathbb{V}|_{\{1,2,4\}} = U7 \in \mathbb{Q}$ ✓

Safety requires $\mathbb{Q}|_{\{1,2,4\}}$ to enjoy *quorum intersection*.

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S}|_{\{1,2,4\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{1,2,4\}}(1) = \{\{1,2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(2) = \{\{1,2\},\{2\}\}$
$\mathbb{S}|_{\{1,2,4\}}(4) = \{\{4\}\}$



$U1 = \{1,2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U2' = \{2\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U4 = \{4\} \in \mathbb{Q}|_{\{1,2,4\}}$
$U7 = \{1,2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$

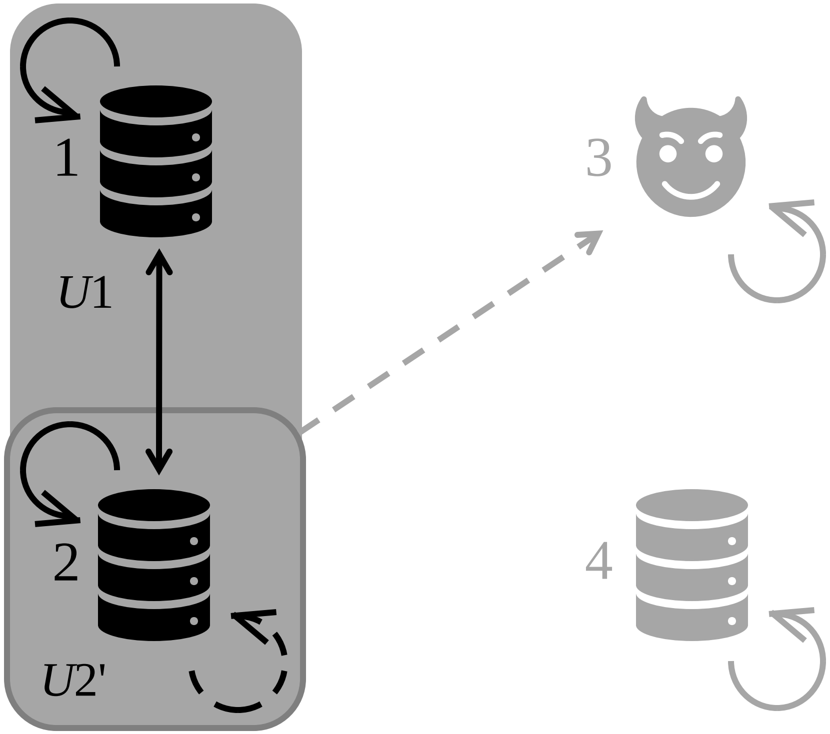
$\mathbb{V}|_{\{1,2,4\}} = U7 \in \mathbb{Q}$ ✓
$U1 \cap U4 = \emptyset$ ✗

Safety requires $\mathbb{Q}|_{\{1,2,4\}}$ to enjoy *quorum intersection*.

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S}|_{\{1,2\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{1,2\}}(1) = \{\{1,2\}\}$
$\mathbb{S}|_{\{1,2\}}(2) = \{\{1,2\},\{2\}\}$



$U1 = \{1,2\} \in \mathbb{Q}|_{\{1,2\}}$
$U2' = \{2\} \in \mathbb{Q}|_{\{1,2\}}$

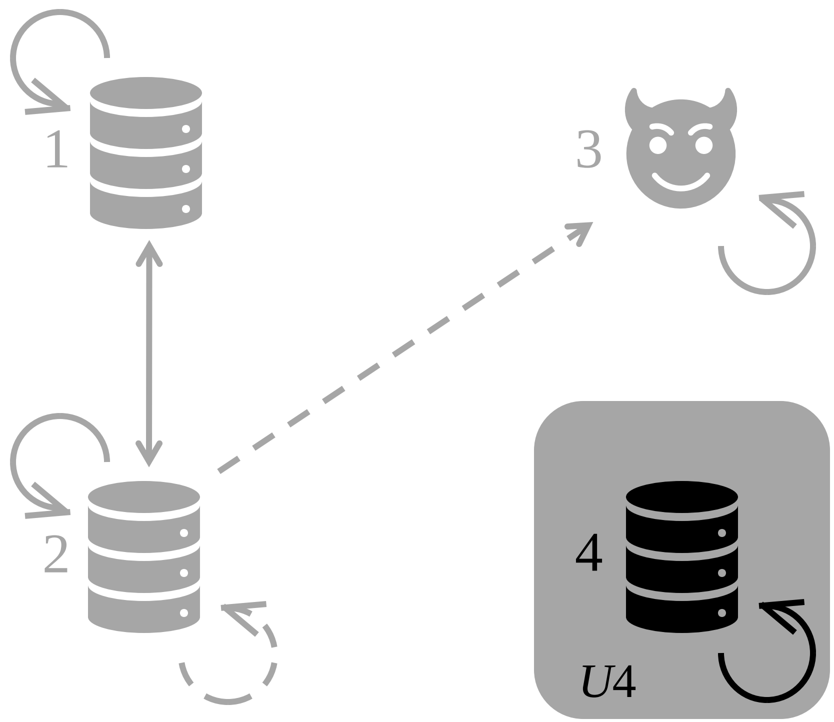
$\mathbb{V}|_{\{1,2\}} = U1 \in \mathbb{Q}$ ✓
$U1 \cap U2' \neq \emptyset$ ✓

{1,2} is an intact set which can operate correctly!

