



Ethereum Consensus: The Road to Faster Finality

Yann Vonlanthen
Ethereum Foundation

Ethereum Consensus: The Road to Faster Finality

Yann Vonlanthen

Consensus Team @ Ethereum Foundation



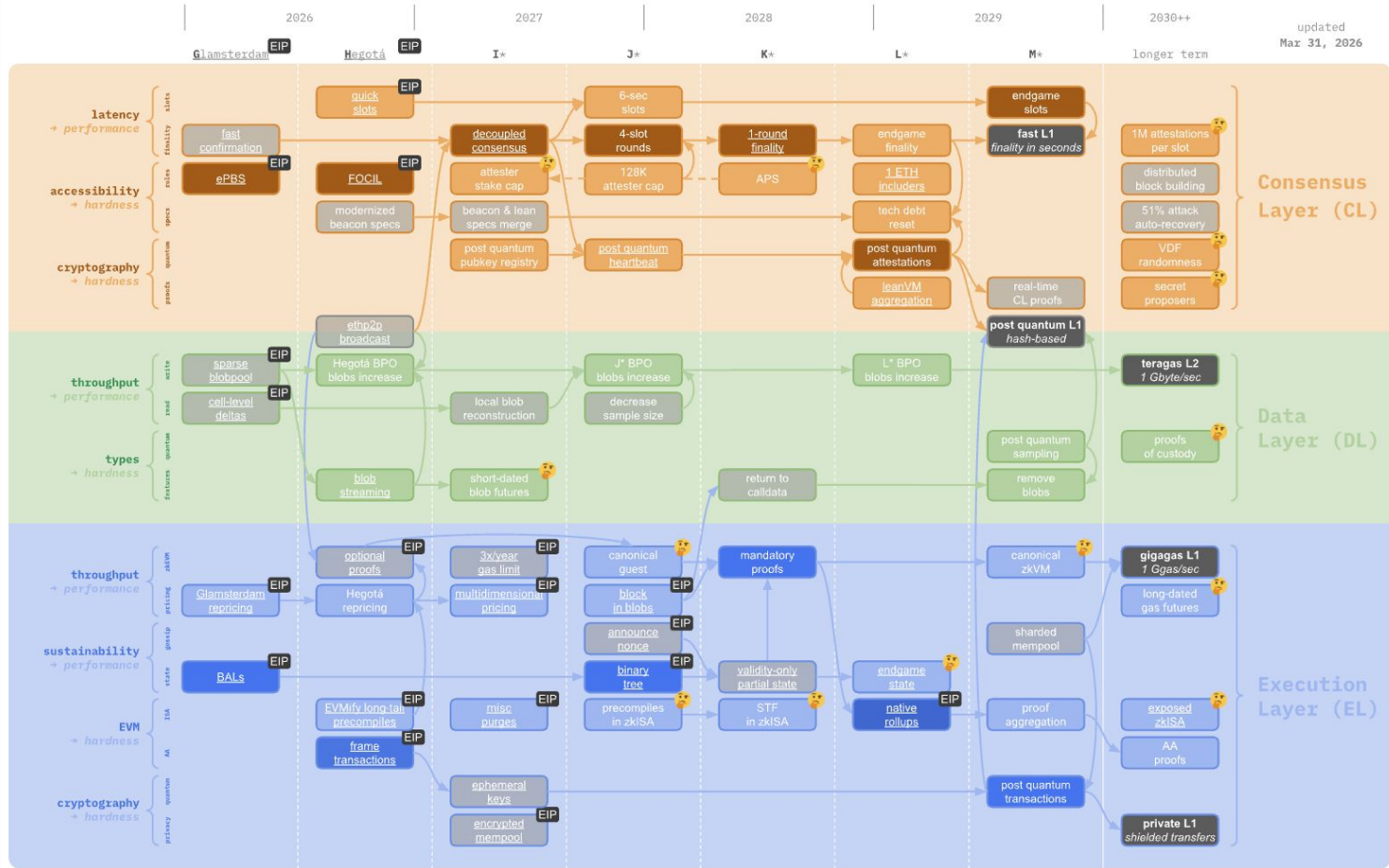
Francesco D'Amato

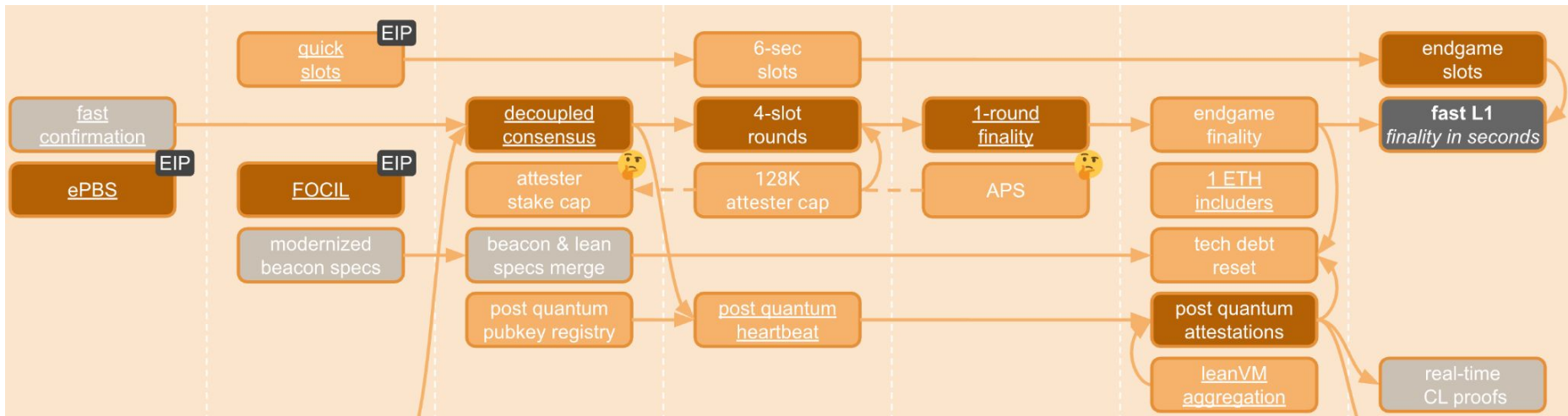


Roberto Saltini

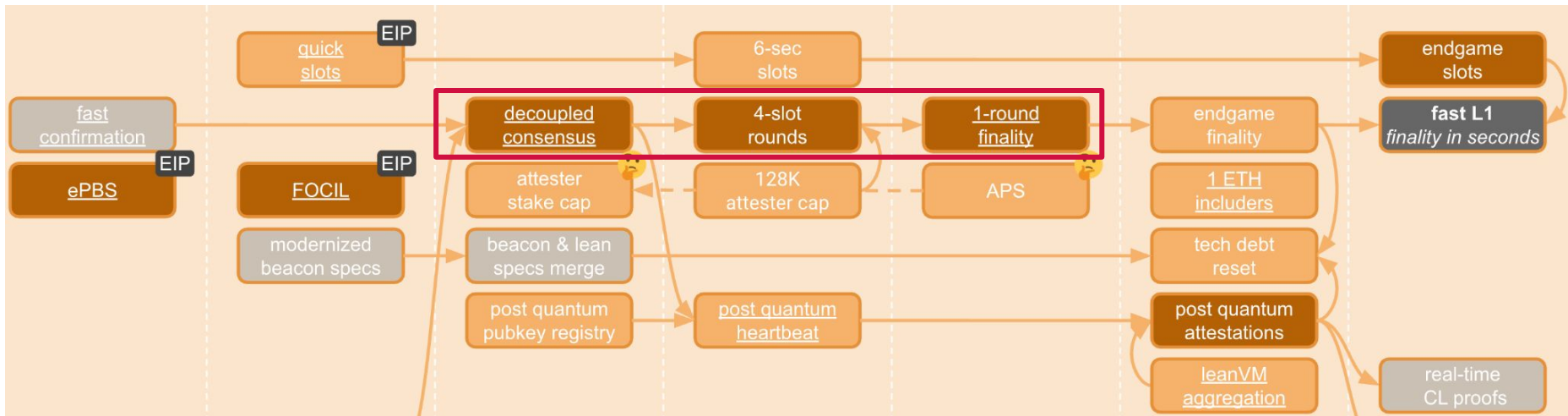


Luca Zanolini

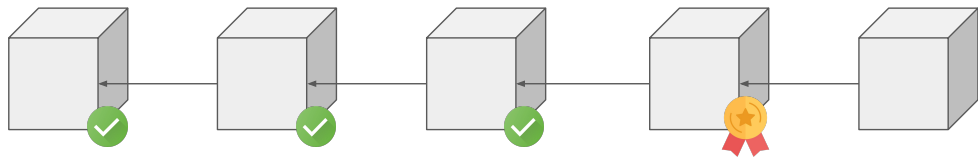


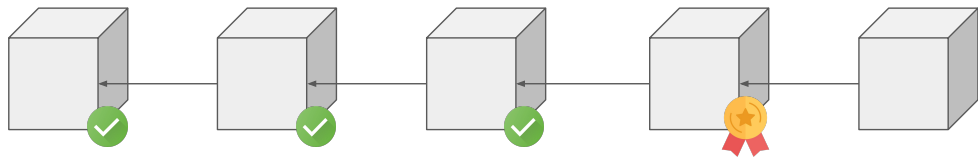


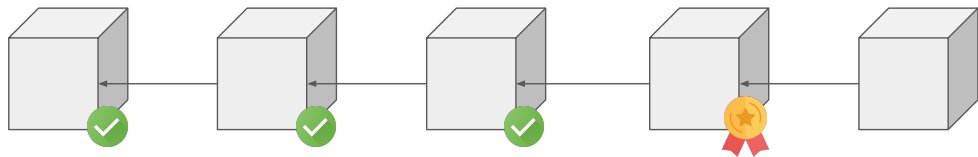
Consensus Layer

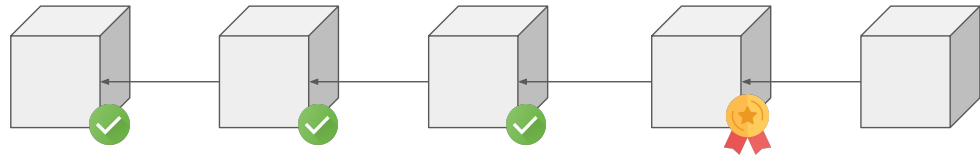


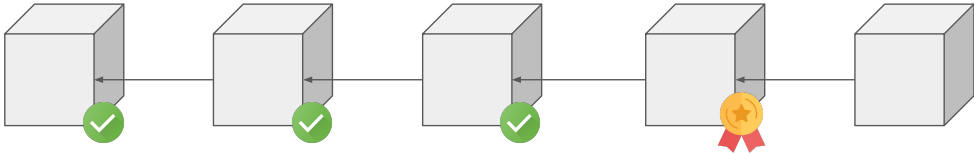
Consensus Layer

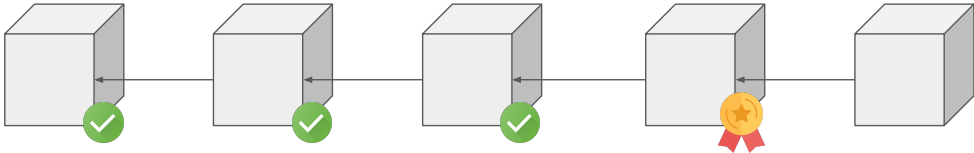


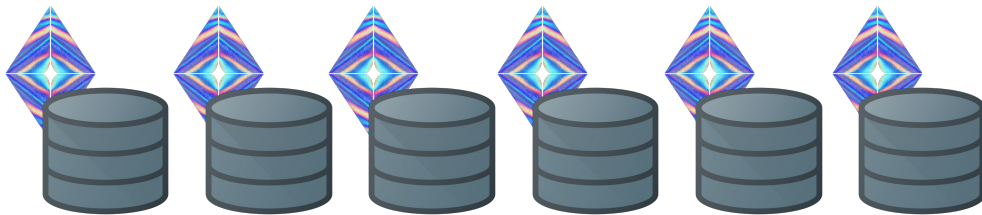
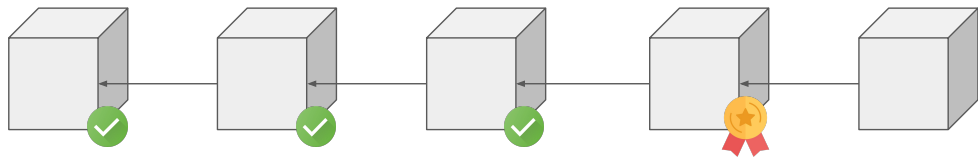


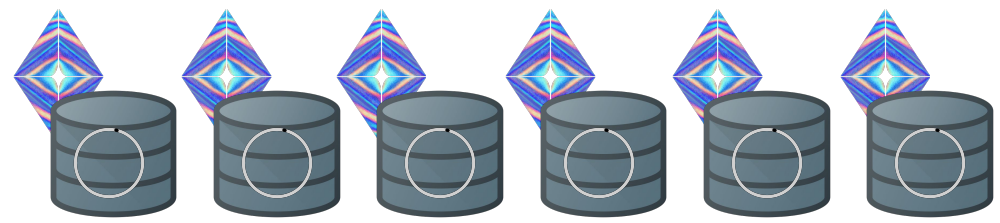
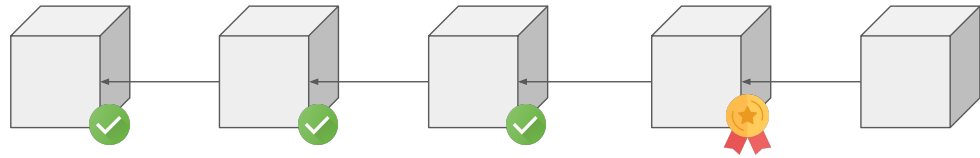


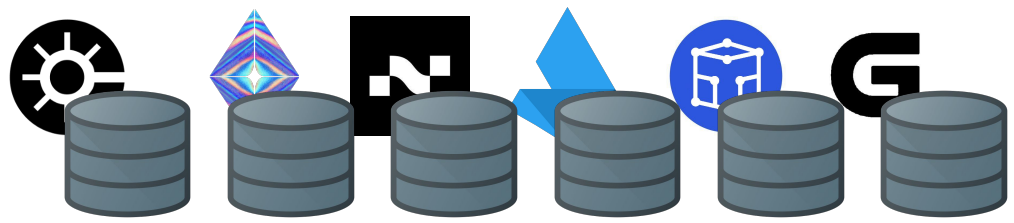
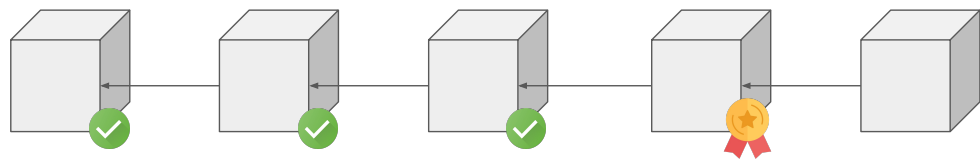


