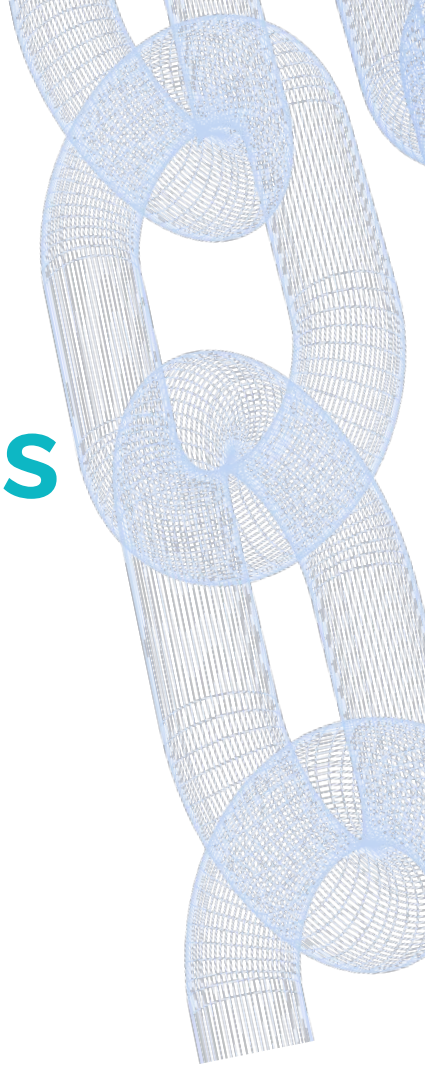
Streamlet: Textbook Streamlined Blockchains



Elaine Shi (CMU/Cornell)

Joint work with Benjamin Chan

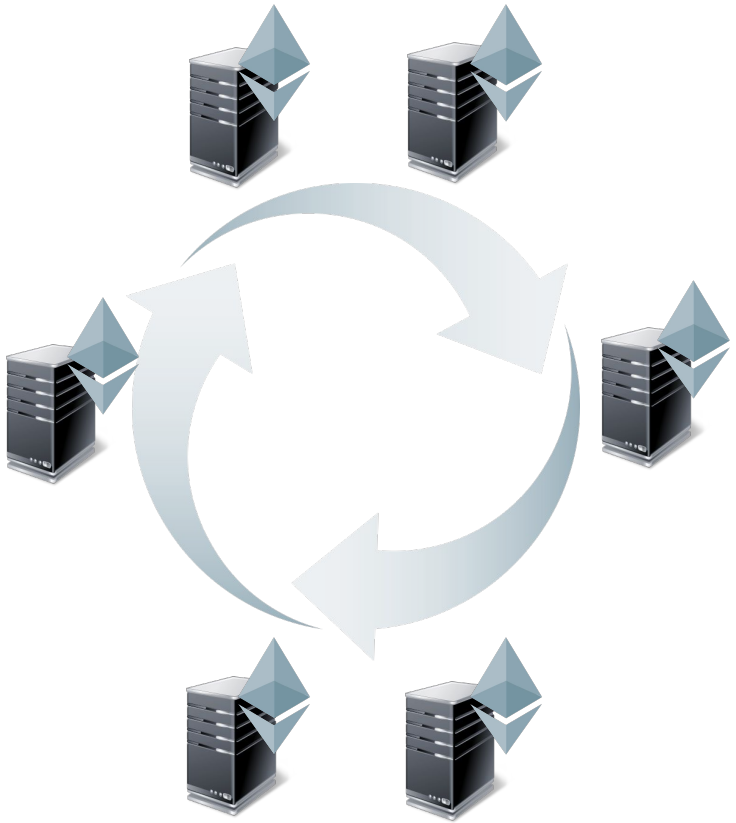
Textbook: www.distributedconsensus.net

Blockchain

(a.k.a. state machine replication, consensus)

Consistency:
Honest nodes agree on log

Liveness:
TXs are incorporated soon



Blockchain: A 30-year-old Problem

YAHOO!

facebook

reddit

Apache Zookeeper

Solr

UBER

ebay

rackspace

NetApp

Cryptocurrencies brought consensus to a large scale

Enables permissionless consensus

Proof of work

Proof of work

Proof of stake

parity

Rely on **permissioned** consensus

Proof of work

Proof of stake

parity

Classical consensus landscape

Complex
Difficult to understand
Error-prone to implement

"Paxos Made Moderately Complex"
[ACM Computing Surveys'15]

"Zyzzyva: Speculative Byzantine Fault Tolerance" [Communications of the ACM'09]

"Paxos Made Simple"

"The ABCDs of Paxos" [PODC'01]

"RAFT: In search of an understandable consensus algorithm" [Usenix ATC'14]

... ...

Streamlet

 Simple

 Natural

 Unified, for pedagogy & implementation

We can construct a blockchain through sequential composition of Byzantine Agreement (BA)

We can construct a blockchain through sequential composition of Byzantine Agreement (BA)

Direct blockchain construction

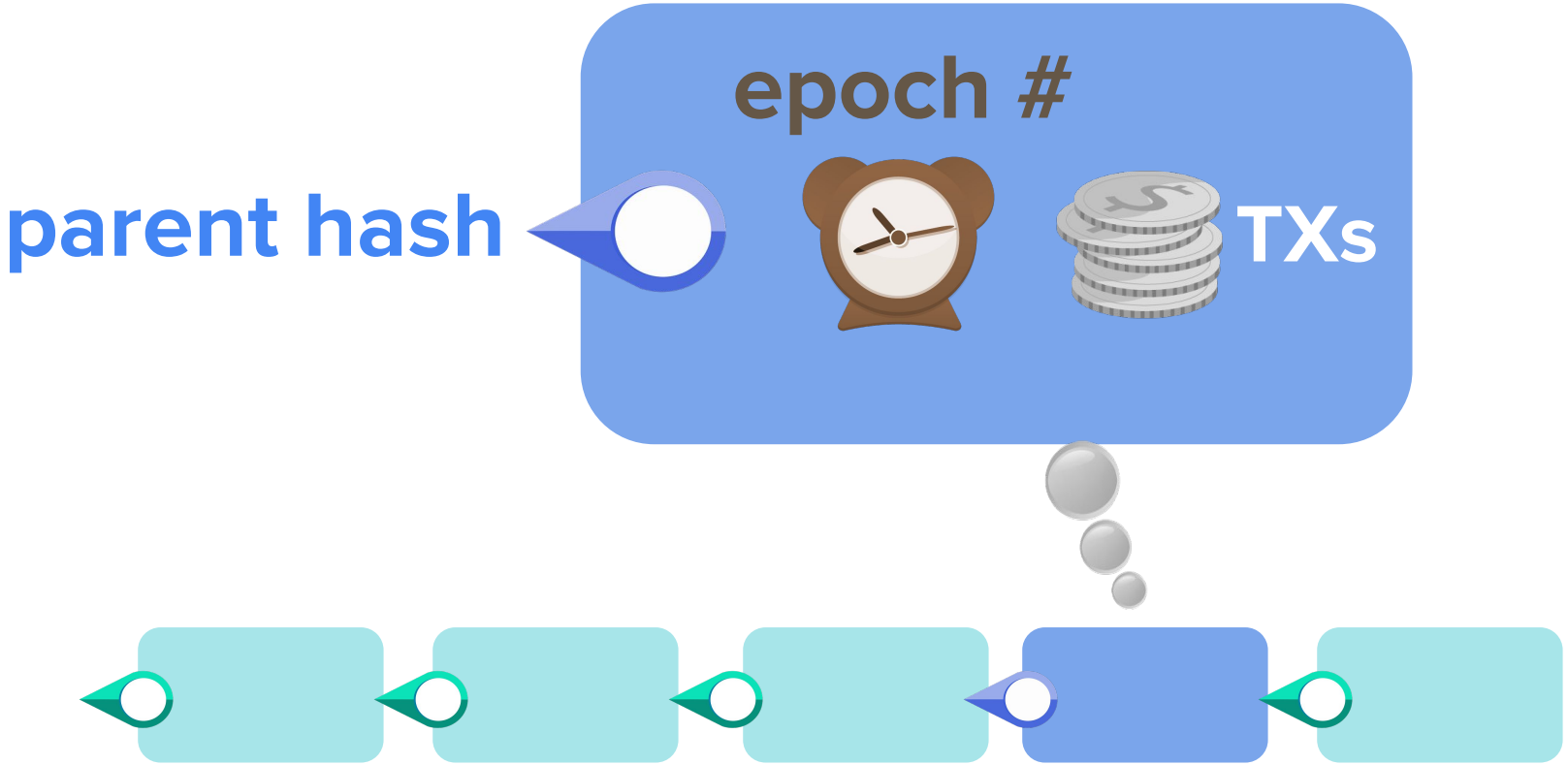
(e.g., pbft, paxos)