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S}|_{\{4\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{4\}}(4) = \{\{4\}\}$



$U4 = \{4\} \in \mathbb{Q}|_{\{4\}}$

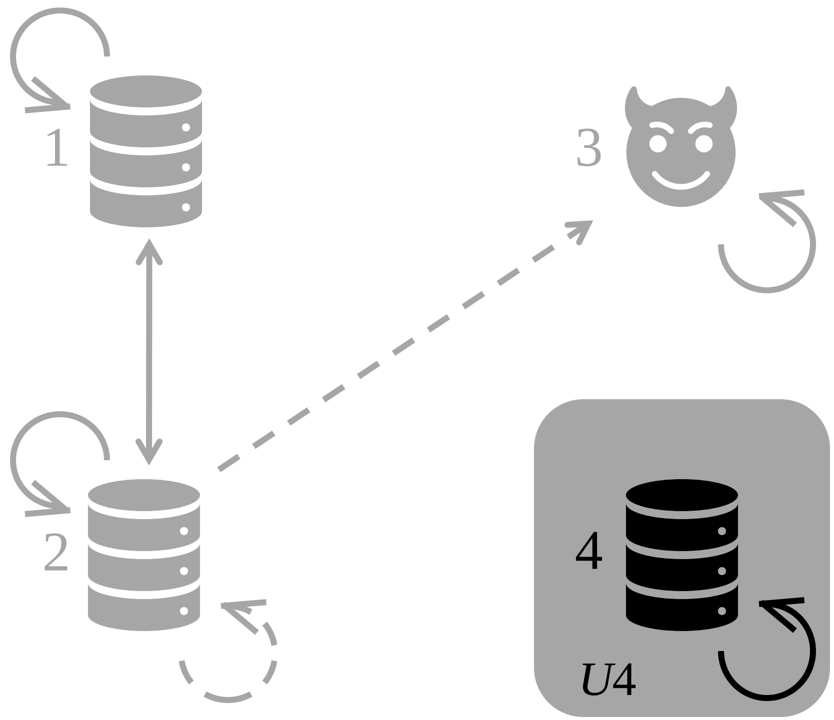
$\mathbb{V}|_{\{4\}} = U4 \in \mathbb{Q}$ ✓
$U4 \cap U4 \neq \emptyset$ ✓

{4} is an intact set which can operate correctly!

Characterising intact sets

$\mathbb{V} = \{1,2,3,4\}$

$\mathbb{S}|_{\{4\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$
$\mathbb{S}|_{\{4\}}(4) = \{\{4\}\}$

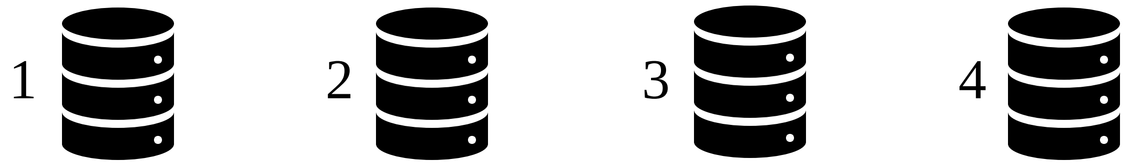


$U4 = \{4\} \in \mathbb{Q}|_{\{4\}}$

$\mathbb{V}|_{\{4\}} = U4 \in \mathbb{Q}$ ✓
$U4 \cap U4 \neq \emptyset$ ✓

Both $\{1,2\}$ and $\{4\}$ are *maximal* intact sets.

Cardinality-based quorum systems $3f + 1$



$$\mathbb{S} : \mathbb{V} \to 2^{2^{\mathbb{V}}}$$
$$\mathbb{S}(1) = \{\{1,2,3\},\{1,2,4\},\{1,3,4\}\}$$
$$\mathbb{S}(2) = \{\{1,2,3\},\{1,2,4\},\{2,3,4\}\}$$
$$\mathbb{S}(3) = \{\{1,2,3\},\{1,3,4\},\{2,3,4\}\}$$
$$\mathbb{S}(4) = \{\{1,2,4\},\{1,3,4\},\{2,3,4\}\}$$

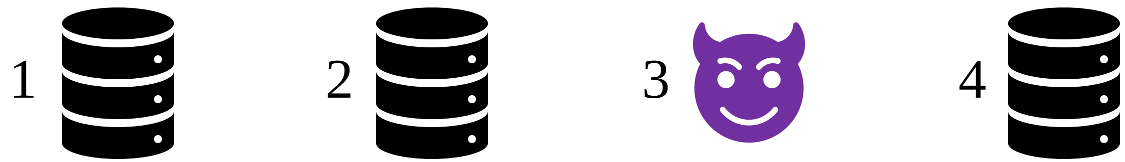
$$U1 = \{1,2,3\} \in \mathbb{Q}$$
$$U2 = \{1,2,4\} \in \mathbb{Q}$$
$$U3 = \{1,3,4\} \in \mathbb{Q}$$
$$U4 = \{2,3,4\} \in \mathbb{Q}$$

Cardinality-based quorum systems $3f + 1$

$$\mathbb{S}|_{\{1,2,4\}} : \mathbb{V} \rightarrow 2^{2^{\mathbb{V}}}$$
$$\mathbb{S}|_{\{1,2,4\}}(1) = \{\{1,2\},\{1,2,4\},\{1,4\}\}$$
$$\mathbb{S}|_{\{1,2,4\}}(2) = \{\{1,2\},\{1,2,4\},\{2,4\}\}$$
$$\mathbb{S}|_{\{1,2,4\}}(4) = \{\{1,2,4\},\{1,4\},\{2,4\}\}$$

$$U1' = \{1,2\} \in \mathbb{Q}|_{\{1,2,4\}}$$
$$U2 = \{1,2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$$
$$U3' = \{1,4\} \in \mathbb{Q}|_{\{1,2,4\}}$$
$$U4' = \{2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$$

Cardinality-based quorum systems $3f + 1$



$\mathbb{S}|_{\{1,2,4\}} : \mathbb{V} \to 2^{2^{\mathbb{V}}}$

$\mathbb{S}|_{\{1,2,4\}}(1) = \{\{1,2\},\{1,2,4\},\{1,4\}\}$

{1,2,4} is the maximal intact set.
Every two correct nodes block the other correct node.