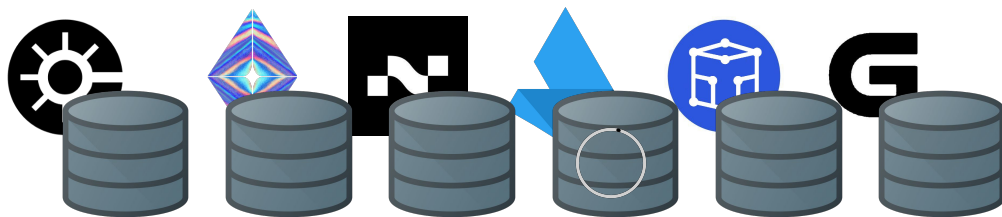
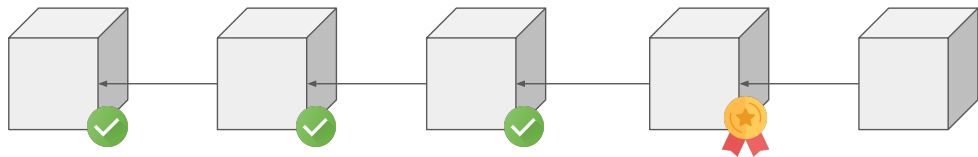


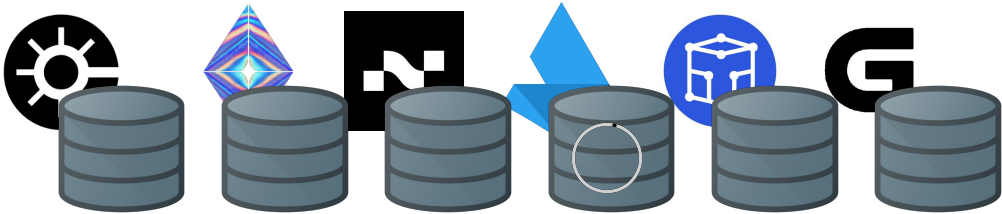
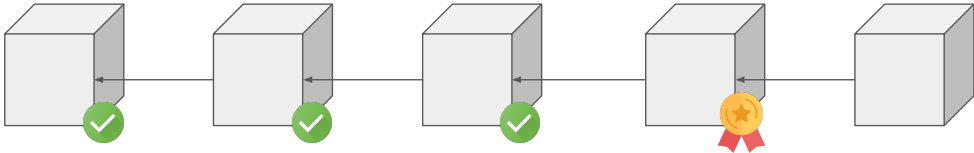










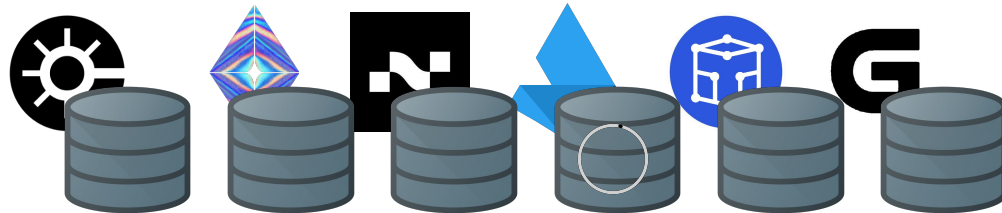


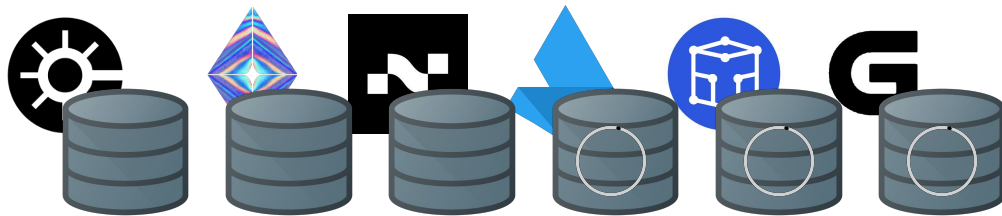
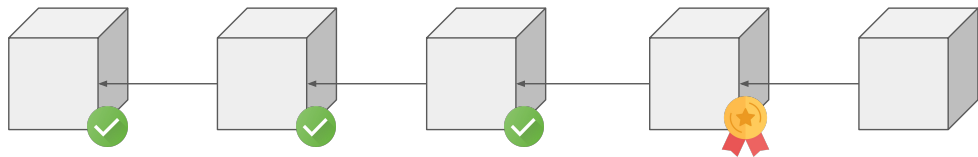
Validator Activity in Ethereum Drops 25% Following Fusaka Deployment

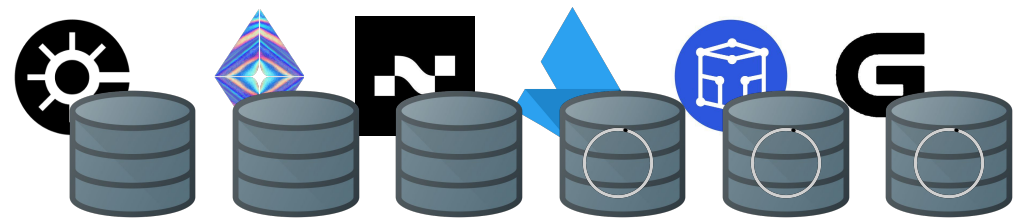
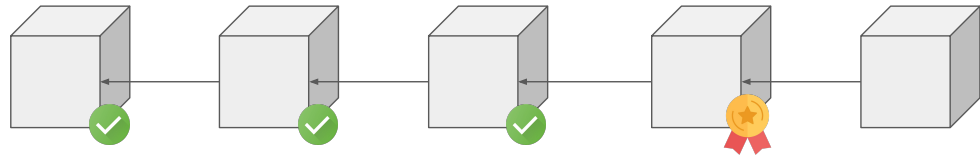
Validator activity in Ethereum drops 25% following Fusaka deployment.

05.12.2025 • ForkLog

Shortly after the deployment of Fusaka, a malfunction occurred in the popular consensus client Prysm, disabling a portion of Ethereum validators.







<https://blog.sigmaprime.io/pectra-holesky-incident.html>

Article — Analysis

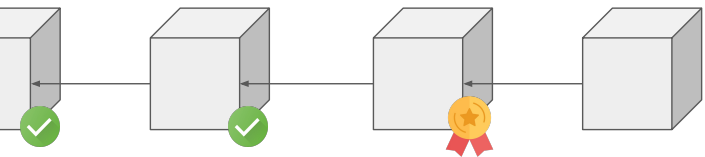
The Pectra Holesky Incident

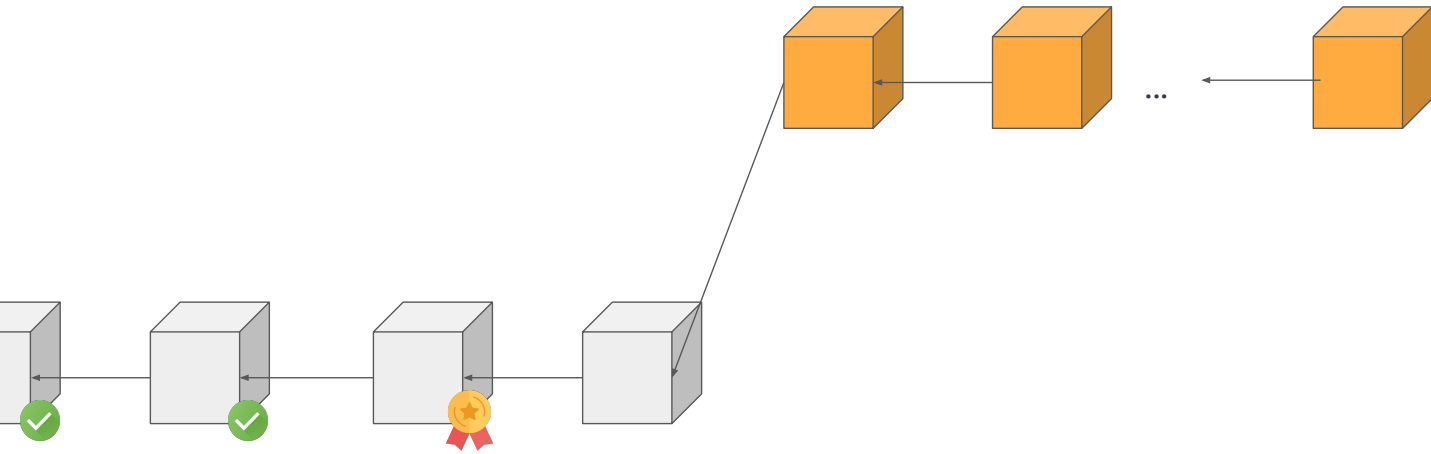


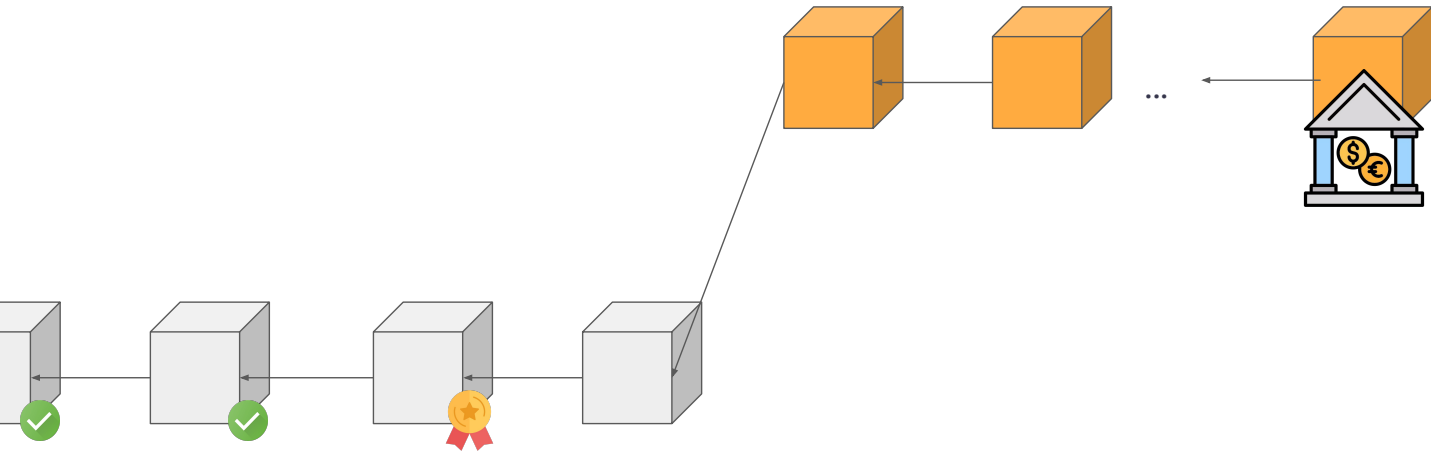
Eitan Seri-Levi

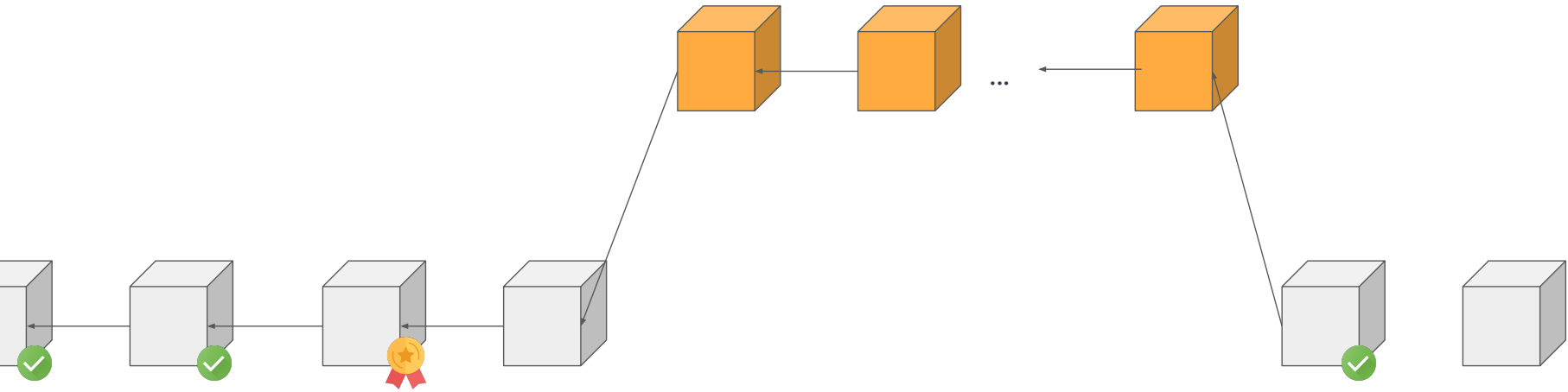
Tue 08 April 2025

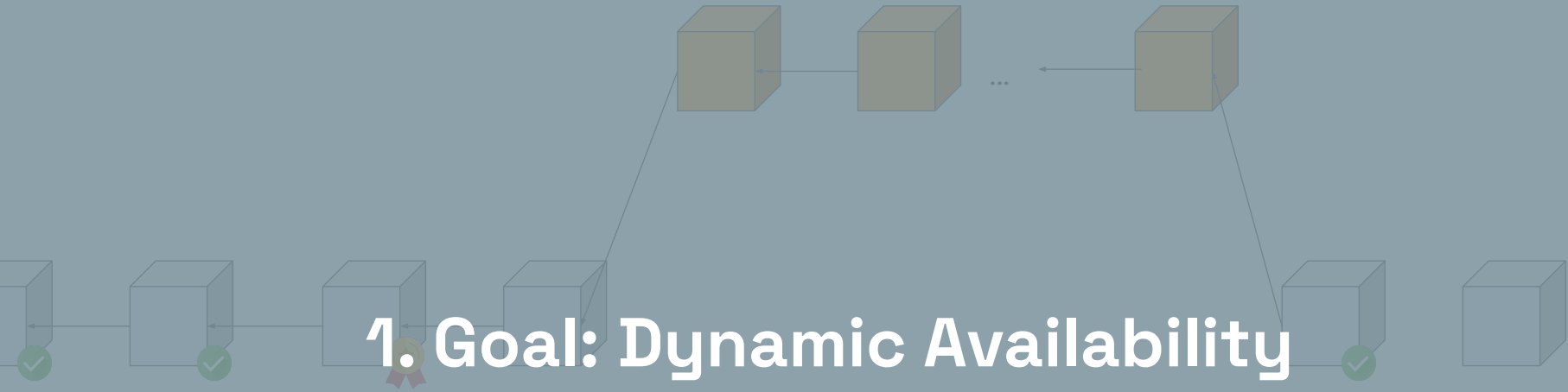












Why Ethereum Needs a Dynamically Available Protocol

■ Consensus



luca_zanolini

2  12d

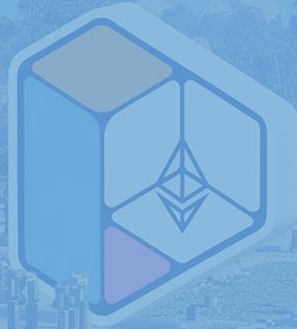
Author: [Luca Zanolini](#) ⁹

Huge thanks to [Ben](#) ⁹, [Francesco](#) ³, [Joachim](#) ⁴, [Justin](#) ⁶, [Mikhail](#) ³, [Roberto](#) ¹, [Thomas](#) ³, [Vitalik](#) ¹, and [Yann](#) ⁴ for their feedback.