Roadmap

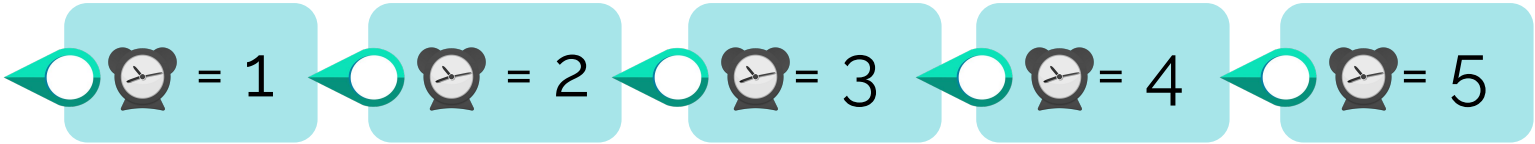
Classical approaches
(e.g., pbft, paxos)

Streamlet: a streamlined blockchain

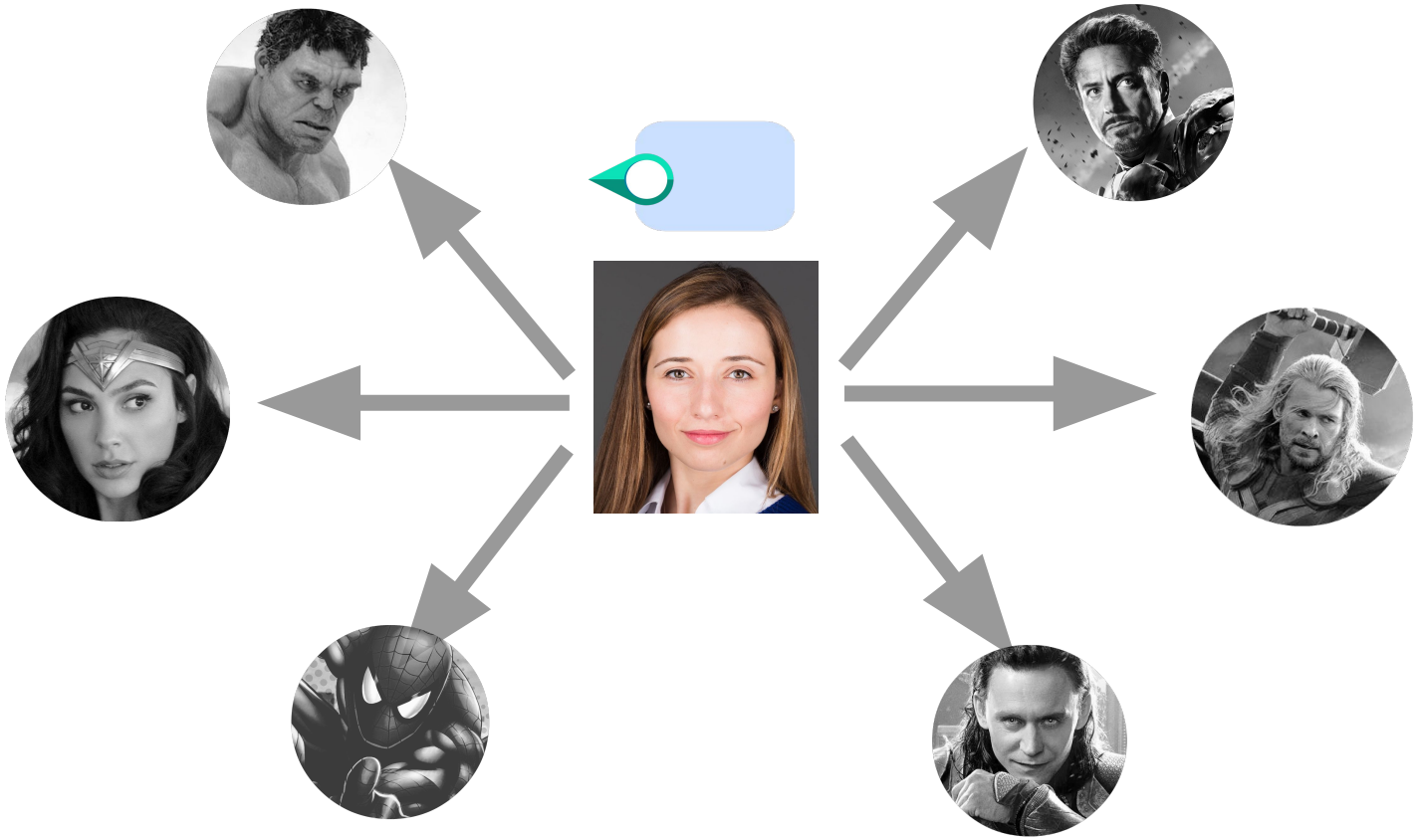
Block Format



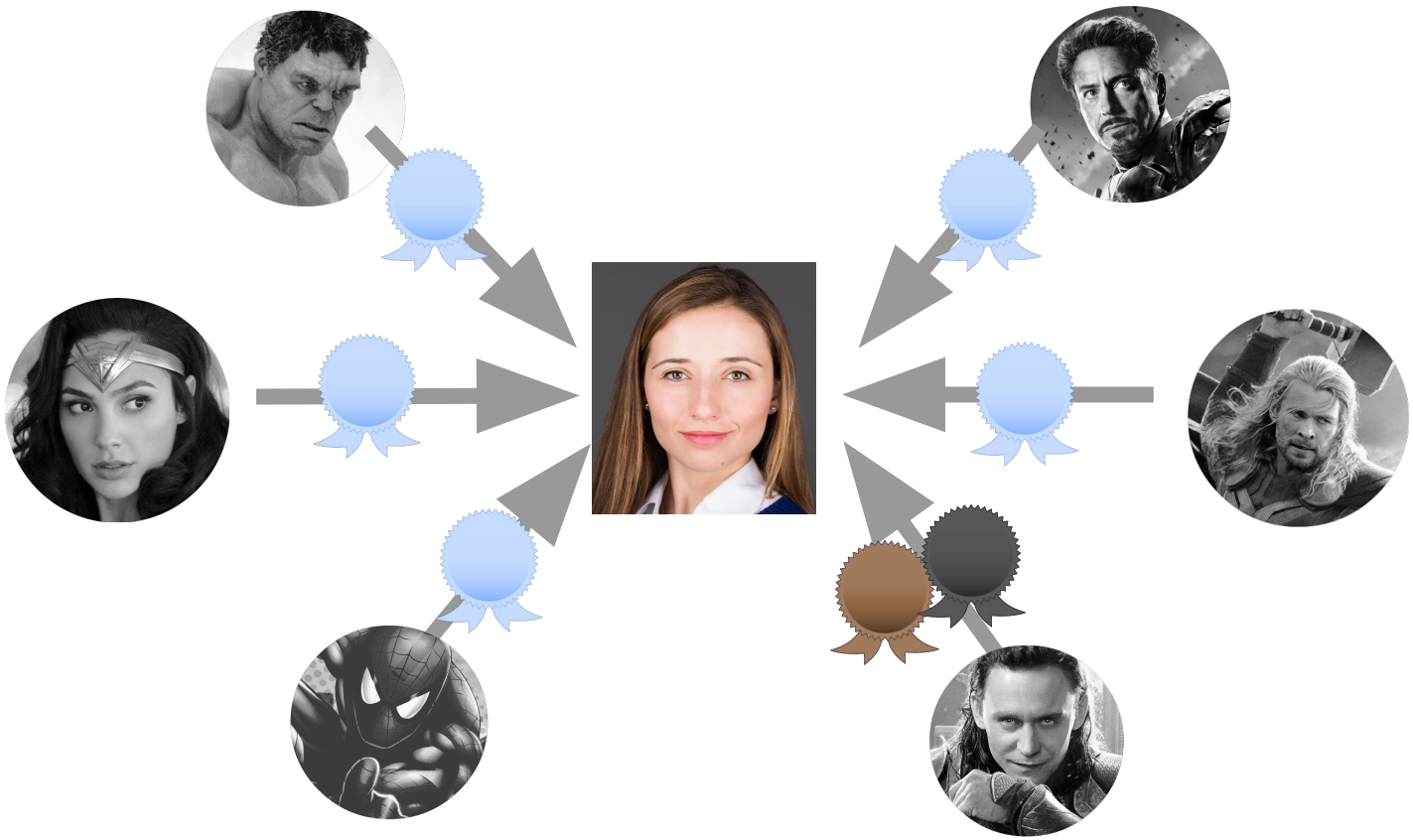
Assume: ⏰s increment in a valid blockchain