$U1' = \{1,2\} \in \mathbb{Q}|_{\{1,2,4\}}$

$U2 = \{1,2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$

$U3' = \{1,4\} \in \mathbb{Q}|_{\{1,2,4\}}$

$U4' = \{2,4\} \in \mathbb{Q}|_{\{1,2,4\}}$

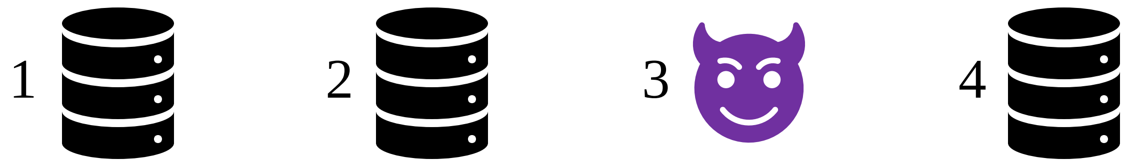
Federating Voting

Federated voting over $3f + 1$

Federated voting over $3f + 1$

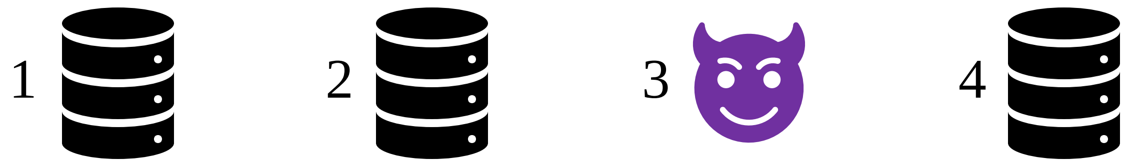
vote(✓)	vote(✓)		vote(✗)

Federated voting over $3f + 1$



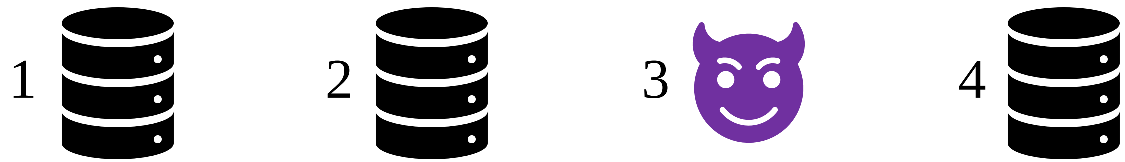
vote(✓)	vote(✓)		vote(✗)
VOTE(✓)	VOTE(✓)		VOTE(✗)

Federated voting over $3f + 1$



vote(✓)	vote(✓)		vote(✗)
VOTE(✓)	VOTE(✓)	VOTE(✓)	VOTE(✗)

Federated voting over $3f + 1$



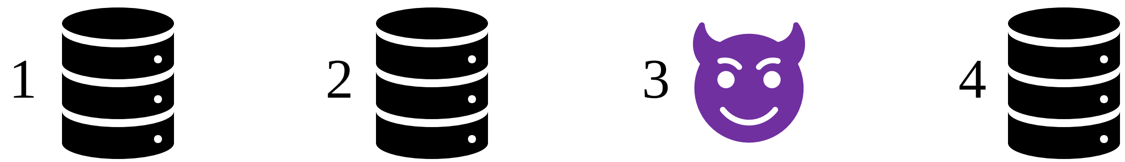
vote(✓)	vote(✓)		vote(✗)
VOTE(✓)	VOTE(✓)	VOTE(✓)	VOTE(✗)
READY(✓)	READY(✓)		

Federated voting over $3f + 1$

vote(✓)	vote(✓)		vote(✗)
VOTE(✓)	VOTE(✓)	VOTE(✓)	VOTE(✗)
READY(✓)	READY(✓)		
			READY(✓)

$\{1,2\}$ is a 4-blocking set.

Federated voting over $3f + 1$



vote(✓)	vote(✓)		vote(✗)
VOTE(✓)	VOTE(✓)	VOTE(✓)	VOTE(✗)
READY(✓)	READY(✓)		
			READY(✓)
deliver(✓)	deliver(✓)		deliver(✓)

Federated voting over $3f + 1$

vote(✓)	vote(✓)		vote(✗)
VOTE(✓)	VOTE(✓)	VOTE(✓)	VOTE(✗)
READY(✓)	READY(✓)		
			READY(✓)
deliver(✓)	deliver(✓)		deliver(✓)

Node v can compute quorums to which v belongs and v-blocking sets with only local information!

Guarantess of federated voting

Federated voting ensures propeties similar to those of Bracha broadcast [Bra87], but relative to intact sets.

Given a maximal intact set I:

Safety

(*Consistency*) No two nodes in I deliver different values.

Liveness

(*Totality*) If a node in I delivers a value, then every node in I eventually delivers a value.

Stellar Consensus Protocol (SCP)

Ballots

- Ballots from *Paxos* [Lam98] to neutralise stuck values:

  A *ballot* $\langle n, \boldsymbol{x} \rangle$ attaches a *round counter* $n \in \mathbb{N}^{+}$ to the value $\boldsymbol{x}$.

Ballots

- Ballots from *Paxos* [Lam98] to neutralise stuck values:

  A *ballot* $\langle n, \boldsymbol{x}\rangle$ attaches a *round counter* $n \in \mathbb{N}^+$ to the value $\boldsymbol{x}$.

- Ballots are alphabetically ordered on their counter and value:

  The special *null ballot* $\langle 0, \perp\rangle$ is below any other ballot.

Ballots

- Ballots from *Paxos* [Lam98] to neutralise stuck values:

  A *ballot* $\langle n, \boldsymbol{x} \rangle$ attaches a *round counter* $n \in \mathbb{N}^{+}$ to the value $\boldsymbol{x}$.

- Ballots are alphabetically ordered on their counter and value:

  The special *null ballot* $\langle 0, \perp \rangle$ is below any other ballot.

- *Less and incompatible than* relation:

  $\langle n, \boldsymbol{x} \rangle \lessapprox \langle m, \boldsymbol{y} \rangle$ iff $\langle n, \boldsymbol{x} \rangle < \langle m, \boldsymbol{y} \rangle$ and $\boldsymbol{x} \neq \boldsymbol{y}$.

Stages of SCP

Each node considers a *candidate ballot* $b = \langle n, \boldsymbol{x} \rangle$ and proceeds in two stages:

Prepare stage: try to *abort* every ballot $b' \lessapprox b$, i.e., vote ✗ on every $b' \lessapprox b$.

Commit stage: once b is prepared, try to *commit* b, i.e., vote ✓ on b.

Stages of SCP

Each node considers a *candidate ballot* $b = \langle n, \boldsymbol{x} \rangle$ and proceeds in two stages:

Prepare stage: try to *abort* every ballot $b' \lessapprox b$, i.e., vote ✗ on every $b' \lessapprox b$.