We are working on a proposal for the next consensus protocol for Ethereum. A central piece of the new design is a two-layer architecture: a fast available chain — the *heartbeat* — produced by a small randomly-sampled committee, and a separate finality mechanism that trails behind, finalizing blocks the heartbeat has already produced — crucially, with the two layers fully **decoupled**, unlike the current [Gasper](#) ² design where LMD-GHOST and Casper FFG interact in ways that have [proven difficult to reason about](#) ⁶. Vitalik outlined this direction in a recent [post](#) ¹¹.

This post focuses on the first layer — the heartbeat — and on a property we believe should be a strict requirement for it: **dynamic availability**.

Ebb-and-Flow Protocols



Ebb-and-Flow Protocols: A Resolution of the Availability-Finality Dilemma

Joachim Neu
jne@stanford.edu

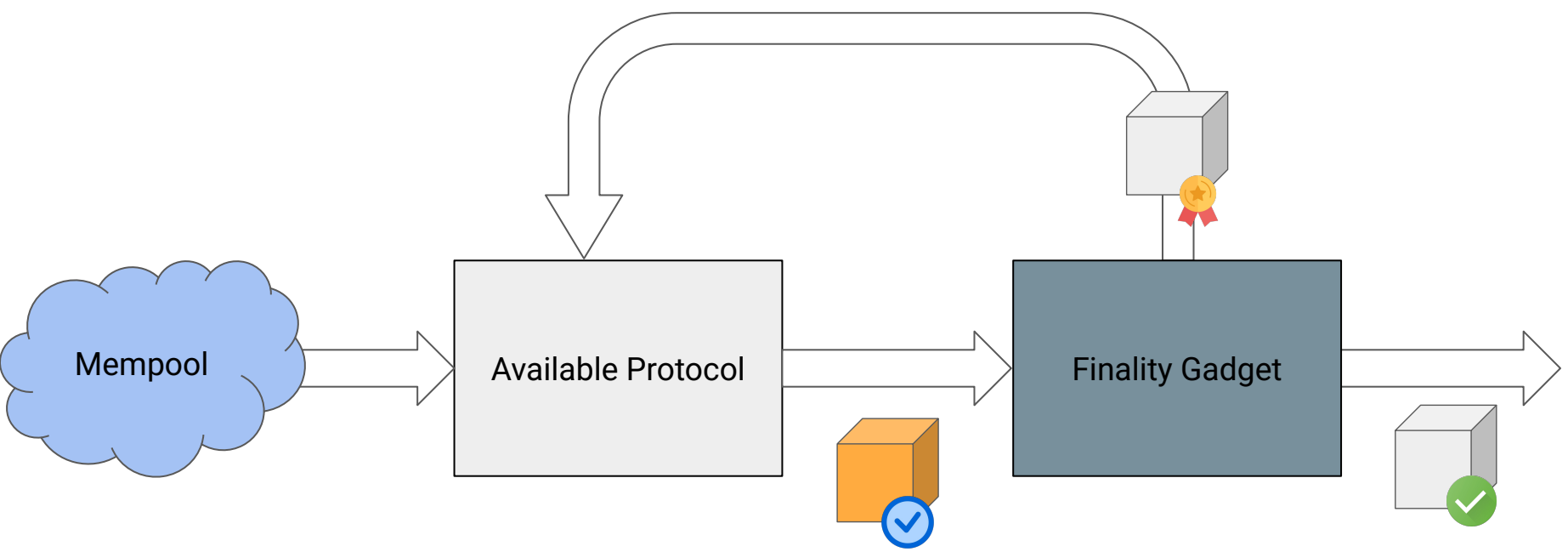
Ertem Nusret Tas
nusret@stanford.edu

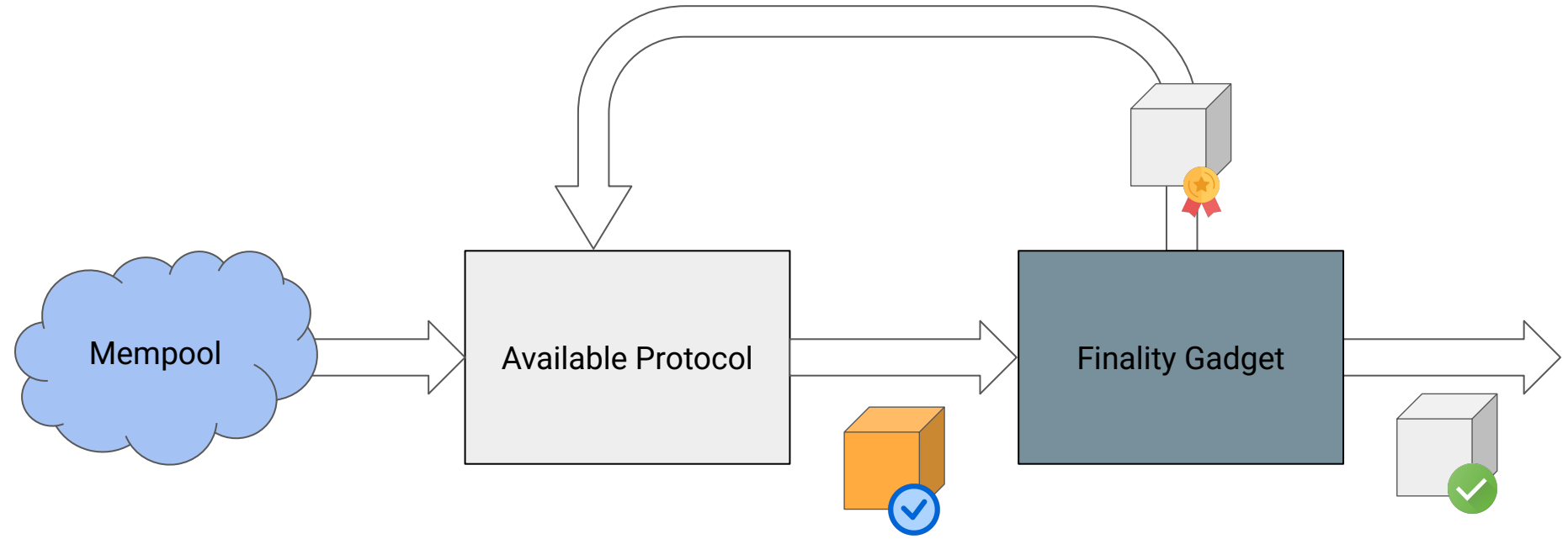
David Tse
dntse@stanford.edu

Abstract—The CAP theorem says that no blockchain can be live under dynamic participation and safe under temporary network partitions. To resolve this availability-finality dilemma, we formulate a new class of flexible consensus protocols, *ebb-and-flow protocols*, which support a full dynamically available ledger in conjunction with a finalized prefix ledger. The finalized ledger falls behind the full ledger when the network partitions but catches up when the network heals. Gasper, the current candidate protocol for Ethereum 2.0's beacon chain, combines the finality gadget Casper FFG with the LMD GHOST fork choice rule and aims to achieve this property. However, we discovered an attack in the standard synchronous network model, highlighting a general difficulty with existing finality-gadget-based designs. We present a construction of provably secure ebb-and-flow protocols with optimal resilience. Nodes run an off-the-shelf dynamically available protocol, take snapshots of the growing available ledger, and input them into a separate off-the-shelf BFT protocol to finalize a prefix. We explore connections with flexible BFT and improve upon the state-of-the-art for that problem.

need to assume all adversary nodes are awake at the beginning [3], [6] or a trusted setup for nodes to join the network [4], [5], but recently it has been shown that these restrictions can be removed using verifiable delay functions [7].

One limitation of dynamically available protocols is that they are not tolerant to network partition: when the network partitions, honest nodes in a dynamically available protocol will think that many nodes are asleep, continue to confirm transactions, and thus is not safe. This is in contrast to permissioned BFT protocols designed for partially synchronous networks, such as PBFT [8], Tendermint [9], [10], Hotstuff [11] and Streamlet [12]. This type of protocols is the basis for permissioned blockchains such as Libra [13], [14] and PoS blockchains such as Algorand [15], [16]. In these protocols, a quorum of two-thirds of the signatures of all the nodes is required to finalize transactions, and hence is safe under





Prefix Property: The available chain is a prefix of the final chain.

Available Chain



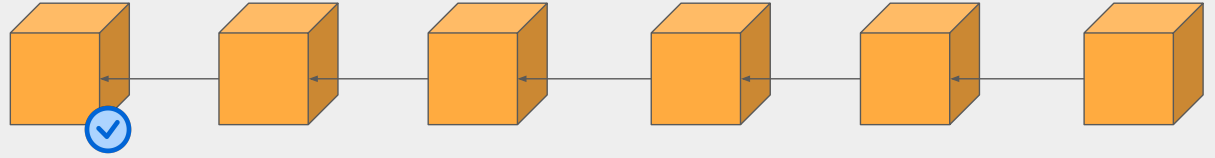
$f < n/2$



Δ (Synchronous)



1 honest awake



Available Chain



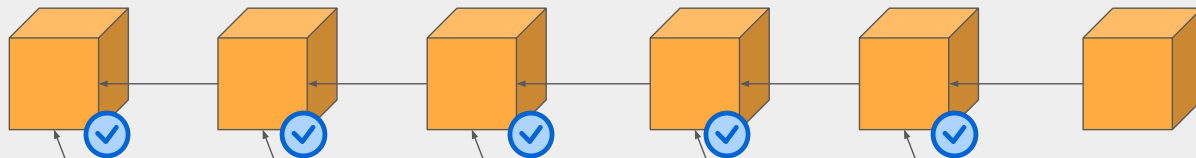
$f < n/2$



Δ (Synchronous)



1 honest awake



Finality Chain



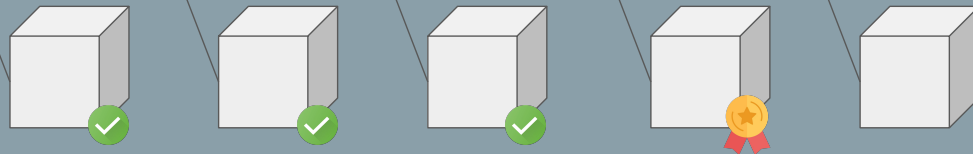
$f < n/3$



Δ after GST (part. sync)



All online after GAT



Available Chain



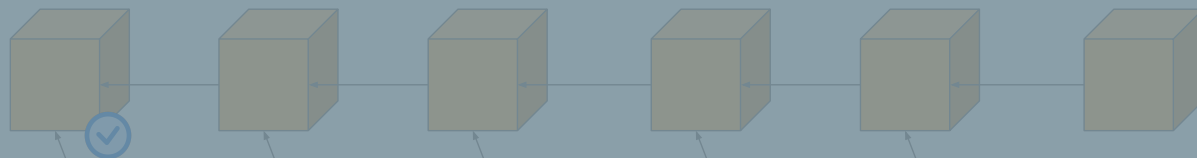
$f < n/2$



Δ (Synchronous)



1 honest awake



2. GOAL: Low-Latency

Finality Chain



$f < n/3$



Δ after GST (part. sync)



All online after GST



Slot 13812415

Epoch 431637



6.3 sec

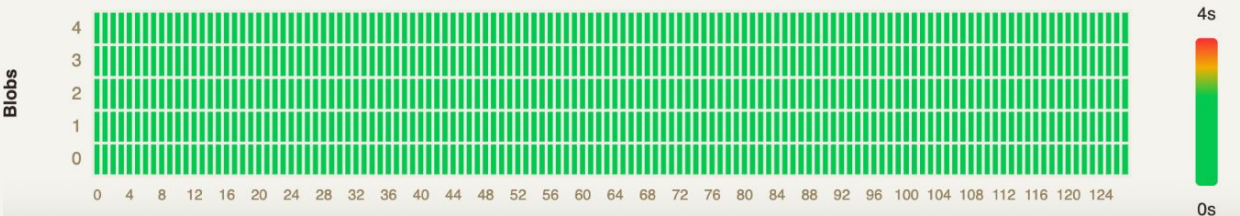


- 6.0s • Attest 3 validators
- 6.1s • Attest 8 validators
- 6.2s • Attest 11 validators
- 6.2s • Attest 1 validator
- 6.3s • Attest 1 validator
- 6.3s • Attest 1 validator
- 6.3s • Attest 2 validators
- 6.4s • Attest 9 validators
- 7.5s • Attest 1 validator
- 9.5s • Attest 1 validator



Data from nodes contributing to Xatu • Not representative of actual Ethereum network distribution