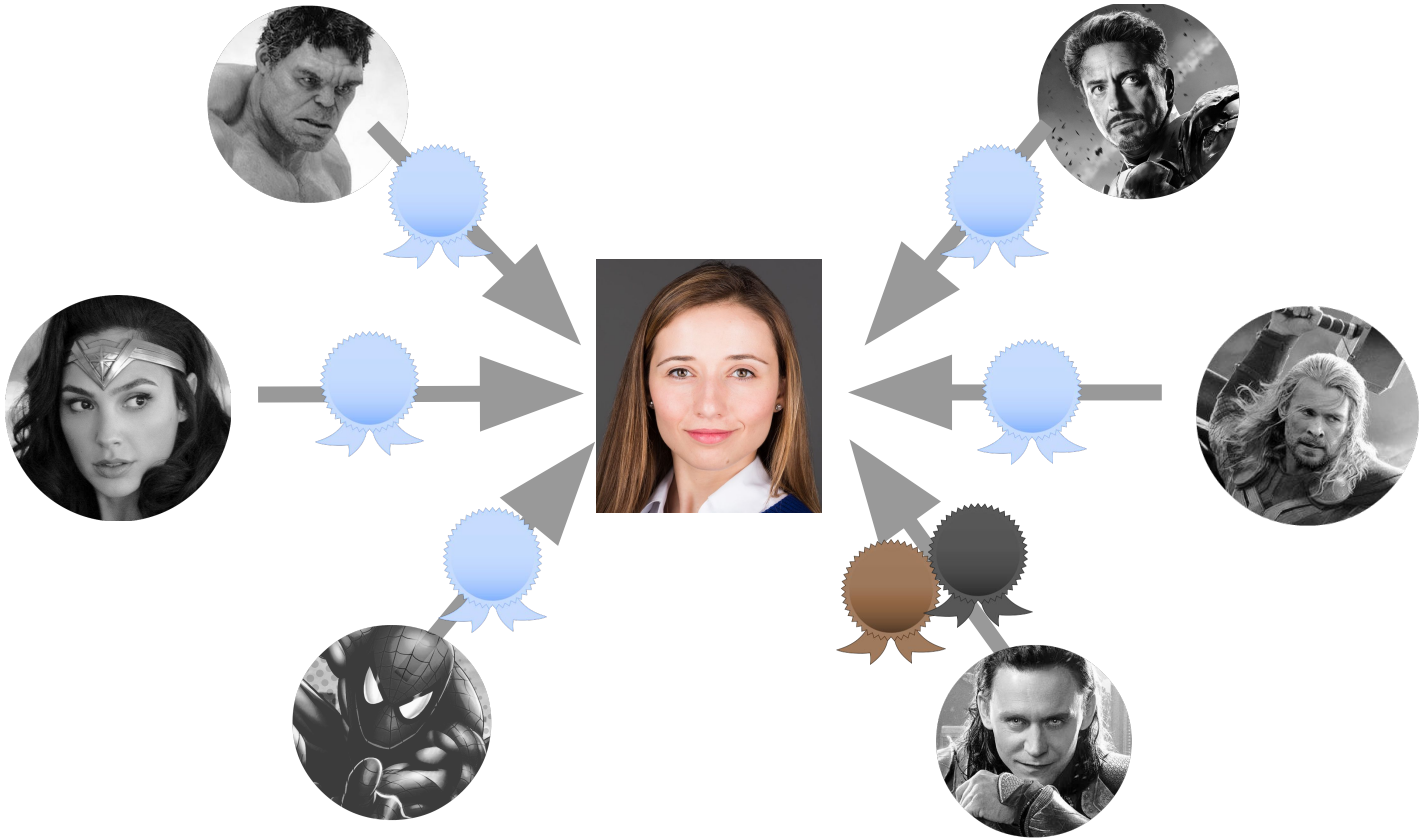
1 Leader proposes block



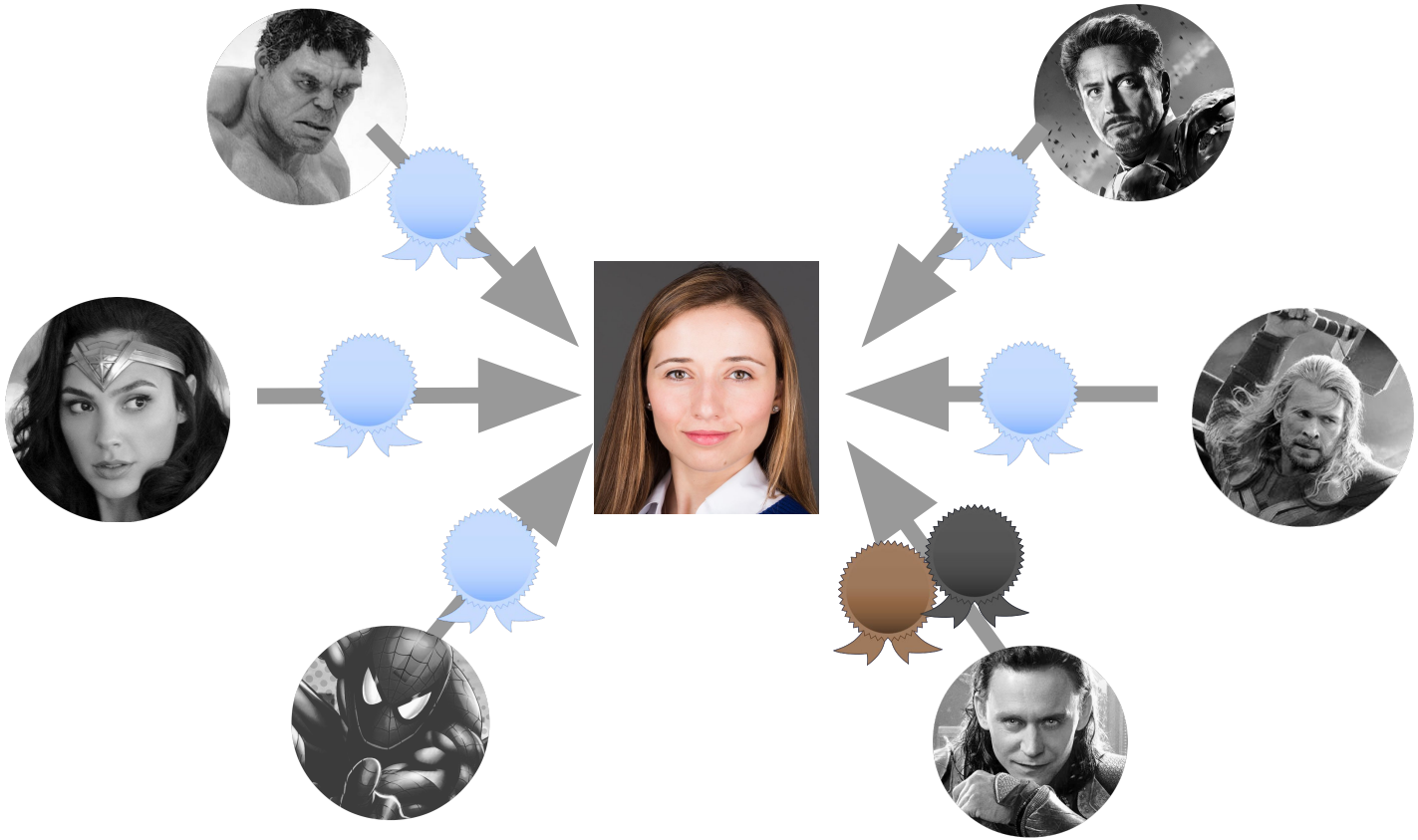
2 Vote



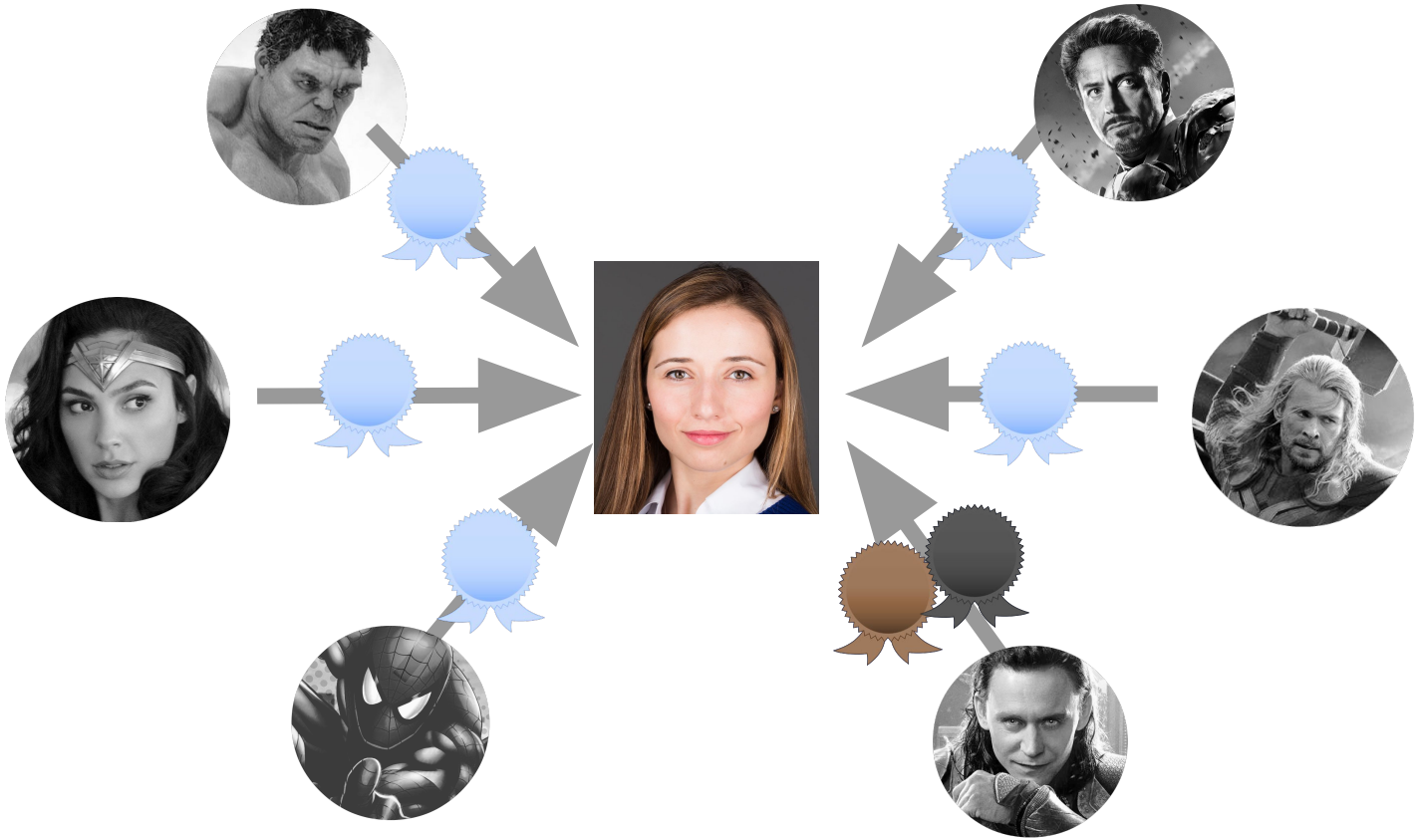
3 Confirm upon ⅔ n votes



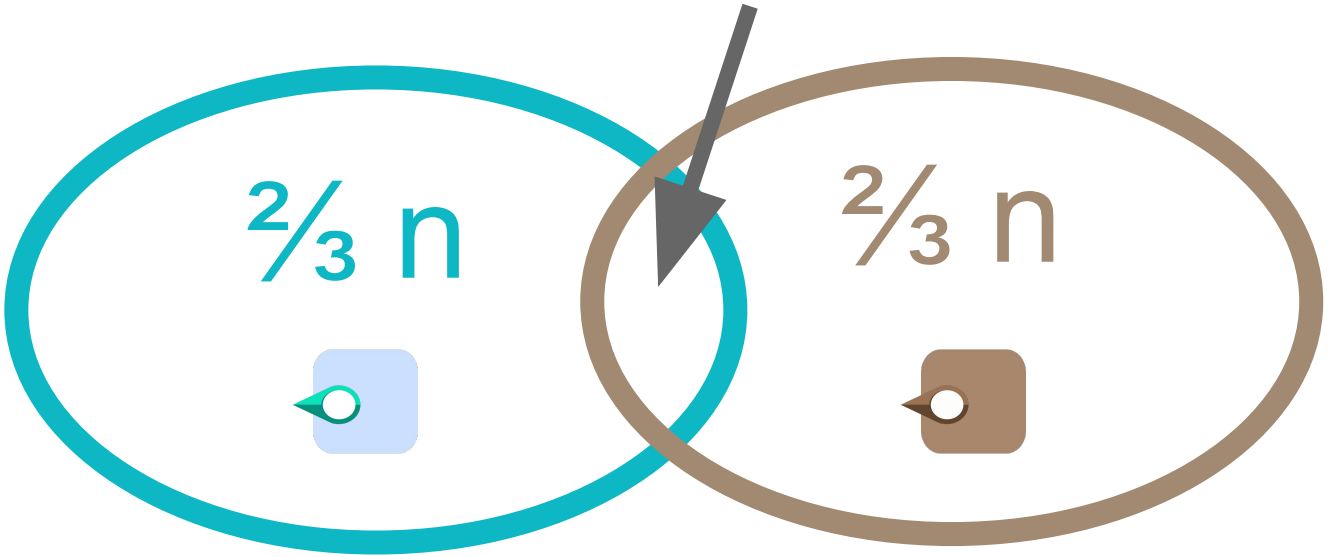