Commit stage: once b is prepared, try to *commit* b, i.e., vote ✓ on b.

In order to ensure liveness:

- Start a timer after receiving a quorum of messages with a new round n.
- After timeout, take as candidate ballot the highest ballot prepared so far with round increased by one, and retry prepare and commit stages.

SCP over $3f + 1$

SCP over $3f + 1$

1 cand:$\langle 0, \perp \rangle$
prep:$\langle 0, \perp \rangle$

2 cand:$\langle 0, \perp \rangle$
prep:$\langle 0, \perp \rangle$

3

4 cand:$\langle 0, \perp \rangle$
prep:$\langle 0, \perp \rangle$

SCP over $3f+1$

1 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 0,\perp\rangle$	2 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 0,\perp\rangle$	3	4 cand: $\langle 1,\mathbf{1}\rangle$ prep: $\langle 0,\perp\rangle$
propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)

SCP over $3f+1$

1 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 0,\perp\rangle$

2 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 0,\perp\rangle$

3

4 cand: $\langle 1,\mathbf{1}\rangle$ prep: $\langle 0,\perp\rangle$

propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)
VOTE(X,$\langle 0,\perp\rangle$) VOTE(X,$\langle 1,\mathbf{1}\rangle$) VOTE($X$,$\langle 1,\mathbf{2}\rangle$)	VOTE(X,$\langle 0,\perp\rangle$) VOTE(X,$\langle 1,\mathbf{1}\rangle$) VOTE($X$,$\langle 1,\mathbf{2}\rangle$)	VOTE(X,$\langle 0,\perp\rangle$) VOTE(X,$\langle 1,\mathbf{1}\rangle$)	VOTE(X,$\langle 0,\perp\rangle$)

SCP over $3f+1$

1 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 0,\perp\rangle$

2 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 0,\perp\rangle$

3

4 cand: $\langle 1,\mathbf{1}\rangle$ prep: $\langle 0,\perp\rangle$

propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)
VOTE(X,$\langle 0,\perp\rangle$) VOTE(X,$\langle 1,\mathbf{1}\rangle$) VOTE($X$,$\langle 1,\mathbf{2}\rangle$)	VOTE(X,$\langle 0,\perp\rangle$) VOTE(X,$\langle 1,\mathbf{1}\rangle$) VOTE($X$,$\langle 1,\mathbf{2}\rangle$)	VOTE(X,$\langle 0,\perp\rangle$) VOTE(X,$\langle 1,\mathbf{1}\rangle$)	VOTE(X,$\langle 0,\perp\rangle$)
start-timer($F(1)$)	start-timer($F(1)$)		start-timer($F(1)$)

SCP over $3f + 1$

1 cand: $\langle 1, \mathbf{3} \rangle$ prep: $\langle 0, \perp \rangle$

2 cand: $\langle 1, \mathbf{3} \rangle$ prep: $\langle 0, \perp \rangle$

3

4 cand: $\langle 1, \mathbf{1} \rangle$ prep: $\langle 0, \perp \rangle$

propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)
VOTE(X,$\langle 0,\perp \rangle$) VOTE(X,$\langle 1,\mathbf{1} \rangle$) VOTE($X$,$\langle 1,\mathbf{2} \rangle$)	VOTE(X,$\langle 0,\perp \rangle$) VOTE(X,$\langle 1,\mathbf{1} \rangle$) VOTE($X$,$\langle 1,\mathbf{2} \rangle$)	VOTE(X,$\langle 0,\perp \rangle$) VOTE(X,$\langle 1,\mathbf{1} \rangle$)	VOTE(X,$\langle 0,\perp \rangle$)
start-timer($F(1)$)	start-timer($F(1)$)		start-timer($F(1)$)
READY(X,$\langle 0,\perp \rangle$) READY(X,$\langle 1,\mathbf{1} \rangle$)	READY(X,$\langle 0,\perp \rangle$) READY(X,$\langle 1,\mathbf{1} \rangle$)		READY(X,$\langle 0,\perp \rangle$)

SCP over $3f+1$

1 cand:$\langle 1,\mathbf{3}\rangle$ prep:$\langle 1,\mathbf{1}\rangle$

2 cand:$\langle 1,\mathbf{3}\rangle$ prep:$\langle 1,\mathbf{1}\rangle$

3

4 cand:$\langle 1,\mathbf{1}\rangle$ prep:$\langle 1,\mathbf{1}\rangle$

propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)
$\texttt{VOTE}(X,\langle 0,\perp\rangle)$ $\texttt{VOTE}(X,\langle 1,\mathbf{1}\rangle)$ $\texttt{VOTE}(X,\langle 1,\mathbf{2}\rangle)$	$\texttt{VOTE}(X,\langle 0,\perp\rangle)$ $\texttt{VOTE}(X,\langle 1,\mathbf{1}\rangle)$ $\texttt{VOTE}(X,\langle 1,\mathbf{2}\rangle)$	$\texttt{VOTE}(X,\langle 0,\perp\rangle)$ $\texttt{VOTE}(X,\langle 1,\mathbf{1}\rangle)$	$\texttt{VOTE}(X,\langle 0,\perp\rangle)$
start-timer($F(1)$)	start-timer($F(1)$)		start-timer($F(1)$)
$\texttt{READY}(X,\langle 0,\perp\rangle)$ $\texttt{READY}(X,\langle 1,\mathbf{1}\rangle)$	$\texttt{READY}(X,\langle 0,\perp\rangle)$ $\texttt{READY}(X,\langle 1,\mathbf{1}\rangle)$		$\texttt{READY}(X,\langle 0,\perp\rangle)$
prepared($\langle 1,\mathbf{1}\rangle$)	prepared($\langle 1,\mathbf{1}\rangle$)		prepared($\langle 1,\mathbf{1}\rangle$)