DATA COLUMN AVAILABILITY



ATTESTATION ARRIVALS





Slot 13812415
Epoch 431637

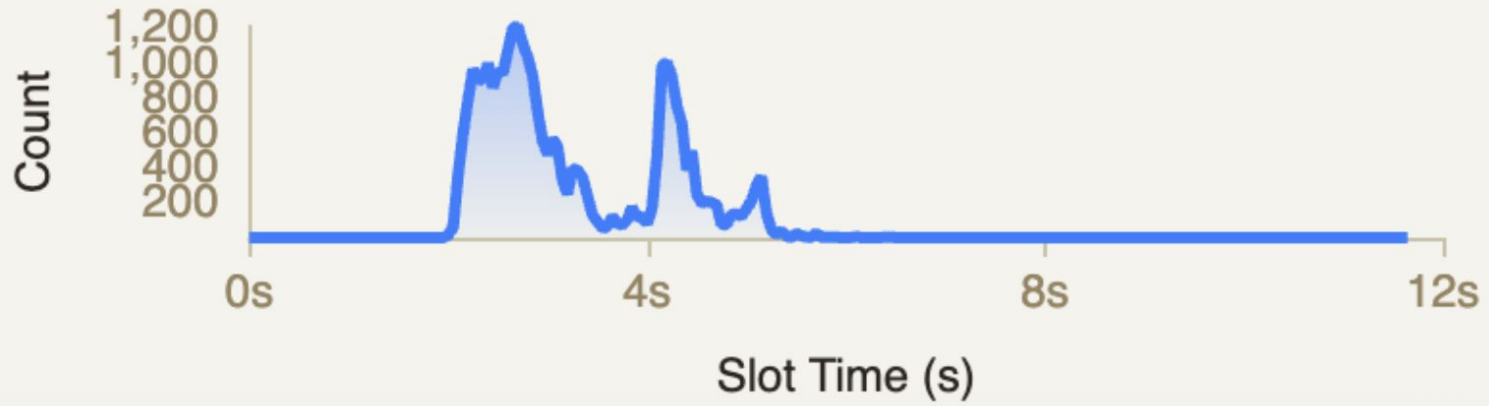
< || > **6.3 sec**

Block | Attestations | Aggregations

0s 4s 8s 12s

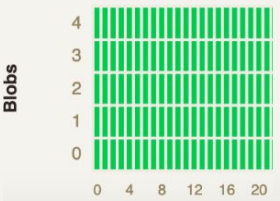
6.0s • Attest 3 validators

ATTESTATION ARRIVALS



Data from nodes contributing

DATA COLUMN AVAILABILITY



Available Chain



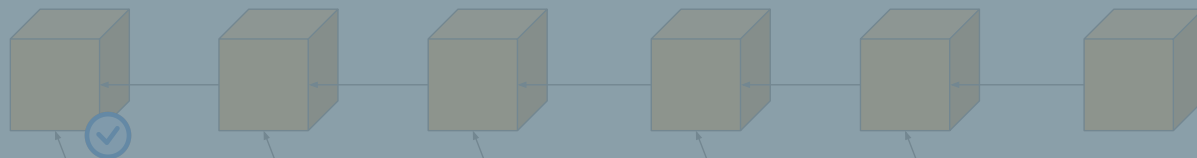
$f < n/2$



Δ (Synchronous)



1 honest awake



2. GOAL: Low-Latency

Finality Chain



$f < n/3$



Δ after GST (part. sync)



All online after GAT



Available Chain



$f < n/2$



Δ (Synchronous)



1 honest awake

Finality Chain



$f < n/3$



Δ after GST (part. sync)



All online after GAT



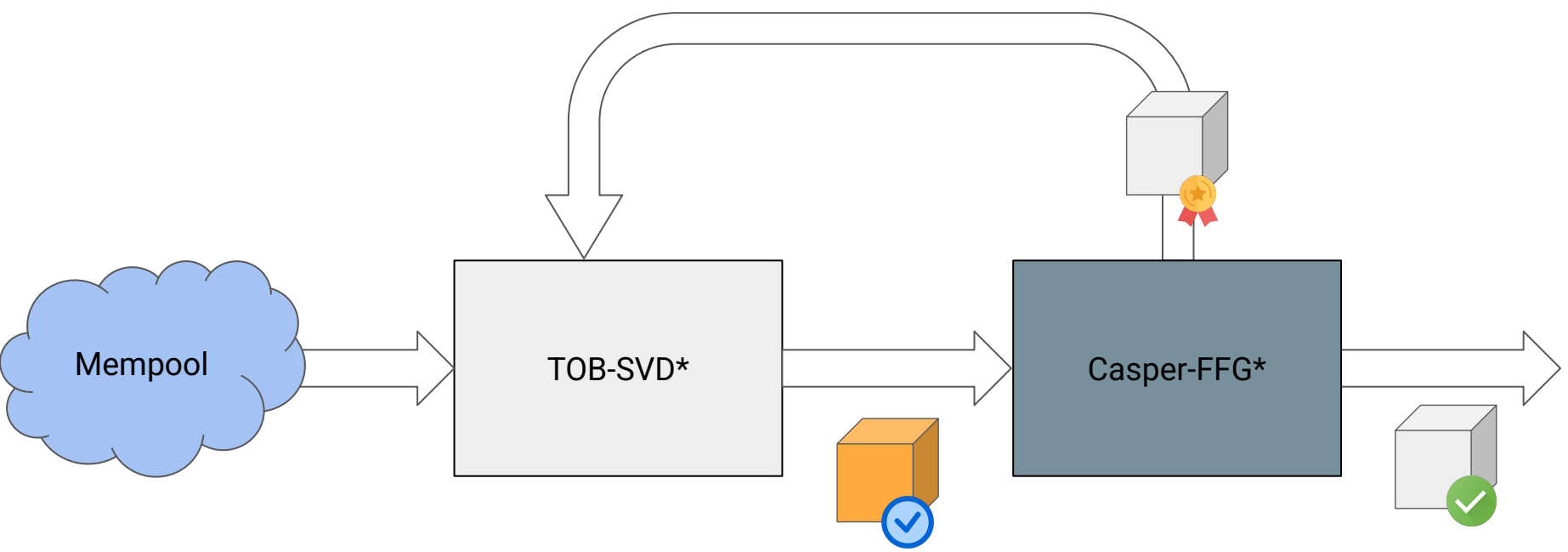
2. Goal: Low-Latency

Few Voting Rounds

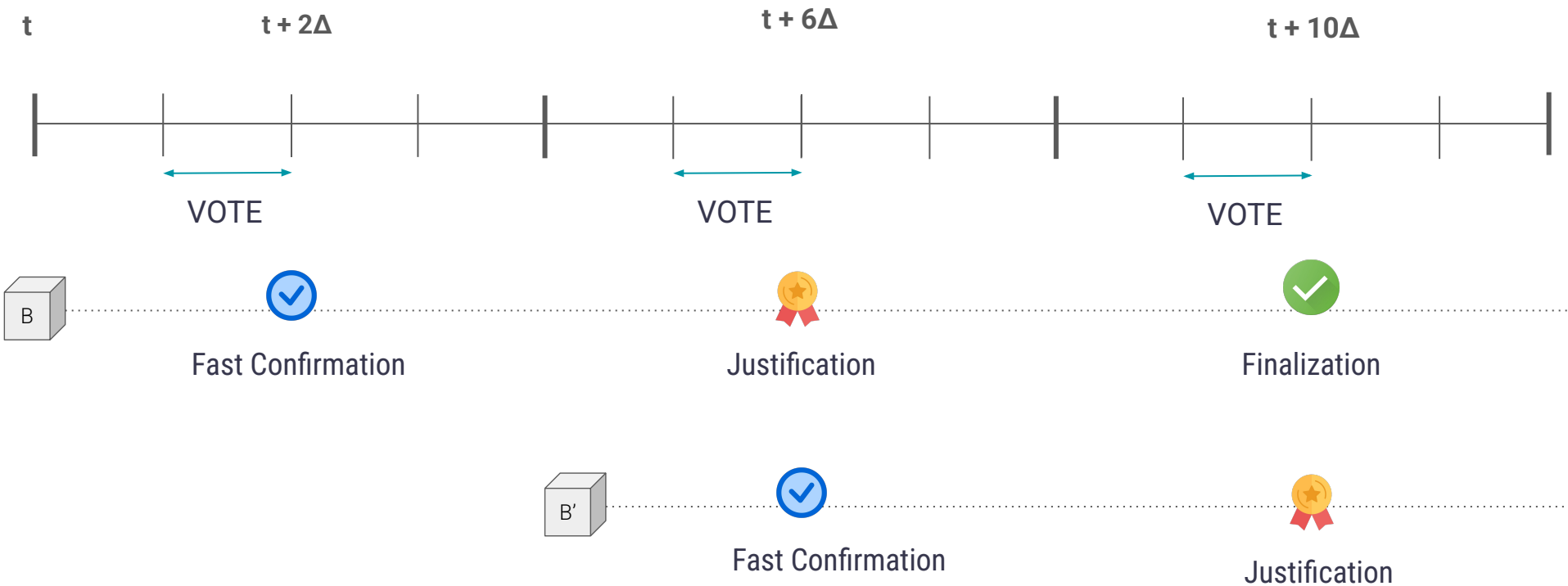


Majorum: Ebb-and-Flow Consensus with Dynamic Quorums

Francesco D'Amato, Roberto Saltini, Thanh-Hai Tran, Yann Vonlanthen, Luca Zanolini
Financial Cryptography and Data Security (FC) 2026, St. Kitts



Majorum



Ex-Ante Reorgs (LMD-GHOST)

VOTE

$t + 6\Delta$

VOTE

$t + 10\Delta$

VOTE

Balancing Attacks (Interaction)

Justification

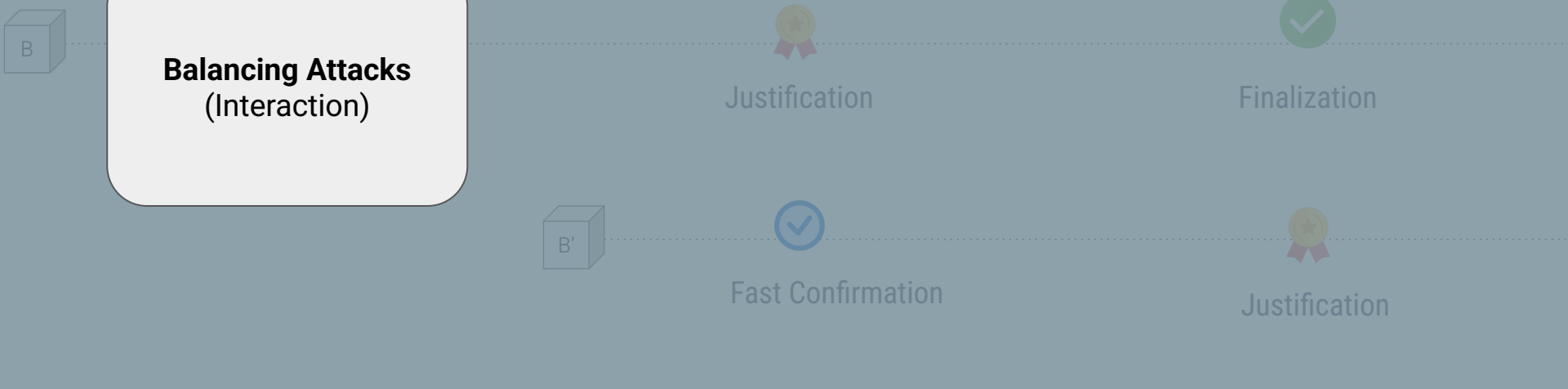
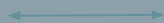
Finalization

Fast Confirmation

Justification



t



Ex-Ante Reorgs
(LMD-GHOST)

VOTE

Balancing Attacks
(Interaction)



$t + 6\Delta$

VOTE

Justification

Fast Confirmation

$t + 10\Delta$

VOTE

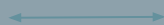
Finalization

Justification

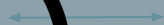


Ex-Ante Reorgs
(LMD-GHOST)

VOTE



VOTE



Block production
vs.
Time to finality



Balancing Attacks
(Interaction)

Justification



Finalization



Fast Confirmation



Justification



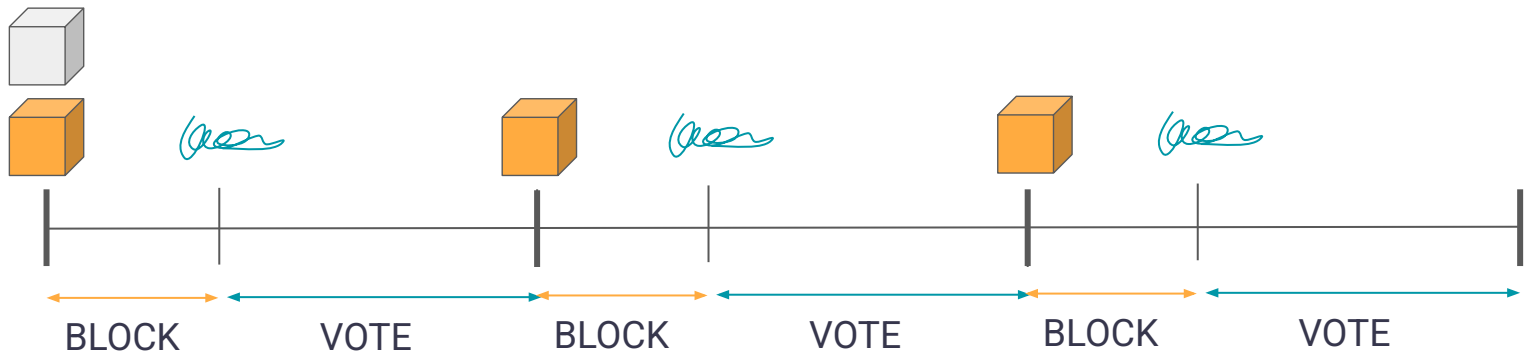
$t + 6\Delta$

$t + 10\Delta$

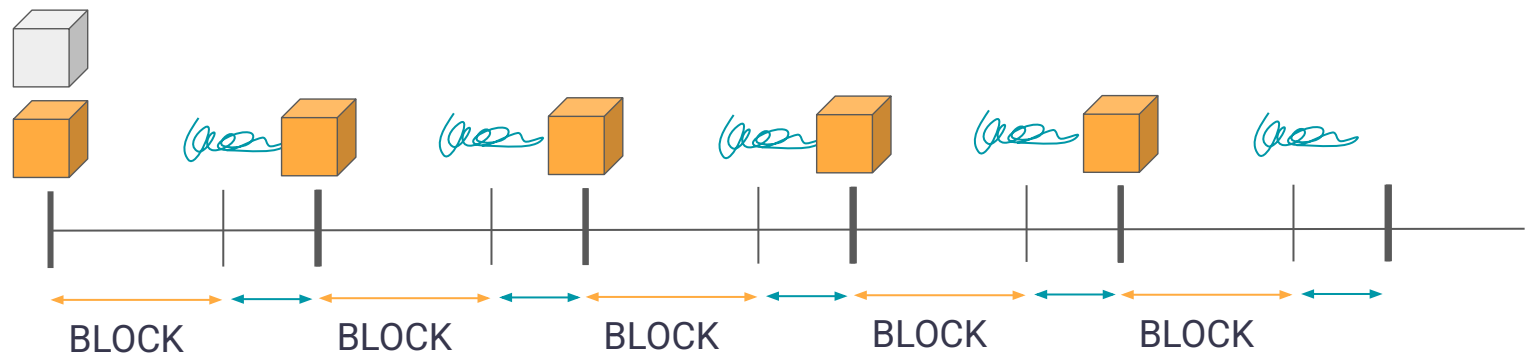
t

Coupled timelines in Gasper

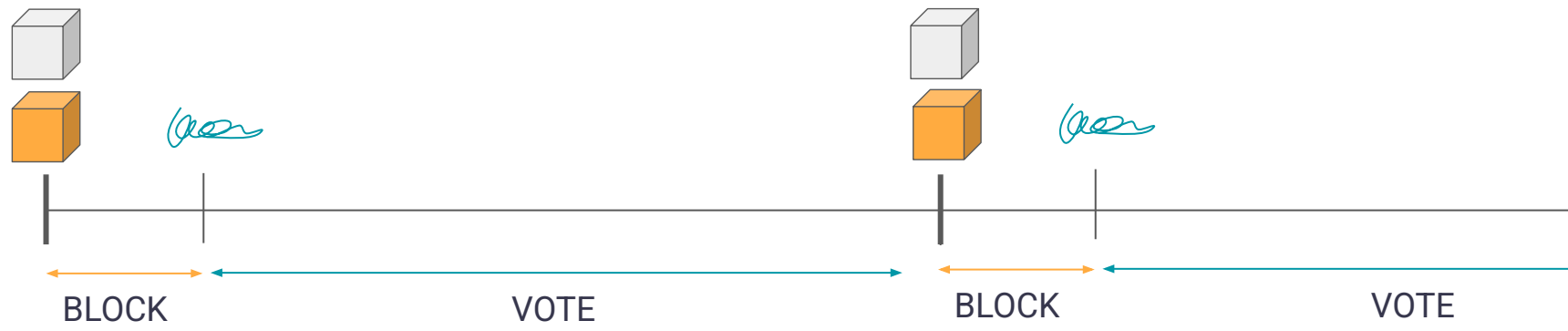
1M Attestations
of ~100B over 32 slots!