⅔ n votes: notarization



Honest nodes vote **uniquely** each epoch

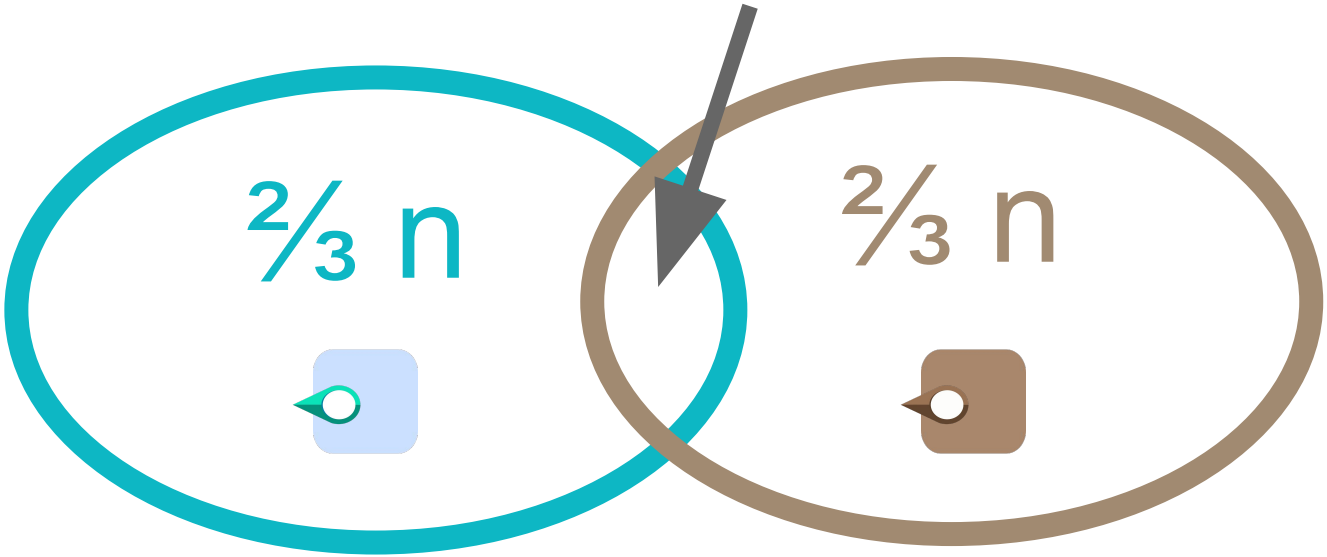


Must intersect at an honest node

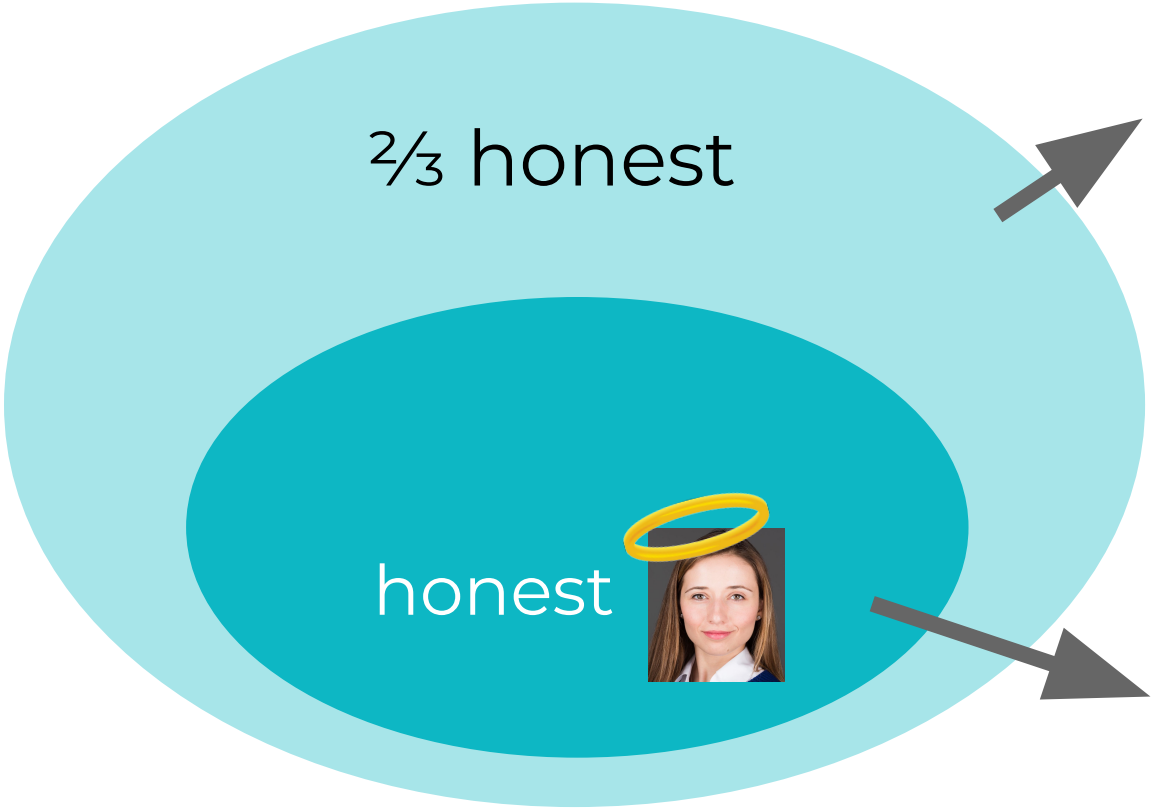


Assume: < ⅓ n corrupt

Must intersect at an honest node