SCP over $3f + 1$

1 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 1,\mathbf{1}\rangle$

2 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 1,\mathbf{1}\rangle$

3

4 cand: $\langle 1,\mathbf{1}\rangle$ prep: $\langle 1,\mathbf{1}\rangle$

propose(**3**)	propose(**3**)		propose(**1**)
VOTE($\times$,$\langle 0,\perp\rangle$) VOTE($\times$,$\langle 1,\mathbf{1}\rangle$) VOTE($\times$,$\langle 1,\mathbf{2}\rangle$)	VOTE($\times$,$\langle 0,\perp\rangle$) VOTE($\times$,$\langle 1,\mathbf{1}\rangle$) VOTE($\times$,$\langle 1,\mathbf{2}\rangle$)	VOTE($\times$,$\langle 0,\perp\rangle$) VOTE($\times$,$\langle 1,\mathbf{1}\rangle$)	VOTE($\times$,$\langle 0,\perp\rangle$)
start-timer($F(1)$)	start-timer($F(1)$)		start-timer($F(1)$)
READY($\times$,$\langle 0,\perp\rangle$) READY($\times$,$\langle 1,\mathbf{1}\rangle$)	READY($\times$,$\langle 0,\perp\rangle$) READY($\times$,$\langle 1,\mathbf{1}\rangle$)		READY($\times$,$\langle 0,\perp\rangle$)
prepared($\langle 1,\mathbf{1}\rangle$)	prepared($\langle 1,\mathbf{1}\rangle$)		prepared($\langle 1,\mathbf{1}\rangle$)
			VOTE($\checkmark$,$\langle 1,\mathbf{1}\rangle$)

SCP over $3f+1$

1 cand:$\langle 1,\mathbf{3}\rangle$ prep:$\langle 1,\mathbf{1}\rangle$

2 cand:$\langle 1,\mathbf{3}\rangle$ prep:$\langle 1,\mathbf{1}\rangle$

3

4 cand:$\langle 1,\mathbf{1}\rangle$ prep:$\langle 1,\mathbf{1}\rangle$

propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)
$\texttt{VOTE}(✗,\langle 0,\perp\rangle)$ $\texttt{VOTE}(✗,\langle 1,\mathbf{1}\rangle)$ $\texttt{VOTE}(✗,\langle 1,\mathbf{2}\rangle)$	$\texttt{VOTE}(✗,\langle 0,\perp\rangle)$ $\texttt{VOTE}(✗,\langle 1,\mathbf{1}\rangle)$ $\texttt{VOTE}(✗,\langle 1,\mathbf{2}\rangle)$	$\texttt{VOTE}(✗,\langle 0,\perp\rangle)$ $\texttt{VOTE}(✗,\langle 1,\mathbf{1}\rangle)$	$\texttt{VOTE}(✗,\langle 0,\perp\rangle)$
start-timer($F(1)$)	start-timer($F(1)$)		start-timer($F(1)$)
$\texttt{READY}(✗,\langle 0,\perp\rangle)$ $\texttt{READY}(✗,\langle 1,\mathbf{1}\rangle)$	$\texttt{READY}(✗,\langle 0,\perp\rangle)$ $\texttt{READY}(✗,\langle 1,\mathbf{1}\rangle)$		$\texttt{READY}(✗,\langle 0,\perp\rangle)$
prepared($\langle 1,\mathbf{1}\rangle$)	prepared($\langle 1,\mathbf{1}\rangle$)		prepared($\langle 1,\mathbf{1}\rangle$)
			$\texttt{VOTE}(✓,\langle 1,\mathbf{1}\rangle)$
			$\texttt{READY}(✗,\langle 1,\mathbf{1}\rangle)$

SCP over $3f + 1$

1 cand: $\langle 1, \mathbf{3} \rangle$ prep: $\langle 1, \mathbf{2} \rangle$

2 cand: $\langle 1, \mathbf{3} \rangle$ prep: $\langle 1, \mathbf{2} \rangle$

3

4 cand: $\langle 1, \mathbf{1} \rangle$ prep: $\langle 1, \mathbf{2} \rangle$

propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)
VOTE(✗,$\langle 0,\perp \rangle$) VOTE(✗,$\langle 1,\mathbf{1} \rangle$) VOTE(✗,$\langle 1,\mathbf{2} \rangle$)	VOTE(✗,$\langle 0,\perp \rangle$) VOTE(✗,$\langle 1,\mathbf{1} \rangle$) VOTE(✗,$\langle 1,\mathbf{2} \rangle$)	VOTE(✗,$\langle 0,\perp \rangle$) VOTE(✗,$\langle 1,\mathbf{1} \rangle$)	VOTE(✗,$\langle 0,\perp \rangle$)
start-timer($F(1)$)	start-timer($F(1)$)		start-timer($F(1)$)
READY(✗,$\langle 0,\perp \rangle$) READY(✗,$\langle 1,\mathbf{1} \rangle$)	READY(✗,$\langle 0,\perp \rangle$) READY(✗,$\langle 1,\mathbf{1} \rangle$)		READY(✗,$\langle 0,\perp \rangle$)
prepared($\langle 1,\mathbf{1} \rangle$)	prepared($\langle 1,\mathbf{1} \rangle$)		prepared($\langle 1,\mathbf{1} \rangle$)
			VOTE(✓,$\langle 1,\mathbf{1} \rangle$)
			READY(✗,$\langle 1,\mathbf{1} \rangle$)
prepared($\langle 1,\mathbf{2} \rangle$)	prepared($\langle 1,\mathbf{2} \rangle$)		prepared($\langle 1,\mathbf{2} \rangle$)