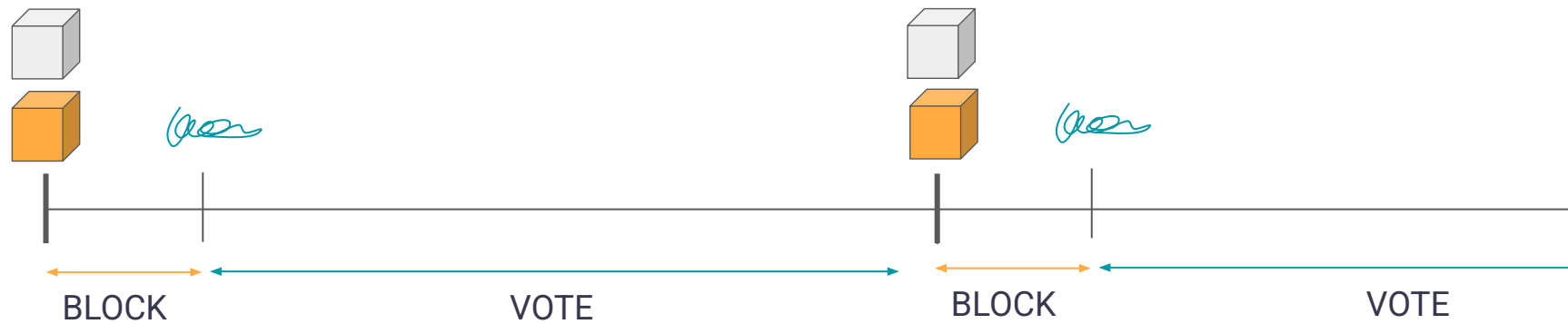
Small committees
→ long epochs
→ slow finality



Large committees
→ long slots
→ slow block production



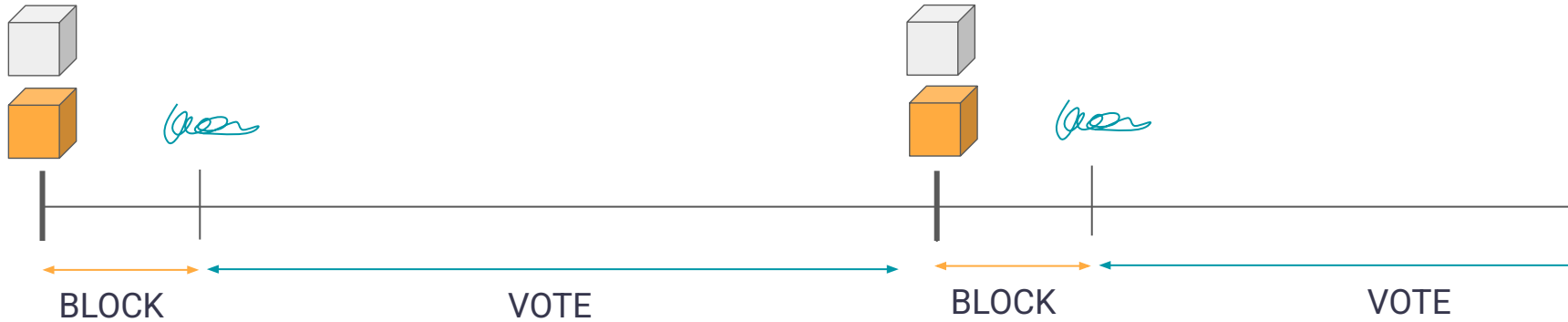
Large committees
→ long slots
→ slow block production



Block production and finality timeline are coupled!

Coupled timelines in Majorum

Large committees
→ long slots
→ slow block production



Block production and finality timeline are coupled!

Coupled timelines in Majorum

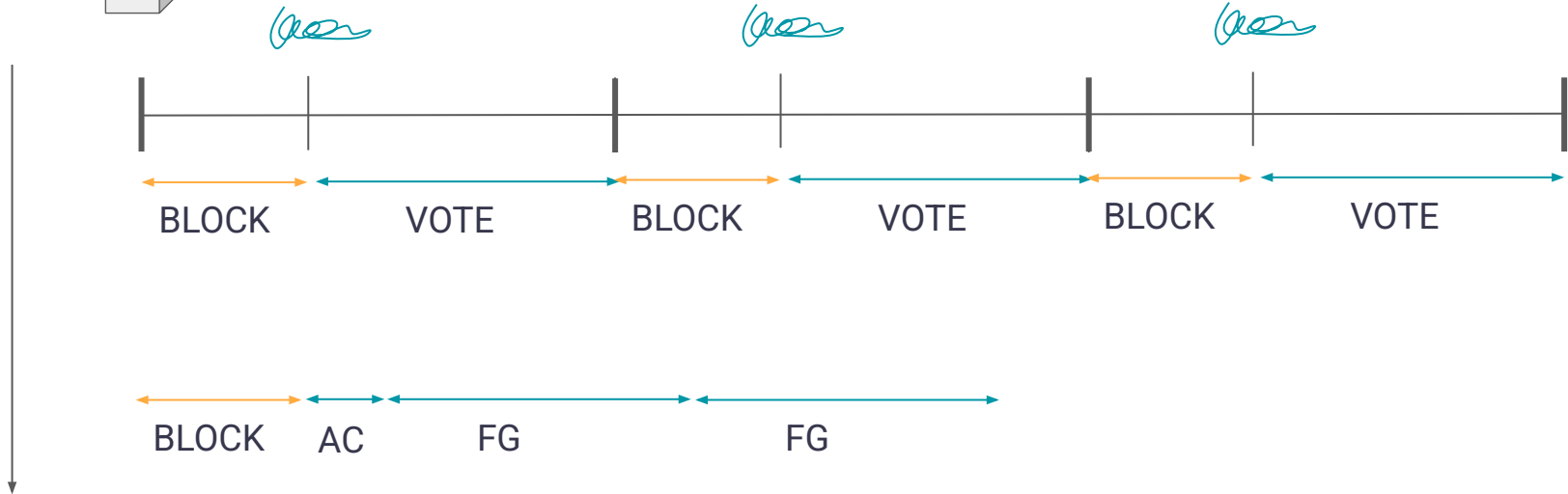
❖ **Finality slows down block production**

- full voting round in the critical path
- need small validator set to have fast slots

❖ **Block production slows down finality**

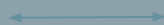
- Slot isn't just voting (block propagation, execution, proposer synchronization mechanisms)

Coupled timelines in Majorum



Ex-Ante Reorgs
(LMD-GHOST)

VOTE



$t + 6\Delta$

VOTE



$t + 10\Delta$

Block production
vs.
finality trade-off



Balancing Attacks
(Interaction)

Justification



Finalization



Fast Confirmation



Justification



t



Ex-Ante Reorgs
(LMD-GHOST)

VOTE

Balancing Attacks
(Interaction)

$t + 6\Delta$

VOTE

Justification

Fast Confirmation

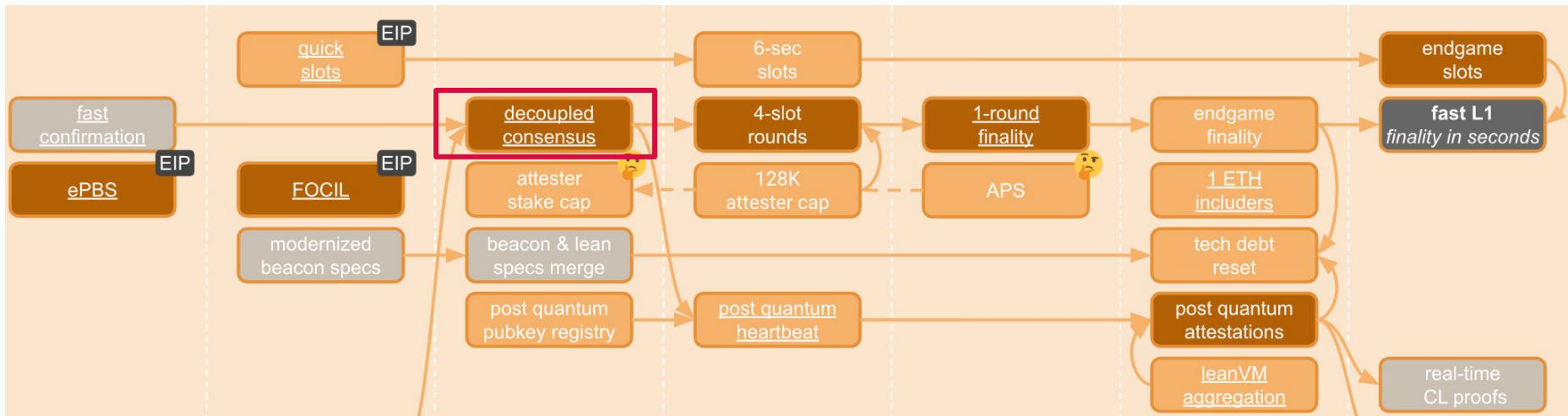
Block production
vs.
finality trade-off

$t + 10\Delta$

Finalization

Justification



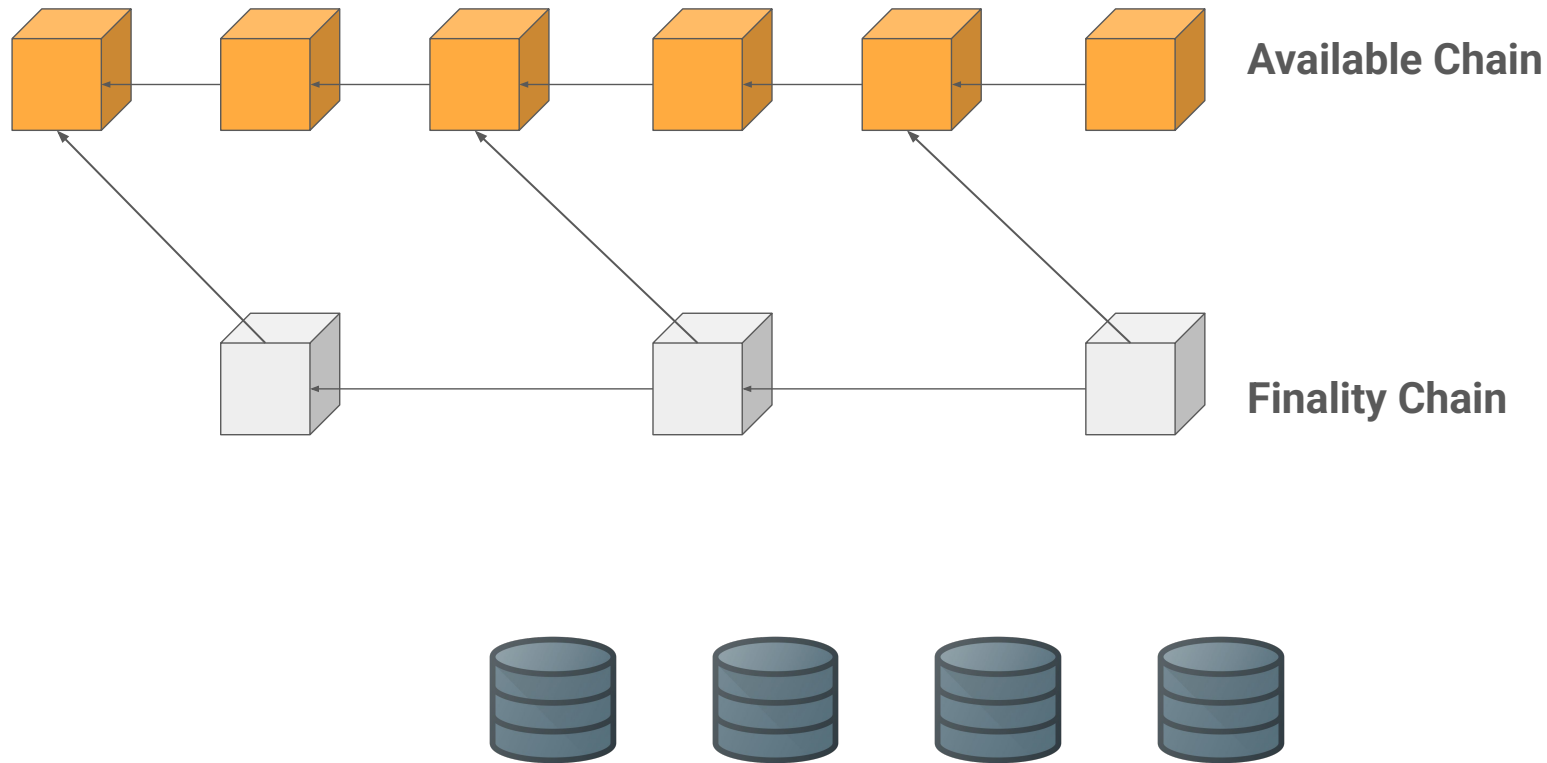


Consensus Layer

Decoupled Consensus



Decoupled Consensus



LMD GHOST with ~256 validators and a fast-following finality gadget



vbuterin

3 Aug 1

Epistemic status: early exploration

Recently, there has been discussion about more aggressive ways to reduce Ethereum's slot time. This can be done in two ways:

1. Reducing the δ parameter (our assumption on maximum expected network latency). This can only be done safely if we get improvements at the p2p layer that reduce latency
2. Re-architecting the slot structure to reduce the number of network latency rounds in one slot.

There is significant p2p hardening and optimization work going on to enable (1); the top



Quote



Copy Quote



Share

is erasure coding. Research work is focusing on (2).