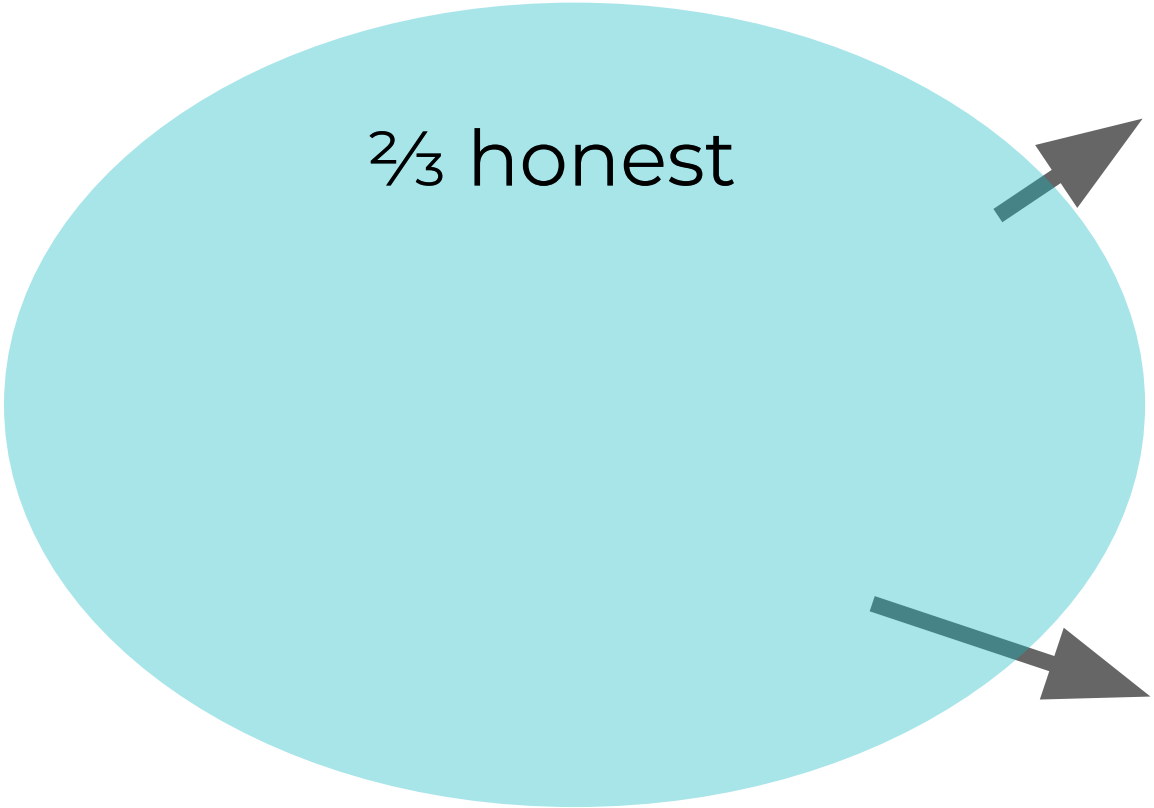


Thus  **=**



Consistency

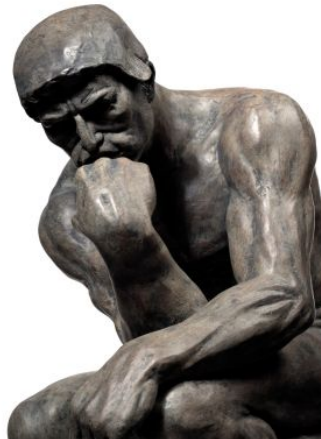
Liveness



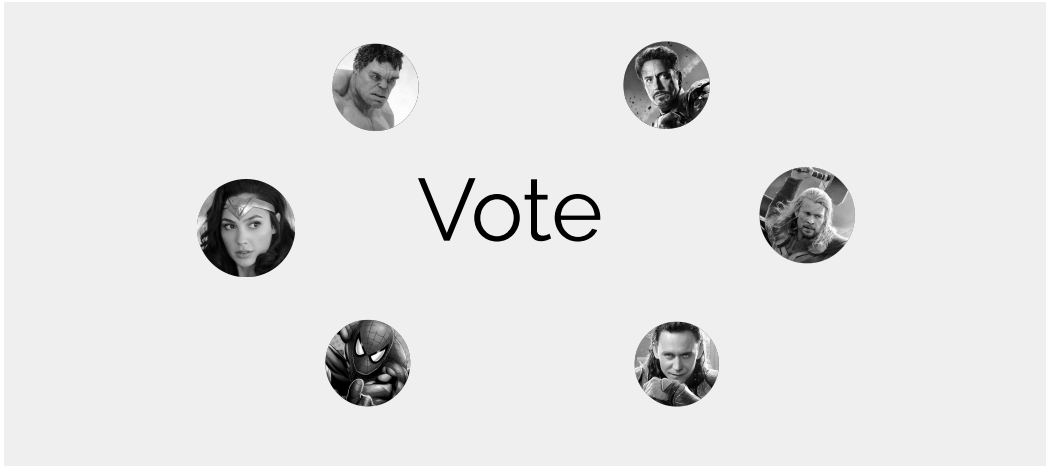
Consistency

Liveness

How do we achieve liveness?