SCP over $3f+1$

1 cand:$\langle 1,\mathbf{3}\rangle$ prep:$\langle 1,\mathbf{2}\rangle$

2 cand:$\langle 1,\mathbf{3}\rangle$ prep:$\langle 1,\mathbf{2}\rangle$

3

4 cand:$\langle 1,\mathbf{2}\rangle$ prep:$\langle 1,\mathbf{2}\rangle$

propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)
VOTE(✗,$\langle 0,\perp\rangle$) VOTE(✗,$\langle 1,\mathbf{1}\rangle$) VOTE(✗,$\langle 1,\mathbf{2}\rangle$)	VOTE(✗,$\langle 0,\perp\rangle$) VOTE(✗,$\langle 1,\mathbf{1}\rangle$) VOTE(✗,$\langle 1,\mathbf{2}\rangle$)	VOTE(✗,$\langle 0,\perp\rangle$) VOTE(✗,$\langle 1,\mathbf{1}\rangle$)	VOTE(✗,$\langle 0,\perp\rangle$)
start-timer($F(1)$)	start-timer($F(1)$)		start-timer($F(1)$)
READY(✗,$\langle 0,\perp\rangle$) READY(✗,$\langle 1,\mathbf{1}\rangle$)	READY(✗,$\langle 0,\perp\rangle$) READY(✗,$\langle 1,\mathbf{1}\rangle$)		READY(✗,$\langle 0,\perp\rangle$)
prepared($\langle 1,\mathbf{1}\rangle$)	prepared($\langle 1,\mathbf{1}\rangle$)		prepared($\langle 1,\mathbf{1}\rangle$)
			VOTE(✓,$\langle 1,\mathbf{1}\rangle$)
			READY(✗,$\langle 1,\mathbf{1}\rangle$)
prepared($\langle 1,\mathbf{2}\rangle$)	prepared($\langle 1,\mathbf{2}\rangle$)		prepared($\langle 1,\mathbf{2}\rangle$)
			VOTE(✓,$\langle 1,\mathbf{2}\rangle$)

SCP over $3f+1$

1 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 1,\mathbf{2}\rangle$

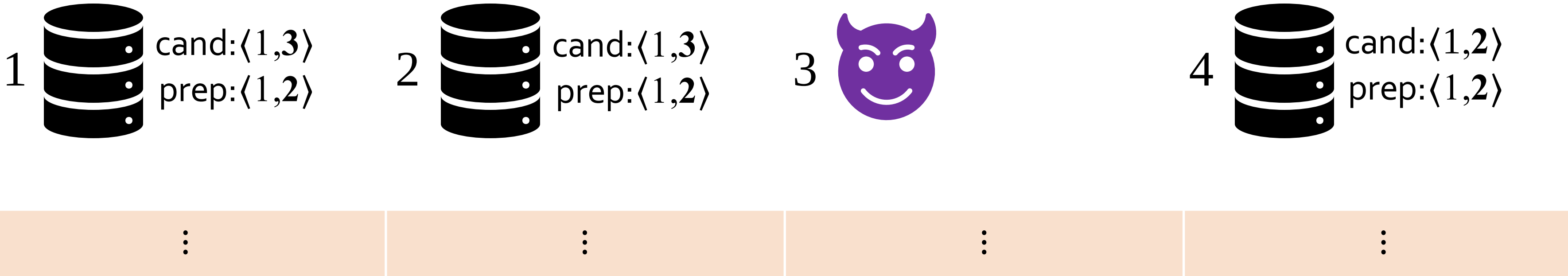
2 cand: $\langle 1,\mathbf{3}\rangle$ prep: $\langle 1,\mathbf{2}\rangle$

3

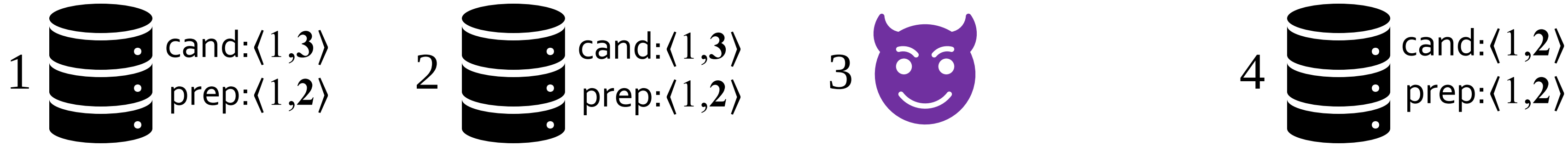
4 cand: $\langle 1,\mathbf{2}\rangle$ prep: $\langle 1,\mathbf{2}\rangle$

propose($\mathbf{3}$)	propose($\mathbf{3}$)		propose($\mathbf{1}$)
VOTE(✗,$\langle 0,\bot\rangle$) VOTE(✗,$\langle 1,\mathbf{1}\rangle$) VOTE(✗,$\langle 1,\mathbf{2}\rangle$)	VOTE(✗,$\langle 0,\bot\rangle$) VOTE(✗,$\langle 1,\mathbf{1}\rangle$) VOTE(✗,$\langle 1,\mathbf{2}\rangle$)	VOTE(✗,$\langle 0,\bot\rangle$) VOTE(✗,$\langle 1,\mathbf{1}\rangle$)	VOTE(✗,$\langle 0,\bot\rangle$)
start-timer($F(1)$)	start-timer($F(1)$)		start-timer($F(1)$)
READY(✗,$\langle 0,\bot\rangle$) READY(✗,$\langle 1,\mathbf{1}\rangle$)	READY(✗,$\langle 0,\bot\rangle$) READY(✗,$\langle 1,\mathbf{1}\rangle$)		READY(✗,$\langle 0,\bot\rangle$)
prepared($\langle 1,\mathbf{1}\rangle$)	prepared($\langle 1,\mathbf{1}\rangle$)		prepared($\langle 1,\mathbf{1}\rangle$)
			VOTE(✓,$\langle 1,\mathbf{1}\rangle$)
			READY(✗,$\langle 1,\mathbf{1}\rangle$)
prepared($\langle 1,\mathbf{2}\rangle$)	prepared($\langle 1,\mathbf{2}\rangle$)		prepared($\langle 1,\mathbf{2}\rangle$)
			VOTE(✓,$\langle 1,\mathbf{2}\rangle$)
⋮	⋮	⋮	⋮

SCP over $3f + 1$


1
cand:$\langle 1,\mathbf{3} \rangle$
prep:$\langle 1,\mathbf{2} \rangle$
2
cand:$\langle 1,\mathbf{3} \rangle$
prep:$\langle 1,\mathbf{2} \rangle$
3
4
cand:$\langle 1,\mathbf{2} \rangle$
prep:$\langle 1,\mathbf{2} \rangle$
⋮
⋮
⋮
⋮

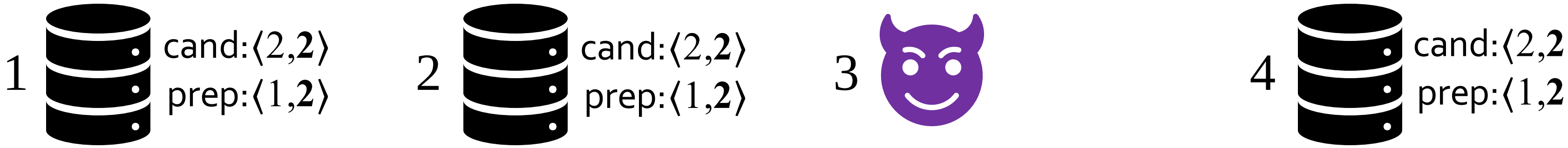
SCP over $3f + 1$



⋮	⋮	⋮	⋮
timeout	timeout		timeout

Eventually the timer for round 1 of each node will timeout...