This post will argue that the optimal approach to (2) may be to move somewhat away from the tight coupling between slots and finality introduced in 3SF ¹⁴, and instead have a more separate LMD GHOST fork choice rule and finality gadget, with different participant counts.

First, let's look at the current slot structure ([source](#) ¹³):

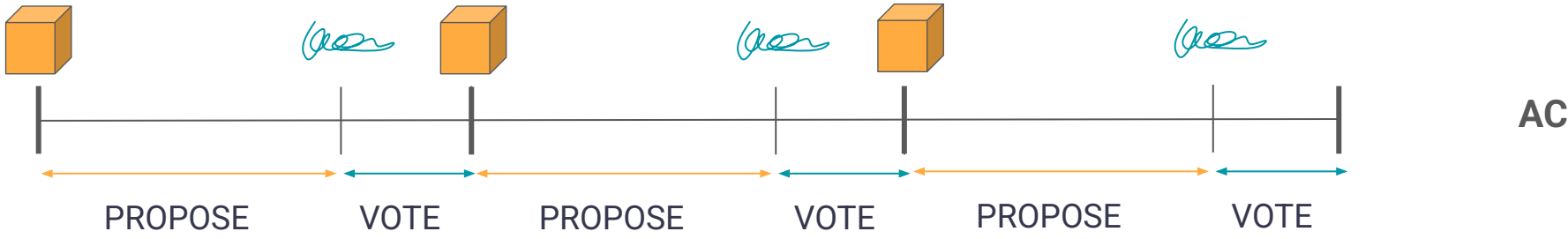
Aug 1

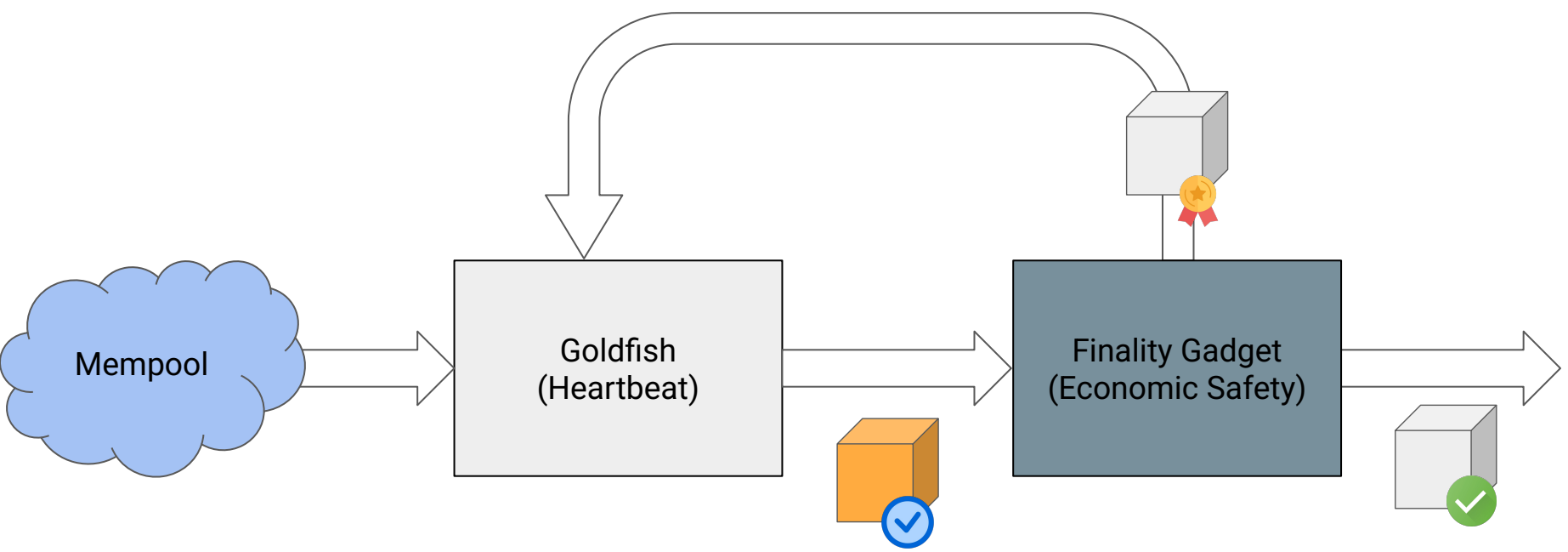
1 / 21

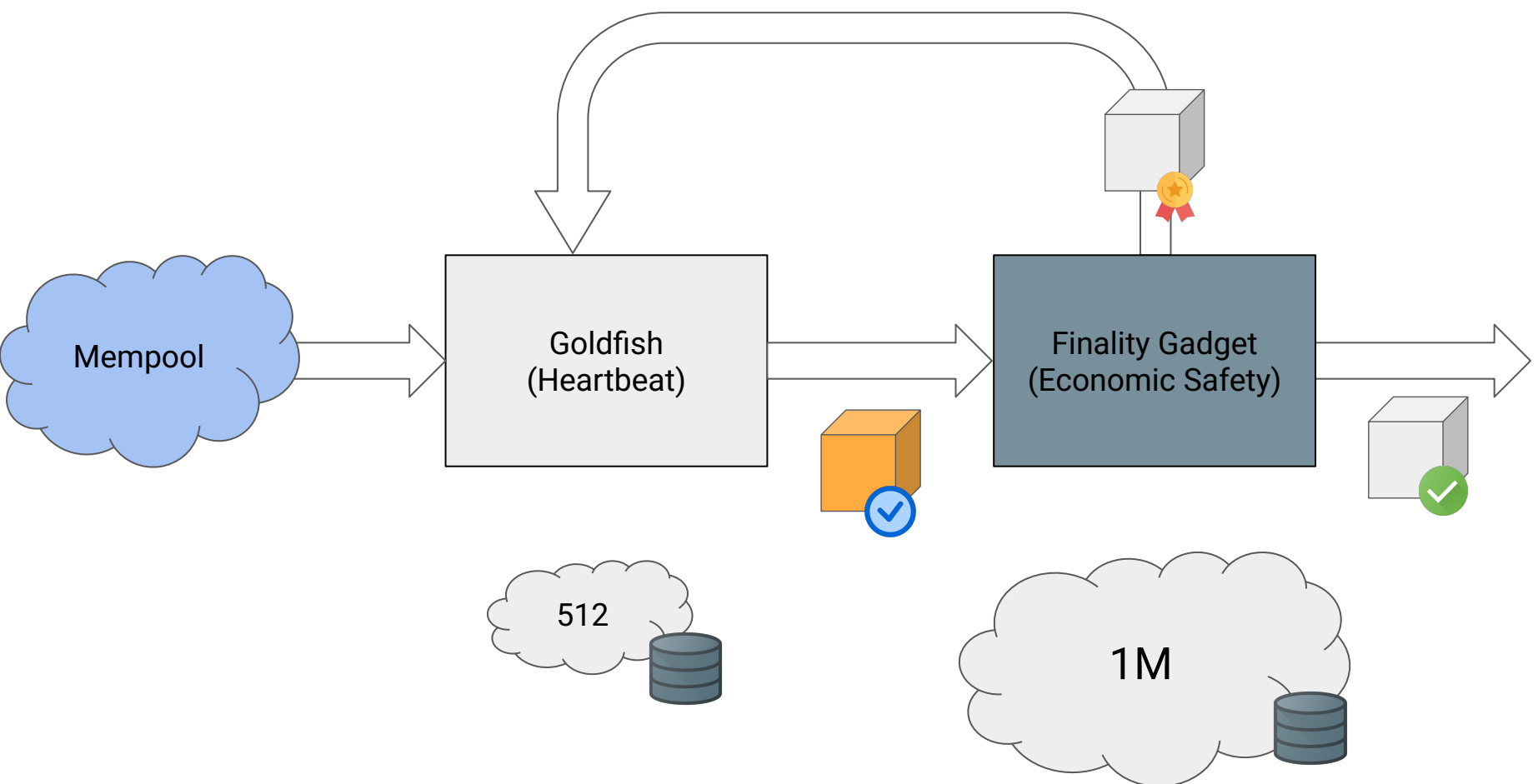
Aug 1

Sep 29









Available Chain



Goldfish



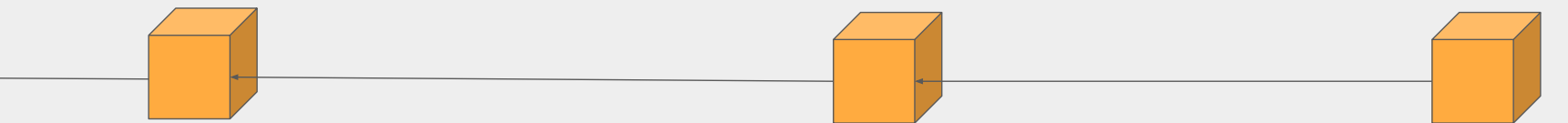
$f < n/2$



Δ (Synchronous)



1 honest awake



Propose

Vote

Fast Confirm

t

$t + \Delta$

$t + 2\Delta$

$t + 3\Delta$



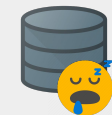
Goldfish



$f < n/2$



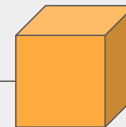
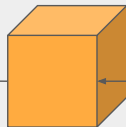
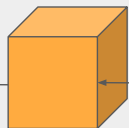
Δ (Synchronous)



1 honest awake



Secret Committee Election



Propose

Vote

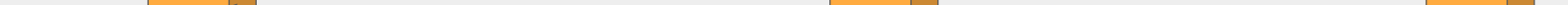
Fast Confirm

t

t + Δ

t + 2 Δ

t + 3 Δ



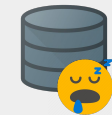
Goldfish



$f < n/2$

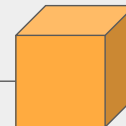
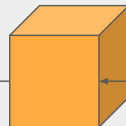
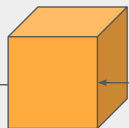


Δ (Synchronous)



1 honest awake

Secret Committee Election



Propose

Vote

Fast Confirm

t

t + Δ

t + 2 Δ

t + 3 Δ



Fast confirm with 512 votes!

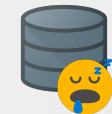
Goldfish



$f < n/2$

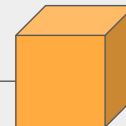
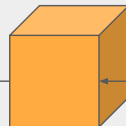
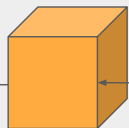


Δ (Synchronous)



1 honest awake

Secret Committee Election



Propose

Vote

Fast Confirm

Rotate committee at every slot!

t

$t + \Delta$

$t + 2\Delta$

$t + 3\Delta$



Fast confirm with 512 votes!

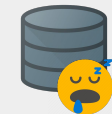
Goldfish



$f < n/2$

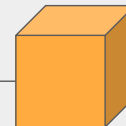
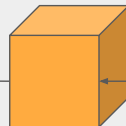
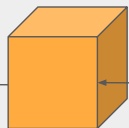


Δ (Synchronous)



1 honest awake

Secret Committee Election



Propose

Vote

Fast Confirm

Rotate committee at every slot!

t

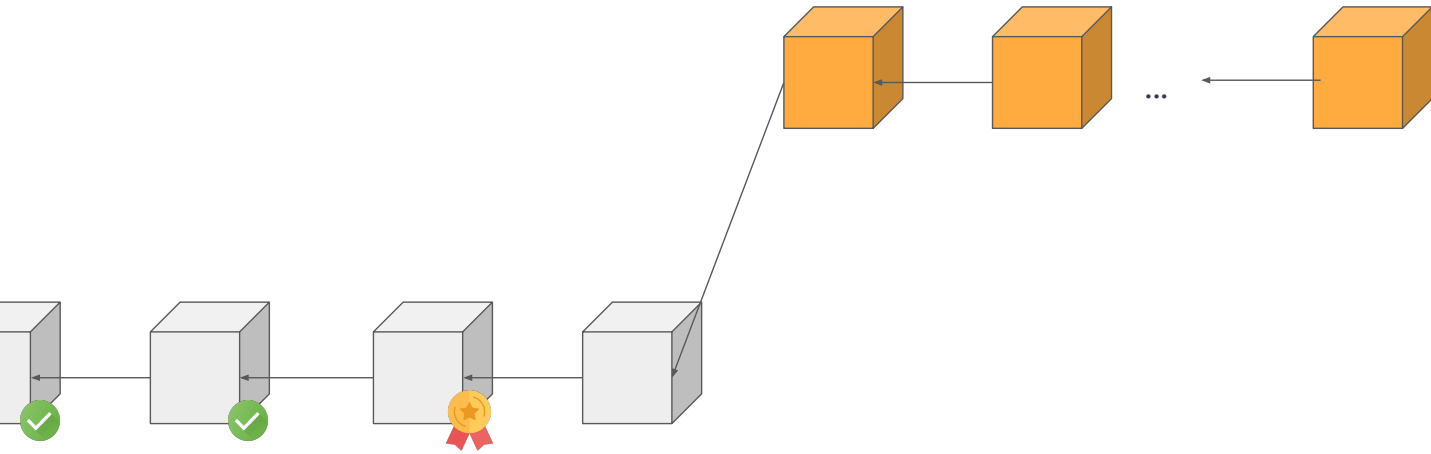
$t + \Delta$

$t + 2\Delta$

$t + 3\Delta$



Fast confirm with 512 votes!