Anatomy of classical consensus



Simple normal path

Complicated recovery path

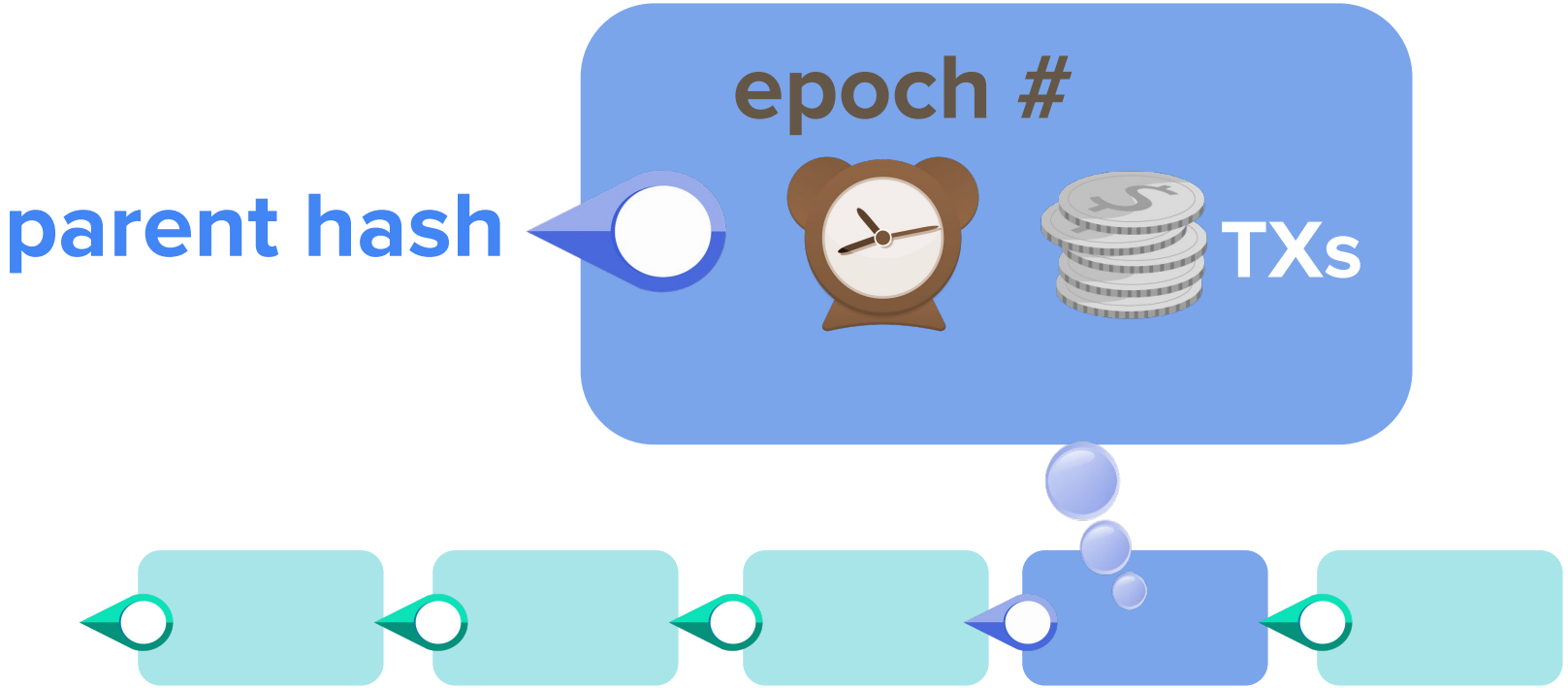
Can we achieve full consensus
(almost) as simply as the normal path?

Roadmap

Classical approaches
(e.g., pbft, paxos)

Streamlet

Assume: epoch = 1 sec ≥ 1 roundtrip



Leader rotation

Player H(e) mod n is the leader in epoch e

Easy to support any other leader-rotation policy

Assume honest nodes do the following

- receives msgs from the network
- echos every fresh msg seen
- updates its longest notarized chain every round

Propose

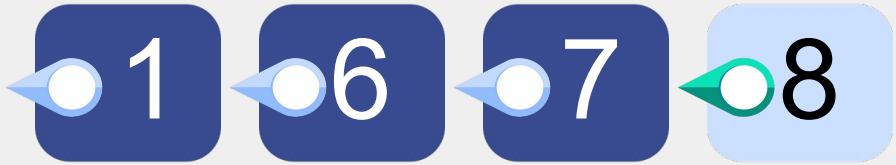
extend longest notarized chain

Vote

for 1st proposal from leader iff it extends from one of the longest notarized chains seen

Every epoch

Finalization: 3 consecutive epochs appear together in a notarized chain, all but last final

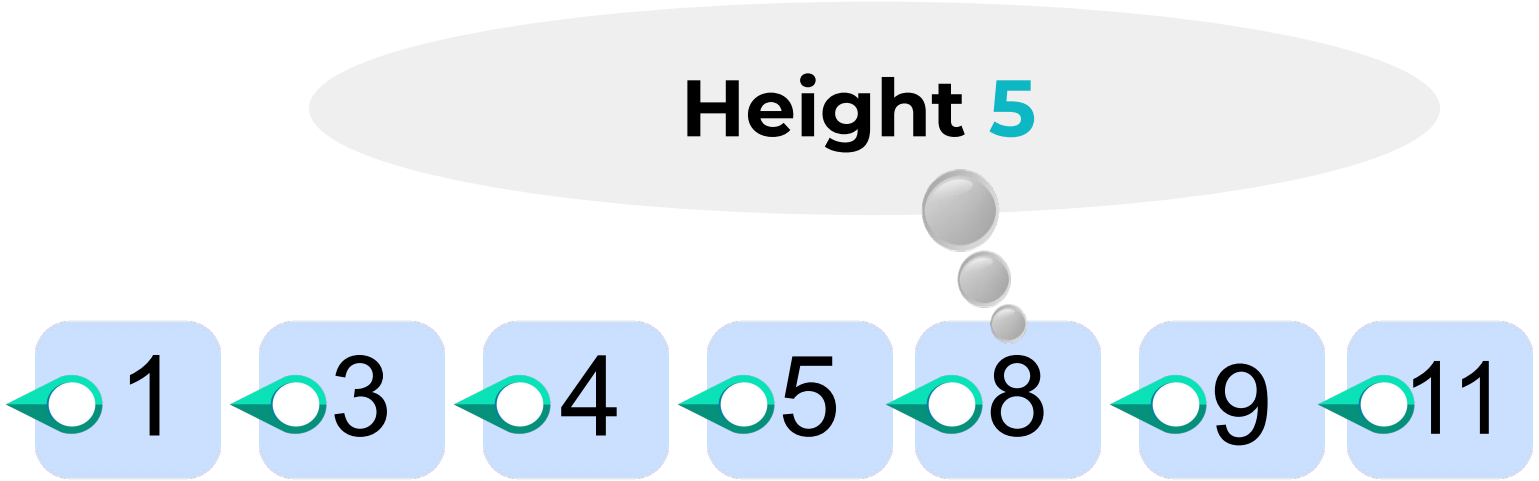


Finalization: 3 consecutive epochs appear together in a notarized chain, all but last final

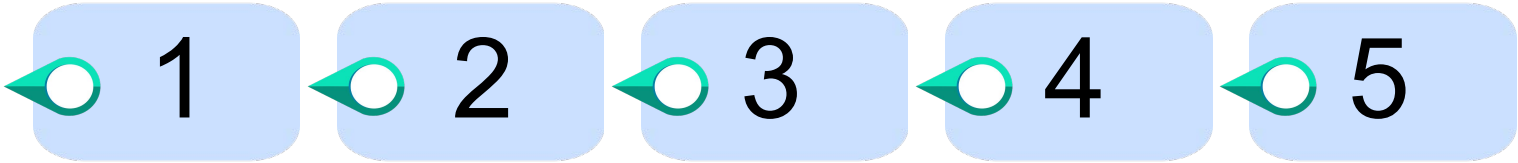
1. Consistency Proof

2. Liveness Proof

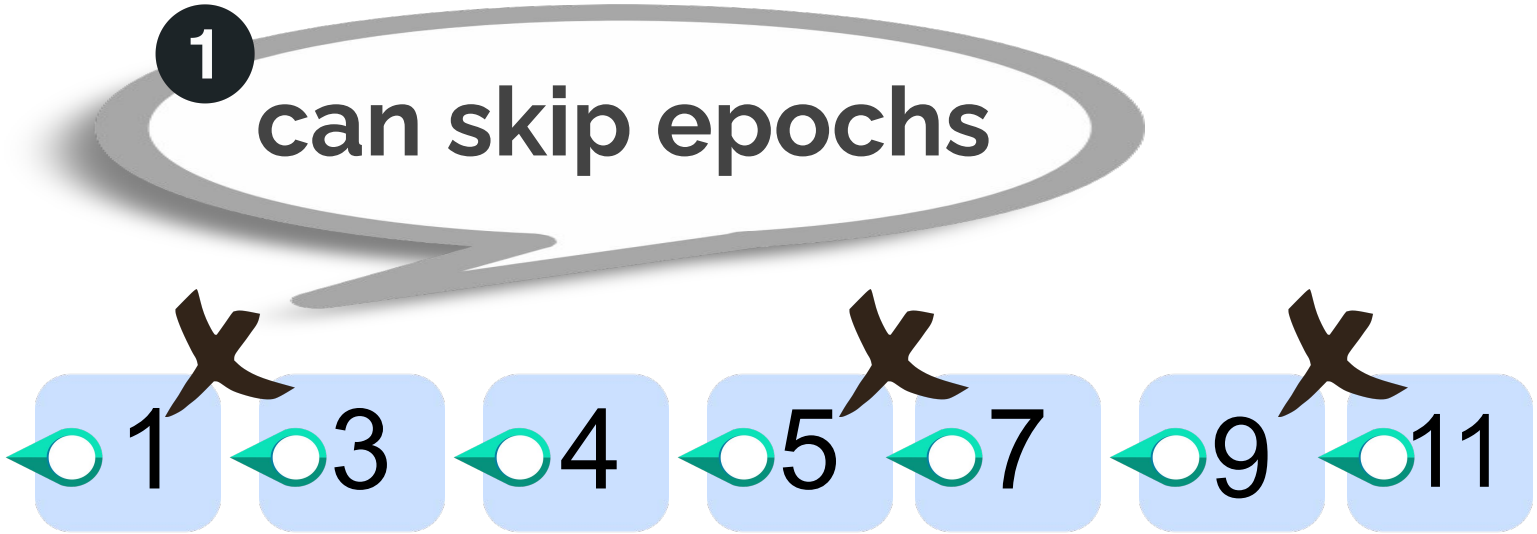
Height = “position in chain”