SCP over $3f+1$



$\vdots$	$\vdots$	$\vdots$	$\vdots$
timeout	timeout		timeout

Eventually the timer for round 1 of each node will timeout…

SCP over $3f+1$

1 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 1,\mathbf{2}\rangle$

2 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 1,\mathbf{2}\rangle$

3

4 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 1,\mathbf{2}\rangle$

⋮	⋮	⋮	⋮
timeout	timeout		timeout
VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)	VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)		VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)

SCP over $3f + 1$

1 cand: $\langle 2, \mathbf{2} \rangle$ prep: $\langle 1, \mathbf{2} \rangle$

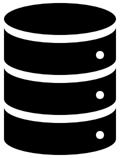
2 cand: $\langle 2, \mathbf{2} \rangle$ prep: $\langle 1, \mathbf{2} \rangle$

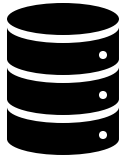
3

4 cand: $\langle 2, \mathbf{2} \rangle$ prep: $\langle 1, \mathbf{2} \rangle$

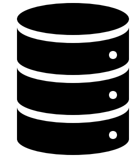
⋮	⋮	⋮	⋮
timeout	timeout		timeout
VOTE(X, $\langle 1, \mathbf{3} \rangle$) … VOTE($X$, $\langle 2, \mathbf{1} \rangle$)	VOTE(X, $\langle 1, \mathbf{3} \rangle$) … VOTE($X$, $\langle 2, \mathbf{1} \rangle$)		VOTE(X, $\langle 1, \mathbf{3} \rangle$) … VOTE($X$, $\langle 2, \mathbf{1} \rangle$)
start-timer($F(2)$)	start-timer($F(2)$)		start-timer($F(2)$)

SCP over $3f + 1$

1  cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 1,\mathbf{2}\rangle$

2  cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 1,\mathbf{2}\rangle$

3 

4  cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 1,\mathbf{2}\rangle$

⋮	⋮	⋮	⋮
timeout	timeout		timeout
VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)	VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)		VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)
start-timer($F(2)$)	start-timer($F(2)$)		start-timer($F(2)$)
READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)	READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)		READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)

SCP over $3f + 1$

1 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

2 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

3

4 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

⋮	⋮	⋮	⋮
timeout	timeout		timeout
VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)	VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)		VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)
start-timer($F(2)$)	start-timer($F(2)$)		start-timer($F(2)$)
READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)	READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)		READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)
prepared($\langle 2,\mathbf{2}\rangle$)	prepared($\langle 2,\mathbf{2}\rangle$)		prepared($\langle 2,\mathbf{2}\rangle$)

SCP over $3f + 1$

1 cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 2,\mathbf{2}\rangle$

2 cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 2,\mathbf{2}\rangle$

3

4 cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 2,\mathbf{2}\rangle$

⋮	⋮	⋮	⋮
timeout	timeout		timeout
VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)	VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)		VOTE(X,$\langle 1,\mathbf{3}\rangle$) … VOTE($X$,$\langle 2,\mathbf{1}\rangle$)
start-timer($F(2)$)	start-timer($F(2)$)		start-timer($F(2)$)
READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)	READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)		READY(X,$\langle 1,\mathbf{3}\rangle$) … READY($X$,$\langle 2,\mathbf{1}\rangle$)
prepared($\langle 2,\mathbf{2}\rangle$)	prepared($\langle 2,\mathbf{2}\rangle$)		prepared($\langle 2,\mathbf{2}\rangle$)
VOTE(✓,$\langle 2,\mathbf{2}\rangle$)	VOTE(✓,$\langle 2,\mathbf{2}\rangle$)		VOTE(✓,$\langle 2,\mathbf{2}\rangle$)

SCP over $3f + 1$

1 cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 2,\mathbf{2}\rangle$

2 cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 2,\mathbf{2}\rangle$

3

4 cand:$\langle 2,\mathbf{2}\rangle$ prep:$\langle 2,\mathbf{2}\rangle$

⋮	⋮	⋮	⋮
timeout	timeout		timeout
VOTE($\times$,$\langle 1,\mathbf{3}\rangle$) … VOTE($\times$,$\langle 2,\mathbf{1}\rangle$)	VOTE($\times$,$\langle 1,\mathbf{3}\rangle$) … VOTE($\times$,$\langle 2,\mathbf{1}\rangle$)		VOTE($\times$,$\langle 1,\mathbf{3}\rangle$) … VOTE($\times$,$\langle 2,\mathbf{1}\rangle$)
start-timer($F(2)$)	start-timer($F(2)$)		start-timer($F(2)$)
READY($\times$,$\langle 1,\mathbf{3}\rangle$) … READY($\times$,$\langle 2,\mathbf{1}\rangle$)	READY($\times$,$\langle 1,\mathbf{3}\rangle$) … READY($\times$,$\langle 2,\mathbf{1}\rangle$)		READY($\times$,$\langle 1,\mathbf{3}\rangle$) … READY($\times$,$\langle 2,\mathbf{1}\rangle$)
prepared($\langle 2,\mathbf{2}\rangle$)	prepared($\langle 2,\mathbf{2}\rangle$)		prepared($\langle 2,\mathbf{2}\rangle$)
VOTE($\checkmark$,$\langle 2,\mathbf{2}\rangle$)	VOTE($\checkmark$,$\langle 2,\mathbf{2}\rangle$)		VOTE($\checkmark$,$\langle 2,\mathbf{2}\rangle$)
READY($\checkmark$,$\langle 2,\mathbf{2}\rangle$)	READY($\checkmark$,$\langle 2,\mathbf{2}\rangle$)		READY($\checkmark$,$\langle 2,\mathbf{2}\rangle$)