Finality Gadget



Finality Gadget



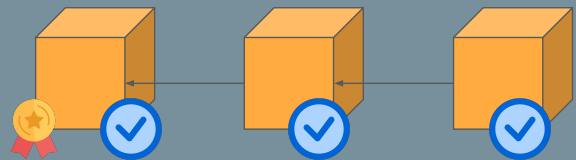
$f < n/3$



Δ after GST (part. sync)



All online after GAT



Pick voting target:

Confirmed head of available chain
(respecting justification)

Finality Gadget



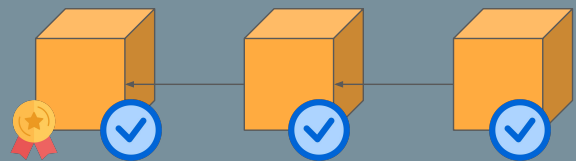
$f < n/3$



Δ after GST (part. sync)



All online after GAT



Pick voting target:

Confirmed head of available chain
(respecting justification)

Finality Gadget



$f < n/3$



Δ after GST (part. sync)



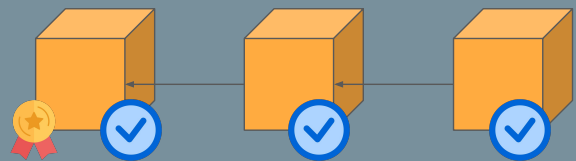
All online after GAT



Pick voting target:

Confirmed head of available chain
(respecting justification)

⇒ fast confirmed



Finality Gadget



$f < n/3$



Δ after GST (part. sync)



All online after GAT

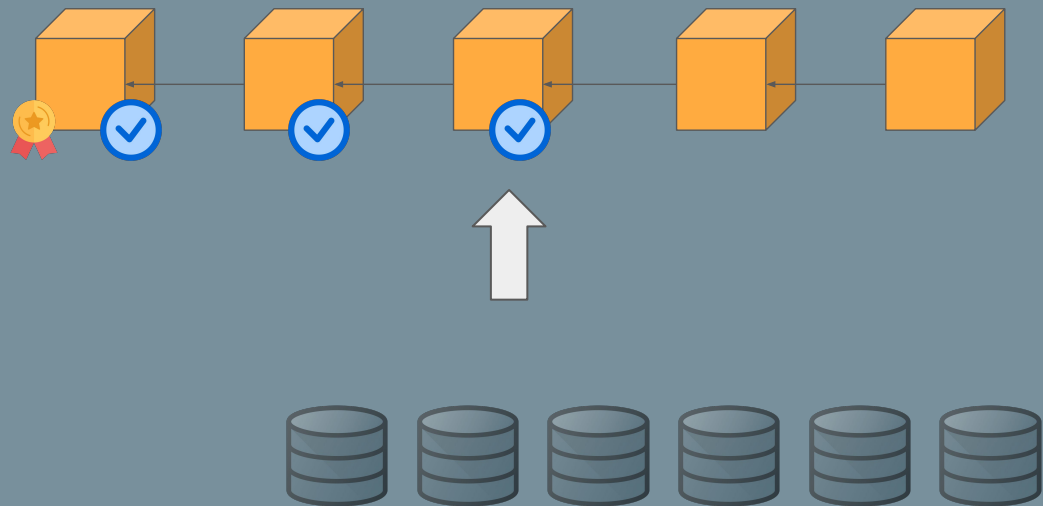


Pick voting target:

Confirmed head of available chain
(respecting justification)

⇒ fast confirmed

⇒ k - deep



Finality Gadget



$f < n/3$



Δ after GST (part. sync)



All online after GAT



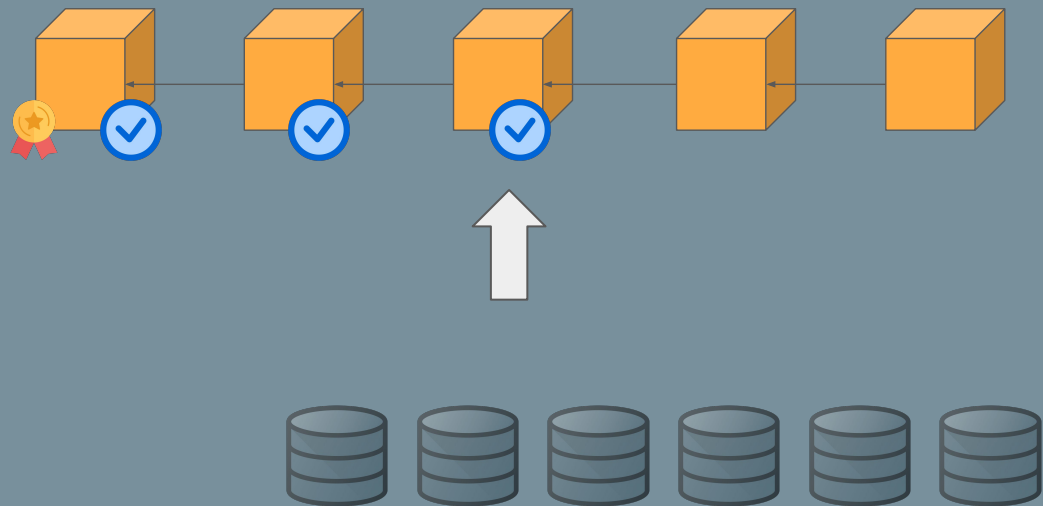
Pick voting target:

Confirmed head of available chain
(respecting justification)

⇒ fast confirmed

⇒ k - deep

(under sleepiness, or asynchrony)



Finality Gadget



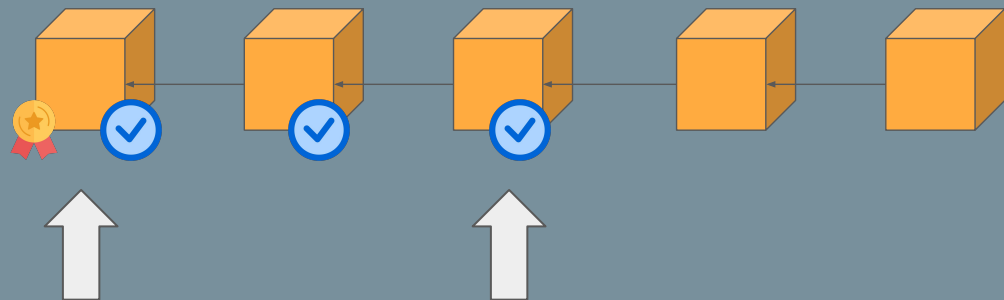
$f < n/3$



Δ after GST (part. sync)



All online after GAT



Pick voting target:

Confirmed head of available chain
(respecting justification)

⇒ fast confirmed

⇒ k - deep

(under sleepiness, or asynchrony)



Pick voting source:

Latest justified block

Finality Gadget



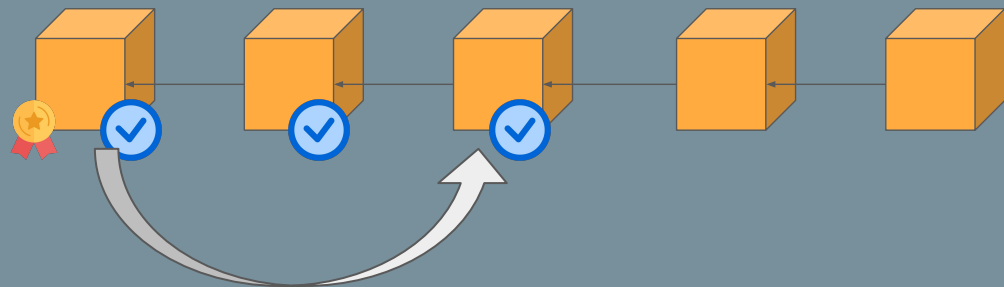
$f < n/3$



Δ after GST (part. sync)



All online after GAT



Pick voting target:

Confirmed head of available chain
(respecting justification)

⇒ fast confirmed

⇒ k - deep

(under sleepiness, or asynchrony)



Pick voting source:

Latest justified block

Finality Gadget



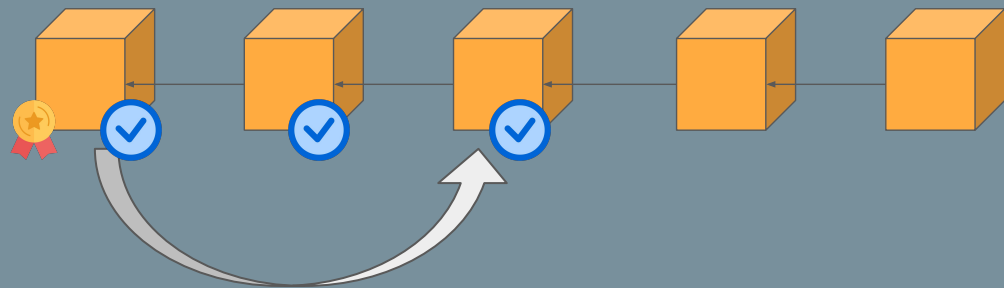
$f < n/3$



Δ after GST (part. sync)



All online after GAT



Justify block B once:
 $\geq 67\%$ votes with target B



Finality Gadget



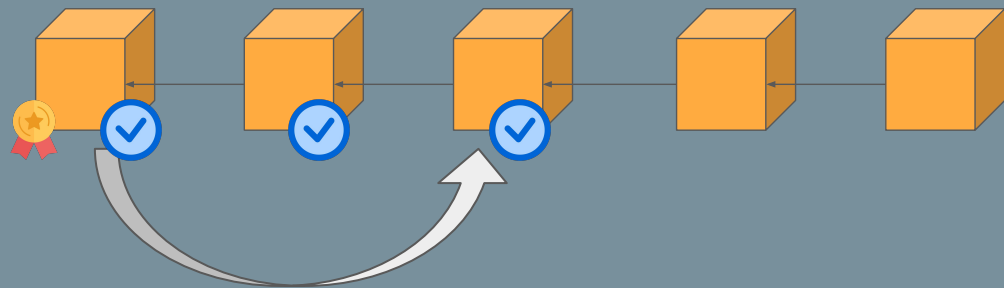
$f < n/3$



Δ after GST (part. sync)



All online after GAT



Justify block B once:
 $\geq 67\%$ votes with target B



Finality Gadget



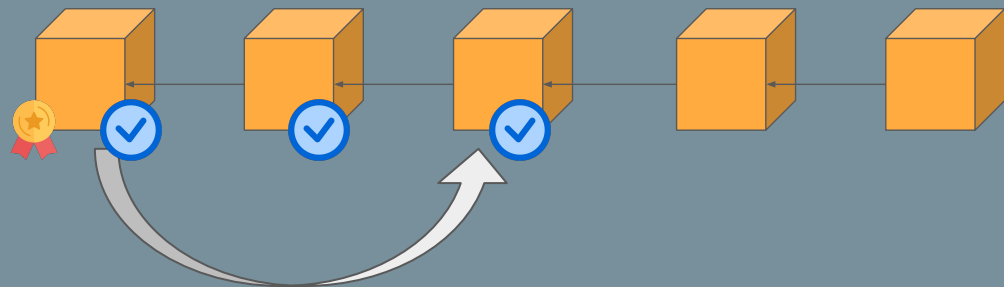
$f < n/3$



Δ after GST (part. sync)



All online after GAT



Justify block B once:
 $\geq 67\%$ votes with target B



Finalize block B once:
 $\geq 67\%$ votes with source B and target $B' > B$



Finality Gadget



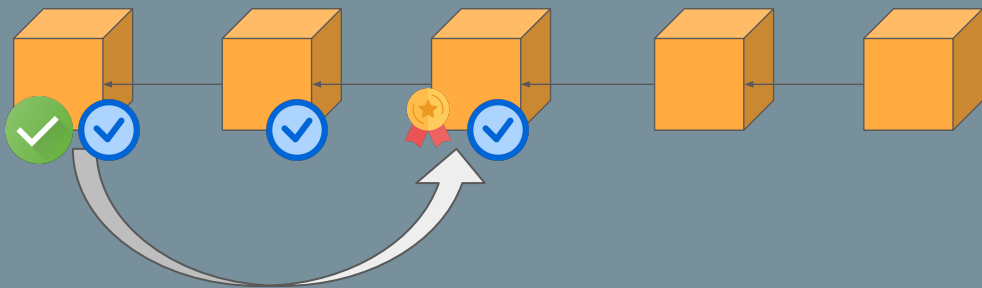
$f < n/3$



Δ after GST (part. sync)