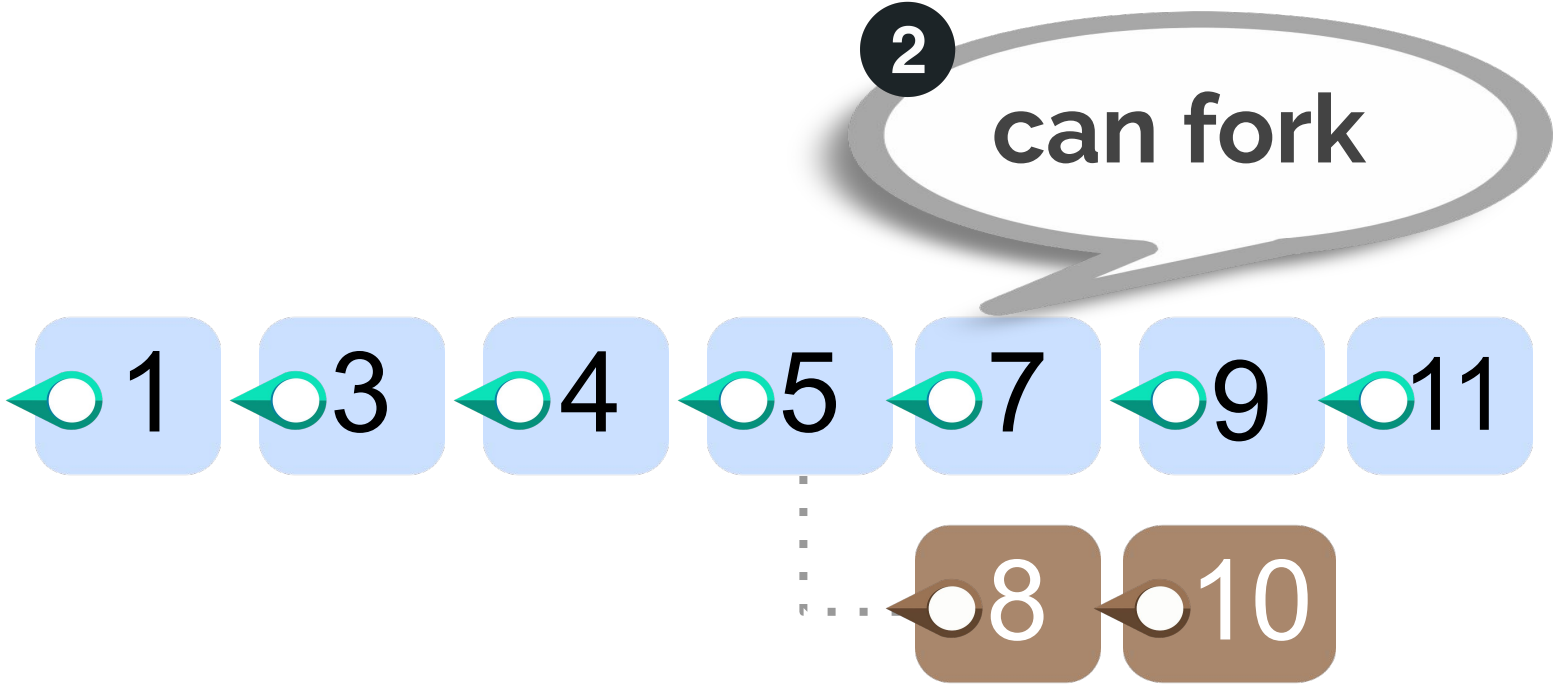
If everyone were honest



Real world



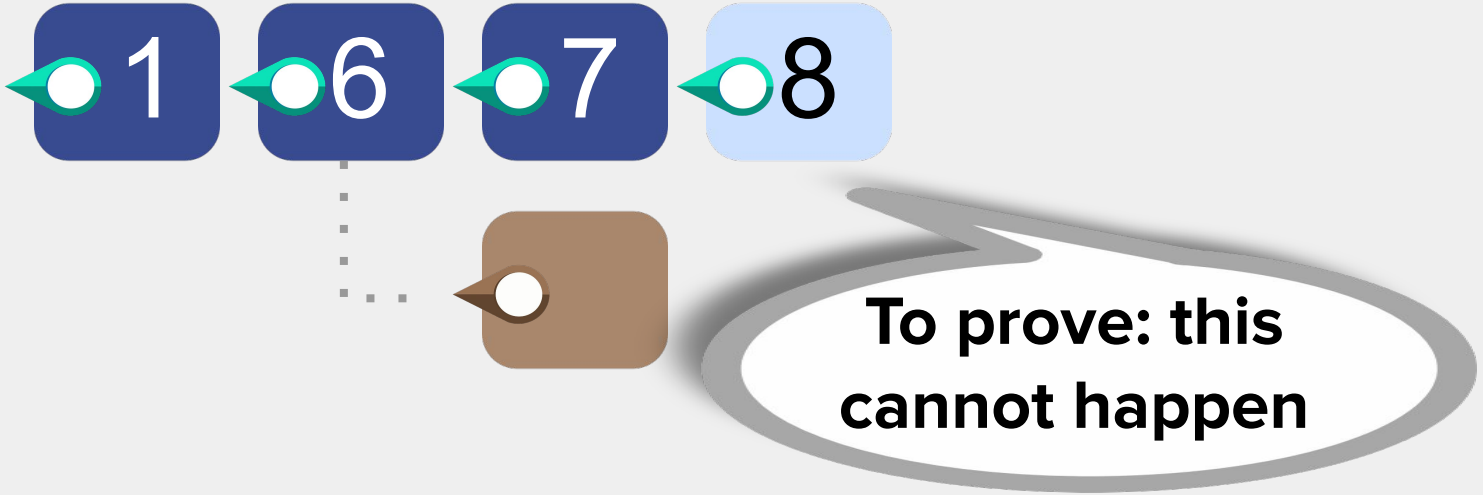
Real world

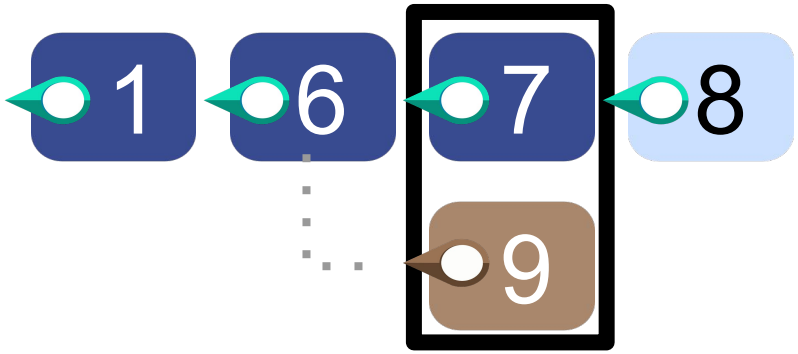


1. Consistency Proof

2. Liveness Proof

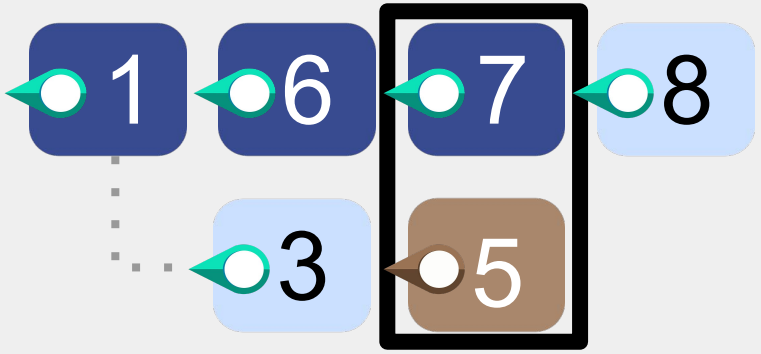
Finalization: 3 consecutive epochs in notarized chain, all but last final



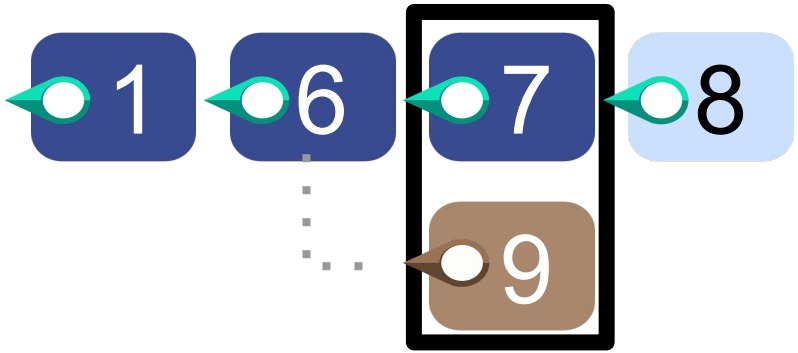


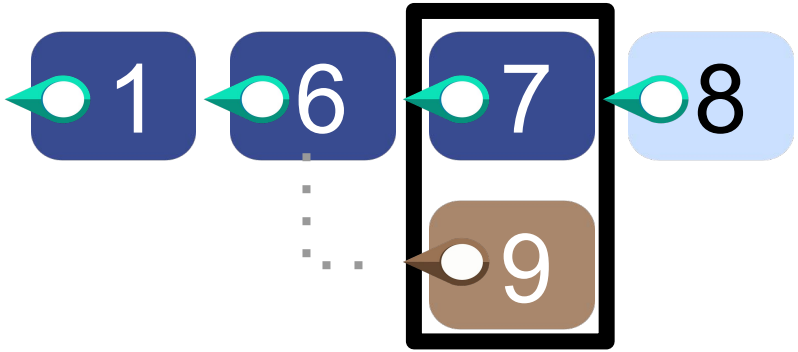
Case 1

Case 2



Lemma: every epoch has at most 1 notarized block.