SCP over $3f + 1$

1 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

2 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

3

4 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

⋮	⋮	⋮	⋮
timeout	timeout		timeout
VOTE(✗, $\langle 1,\mathbf{3}\rangle$) ... VOTE(✗, $\langle 2,\mathbf{1}\rangle$)	VOTE(✗, $\langle 1,\mathbf{3}\rangle$) ... VOTE(✗, $\langle 2,\mathbf{1}\rangle$)		VOTE(✗, $\langle 1,\mathbf{3}\rangle$) ... VOTE(✗, $\langle 2,\mathbf{1}\rangle$)
start-timer($F(2)$)	start-timer($F(2)$)		start-timer($F(2)$)
READY(✗, $\langle 1,\mathbf{3}\rangle$) ... READY(✗, $\langle 2,\mathbf{1}\rangle$)	READY(✗, $\langle 1,\mathbf{3}\rangle$) ... READY(✗, $\langle 2,\mathbf{1}\rangle$)		READY(✗, $\langle 1,\mathbf{3}\rangle$) ... READY(✗, $\langle 2,\mathbf{1}\rangle$)
prepared($\langle 2,\mathbf{2}\rangle$)	prepared($\langle 2,\mathbf{2}\rangle$)		prepared($\langle 2,\mathbf{2}\rangle$)
VOTE(✓, $\langle 2,\mathbf{2}\rangle$)	VOTE(✓, $\langle 2,\mathbf{2}\rangle$)		VOTE(✓, $\langle 2,\mathbf{2}\rangle$)
READY(✓, $\langle 2,\mathbf{2}\rangle$)	READY(✓, $\langle 2,\mathbf{2}\rangle$)		READY(✓, $\langle 2,\mathbf{2}\rangle$)
committed($\langle 2,\mathbf{2}\rangle$)	committed($\langle 2,\mathbf{2}\rangle$)		committed($\langle 2,\mathbf{2}\rangle$)

SCP over $3f + 1$

1 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

2 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

3

4 cand: $\langle 2,\mathbf{2}\rangle$ prep: $\langle 2,\mathbf{2}\rangle$

⋮	⋮	⋮	⋮
timeout	timeout		timeout
VOTE(X, $\langle 1,\mathbf{3}\rangle$) ... VOTE(X, $\langle 2,\mathbf{1}\rangle$)	VOTE(X, $\langle 1,\mathbf{3}\rangle$) ... VOTE(X, $\langle 2,\mathbf{1}\rangle$)		VOTE(X, $\langle 1,\mathbf{3}\rangle$) ... VOTE(X, $\langle 2,\mathbf{1}\rangle$)
start-timer($F(2)$)	start-timer($F(2)$)		start-timer($F(2)$)
READY(X, $\langle 1,\mathbf{3}\rangle$) ... READY(X, $\langle 2,\mathbf{1}\rangle$)	READY(X, $\langle 1,\mathbf{3}\rangle$) ... READY(X, $\langle 2,\mathbf{1}\rangle$)		READY(X, $\langle 1,\mathbf{3}\rangle$) ... READY(X, $\langle 2,\mathbf{1}\rangle$)
prepared($\langle 2,\mathbf{2}\rangle$)	prepared($\langle 2,\mathbf{2}\rangle$)		prepared($\langle 2,\mathbf{2}\rangle$)
VOTE(✓, $\langle 2,\mathbf{2}\rangle$)	VOTE(✓, $\langle 2,\mathbf{2}\rangle$)		VOTE(✓, $\langle 2,\mathbf{2}\rangle$)
READY(✓, $\langle 2,\mathbf{2}\rangle$)	READY(✓, $\langle 2,\mathbf{2}\rangle$)		READY(✓, $\langle 2,\mathbf{2}\rangle$)
committed($\langle 2,\mathbf{2}\rangle$)	committed($\langle 2,\mathbf{2}\rangle$)		committed($\langle 2,\mathbf{2}\rangle$)
decide($\mathbf{2}$)	decide($\mathbf{2}$)		decide($\mathbf{2}$)

SCP and federated voting

Abstract version of SCP:
Uses federated voting as a black box on each ballot:

- Not directly implementable because of infinity of ballots considered.
- Needs to exchange *batches of messages* instead of individual messages.

Concrete version of SCP:
Uses a variation of federated voting on statements $PRE\ b \equiv \{(\text{✗}, b') \mid b' \lessapprox b\}$ and $CMT\ b \equiv (\checkmark, b)$:

- Directly implementable because of finiteness of statements considered.
- Does not use federated voting as a black box.

SCP and federated voting

Abstract version of SCP:

Uses federated voting as a black box on each ballot:

- Not directly implementable because of infinity of ballots considered.
- Needs to exchange *batches of messages* instead of individual messages.

Concrete version of SCP:

Uses a variation of federated voting on statements $PRE\ b \equiv \{(\text{✗}, b') \mid b' \lesssim b\}$ and $CMT\ b \equiv (\checkmark, b)$:

- Directly implementable because of finiteness of statements considered.
- Does not use federated voting as a black box.

Modular proof of correctness:

- Prove abstract SCP correct using previous results on federated voting.
- Show that concrete SCP *observationally refines* abstract SCP.

Conclusions

- Decentralised trust via FBQS, typical of permissionless blockchains.
- Hard guarantees and low latency and energy consumption, typical of permissioned blockchains.
- SCP implements a variant of Byzantine consensus where properties are relative to disjoint fragments with internal consistency:
  - Safety within the intact set.
  - Liveness for intact sets after malicious nodes stop.
- Correctness of SCP proved modularly by using results of federated voting previously investigated, and by refinement.

Conclusions

- Decentralised trust via FBQS, typical of permissionless blockchains.
- Hard guarantees and low latency and energy consumption, typical of permissioned blockchains.
- SCP implements a variant of Byzantine consensus where properties are relative to disjoint fragments with internal consistency:
  - Safety within the intact set.
  - Liveness for intact sets after malicious nodes stop.
- Correctness of SCP proved modularly by using results of federated voting previously investigated, and by refinement.

Thank you!