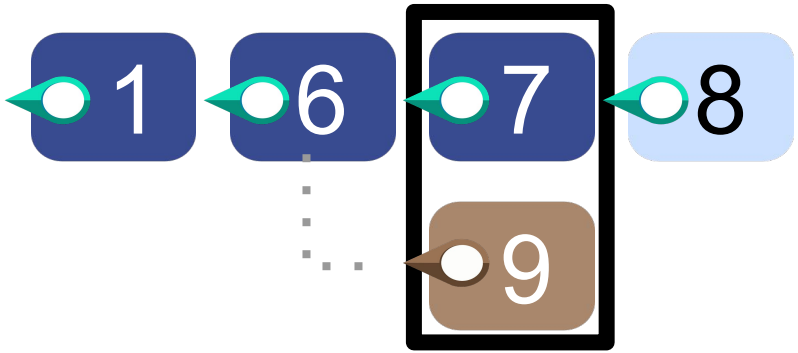

1
6
7
8
9



“many”: > n/3 honest

Proof:

many voted for 8 in epoch 8

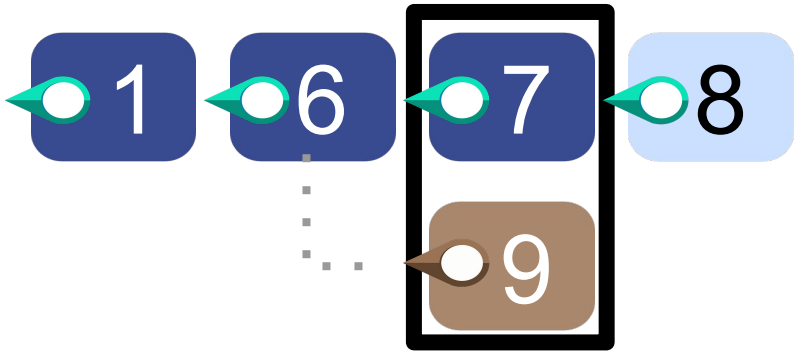


"many": > n/3 honest

Proof:

many voted for 8 in epoch 8

--> many saw 7 notarized in epoch 8



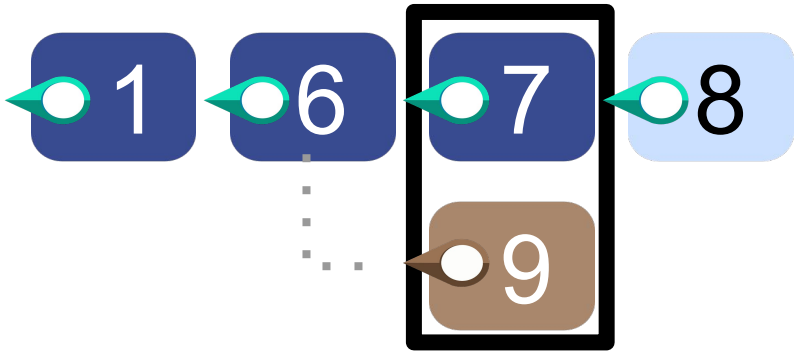
“many”: > n/3 honest

Proof:

many voted for 8 in epoch 8

--> many saw 7 notarized in epoch 8

--> they will not vote for 9 in epoch 9



"many": > n/3 honest

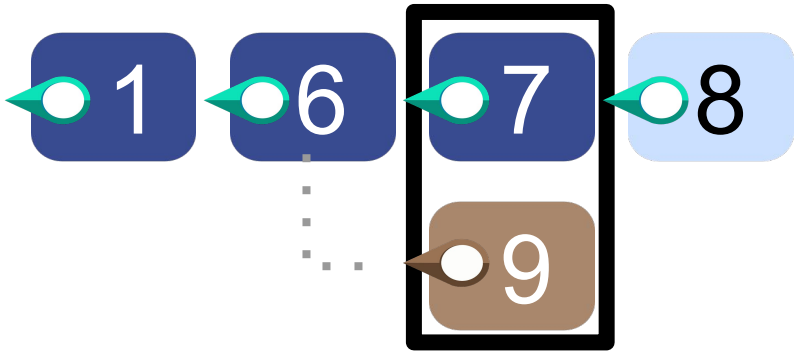
Proof:

many voted for 8 in epoch 8

--> many saw 7 notarized in epoch 8

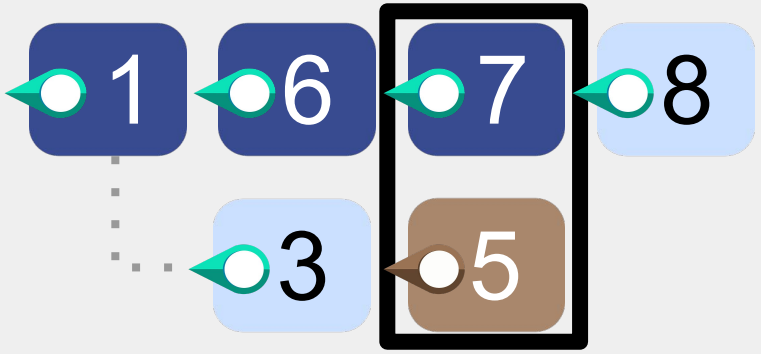
--> they will not vote for 9 in epoch 9

--> 9 cannot gain notarization



Case 1

Case 2



“many” : > n/3 honest

Proof:

many voted for 5 in epoch 5

“many” : > n/3 honest

Proof:

many voted for 5 in epoch 5

--> many saw 3 notarized in epoch 5

“many” : > n/3 honest

Proof:

many voted for 5 in epoch 5

--> many saw 3 notarized in epoch 5

--> they will not vote for 6 in epoch 6

“many” : > n/3 honest

Proof:

many voted for 5 in epoch 5

--> many saw 3 notarized in epoch 5

--> they will not vote for 6 in epoch 6

--> 6 cannot gain notarization

“many” : > n/3 honest

Proof:

many voted for 5 in epoch 5

--> many saw 3 notarized in epoch 5

--> they will not vote for 6 in epoch 6

--> 6 cannot gain notarization

“many” : > n/3 honest

Proof:

many voted for 5 in epoch 5

--> many saw 3 notarized in epoch 5

--> they will not vote for 6 in epoch 6

--> 6 cannot gain notarization

Consistency does NOT depend on sync. assumptions!

Liveness Theorem

During a period of synchrony, honest players' finalized chains grow whenever 5 consecutive epochs have honest leaders.

(and moreover the finalized chains grow by honest blocks)

Partial Synchrony [DLS]

- Protocol knows a delay estimate Δ
- Consistency is guaranteed even if actual delay arbitrarily long
- Liveness only during periods of synchrony

Partial Synchrony [DLS]

Theorem:

Cannot tolerate ⅓ or more corruptions

Summary: streamlined blockchains

- Every epoch allows leader-switch.
- Leader-switch embedded in a unified “propose-vote” paradigm.

Roadmap

½ synchronous

⅓ partially synchronous

Every epoch:

- Leader proposes a block extending longest notarized chain
- Vote on the first proposal iff it extends from one of the longest notarized chains seen
- A block with majority votes is notarized

Finalization:

- 6 consecutive at the end, no conflicting notarization, chop off 5

Read after me:

- Propose-vote, propose-vote, propose-vote
- Boom boom boom
- Don’t finalize upon notarization
- 3 consecutive epochs together, chop off the last and finalize the prefix

Foundations of Distributed Consensus and Blockchains”
www.distributedconsensus.net

Thank You!

runting@gmail.com