All online after GAT

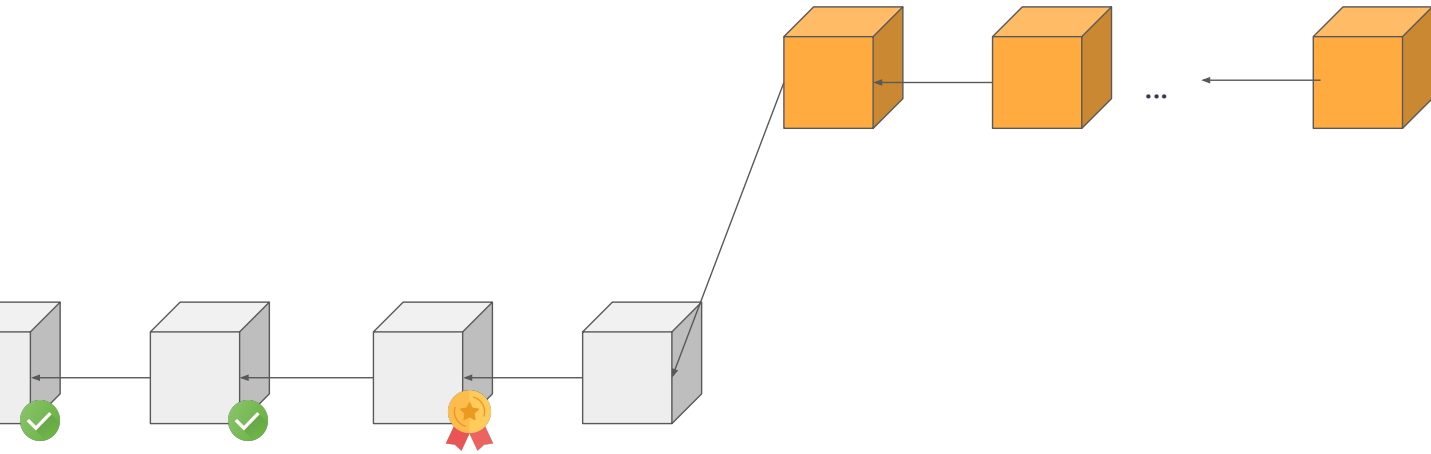


Justify block B once:
 $\geq 67\%$ votes with target B

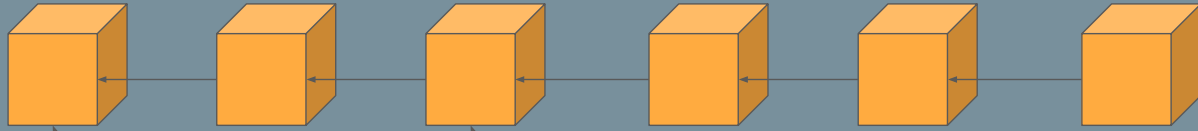


Finalize block B once:
 $\geq 67\%$ votes with source B and target $B' > B$

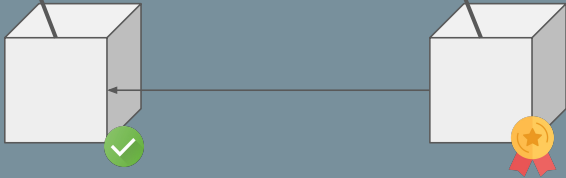




Stabilization Gadget

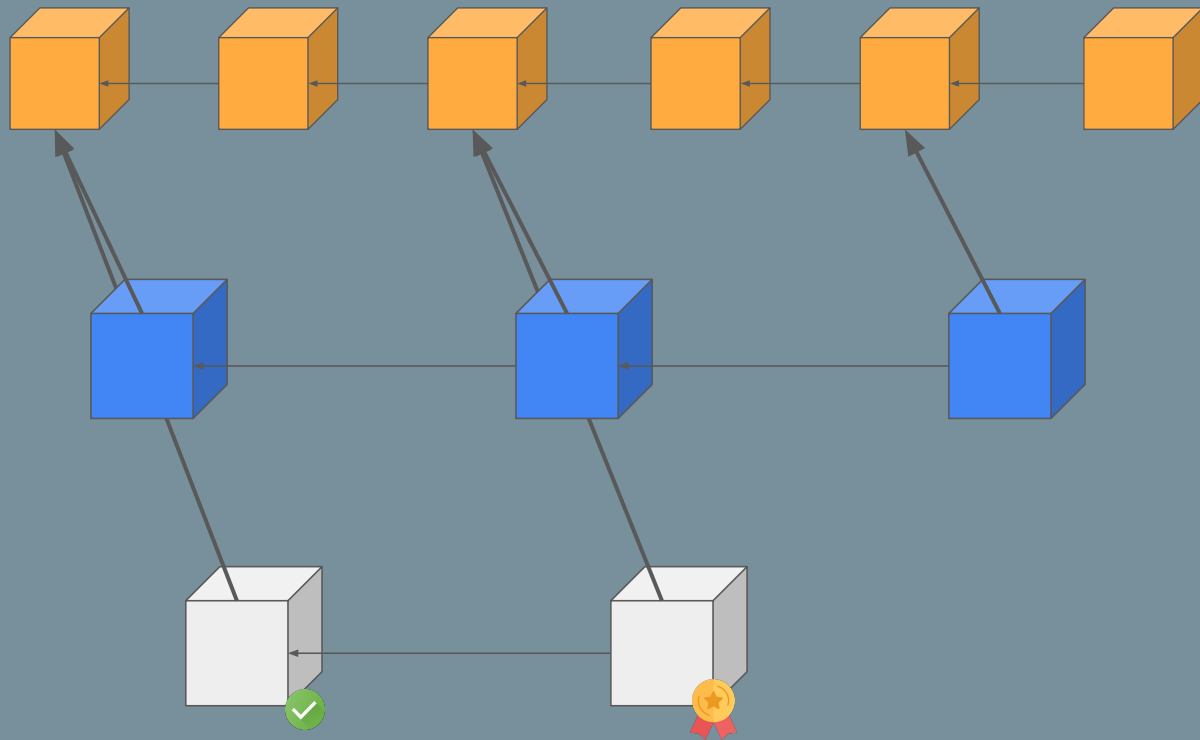


Goldfish (AC)



BFT (FG)

Stabilization Gadget



Goldfish (AC)

RLMD-GHOST (AR)

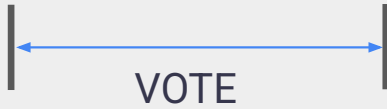
BFT (FG)

Advantages

Incremental Improvements

- consolidation
- p2p
- privacy

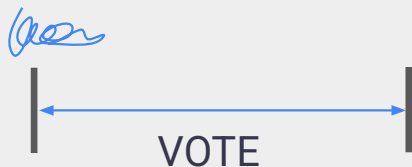
Qwen



Advantages

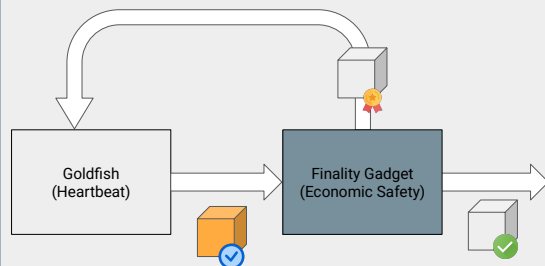
Incremental Improvements

- consolidation
- p2p
- privacy



Flexible

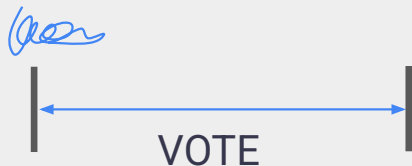
- hard work up front
- swap components later



Advantages

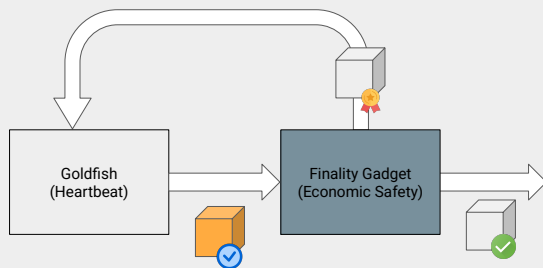
Incremental Improvements

- consolidation
- p2p
- privacy



Flexible

- hard work up front
- swap components later



PQ Ready

- Available chain doesn't need aggregation
- possible short term alternative



Unblocking faster finality with decoupled consensus

Consensus



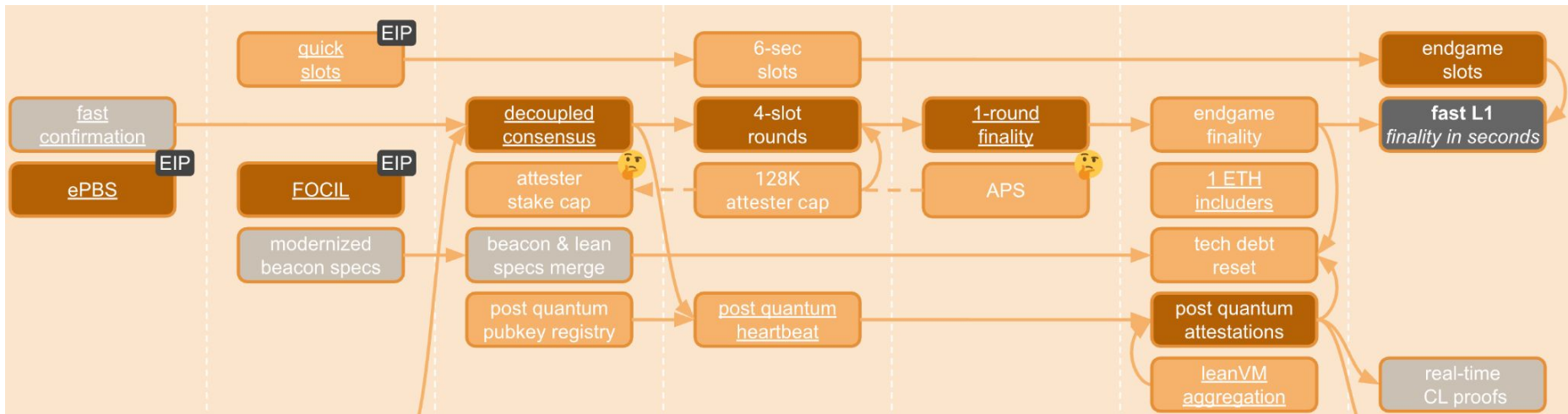
fradamt

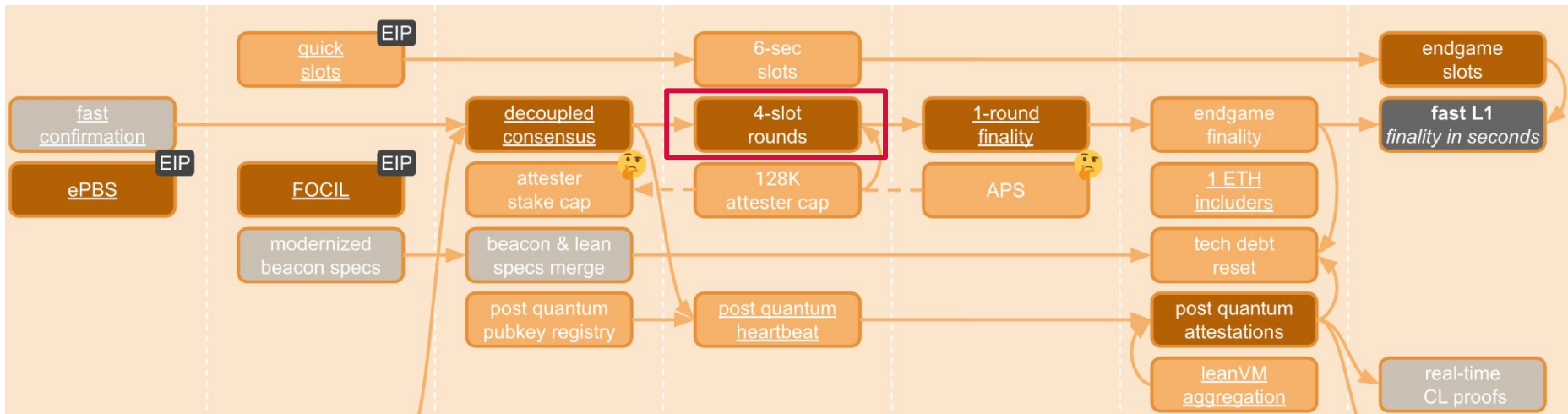
9  13h

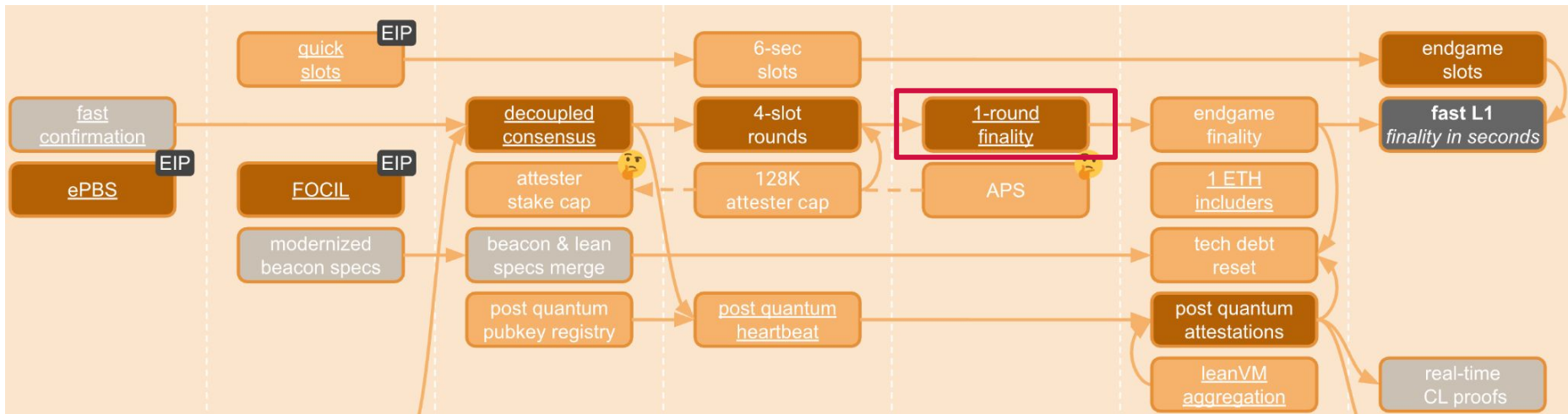
Thanks to the EF Consensus team for discussion and comments

In the current consensus protocol, **the timelines of block production and finality are coupled**. A large committee (consisting of $1/32$ of the whole validator set) attests in every slot, in the critical path. Shrinking the size of this committee, for example to $1/64$ of the validator set, would enable faster slots, but it would slow down finality, because completing a full voting round (one epoch) would now accrue the overhead of 64 slots instead of 32. At one extreme of this tradeoff space, only a single validator votes in every slot, and a full voting round is only completed once every 1M slots or so, which would take more than a day even with 100ms slots. Even for less extreme committee sizes (and resulting epoch lengths), the overhead from doing “finality via slot-by-slot accumulation” cannot be discounted.

On the other hand, increasing the size of the committee would make finality faster, at the cost of slower block production (longer slots). At this other extreme of the tradeoff space, the whole validator set votes in every slot, and slot time largely depends on how fast we can aggregate it.

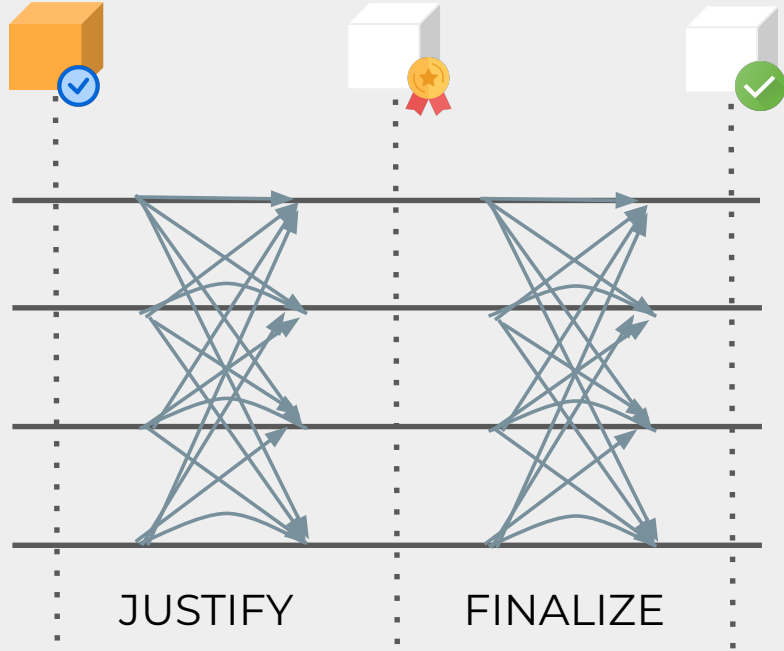


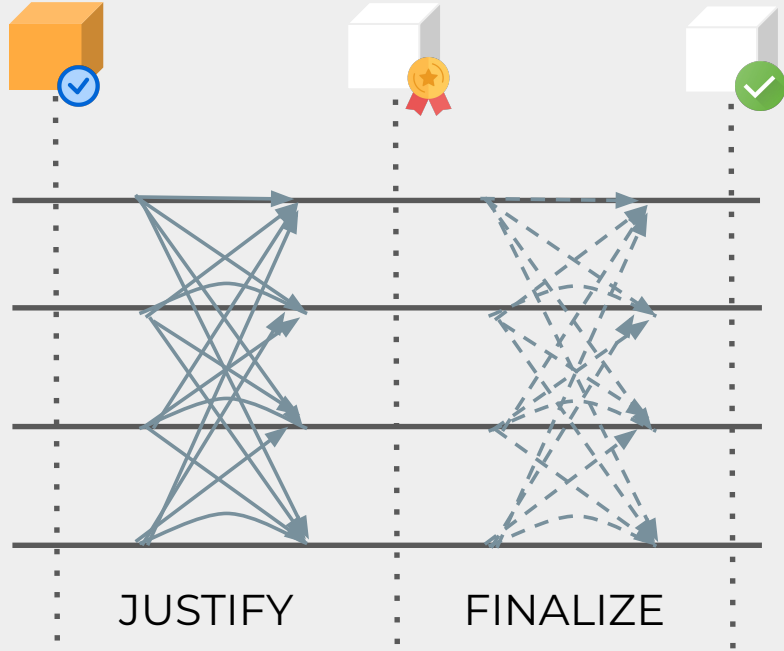


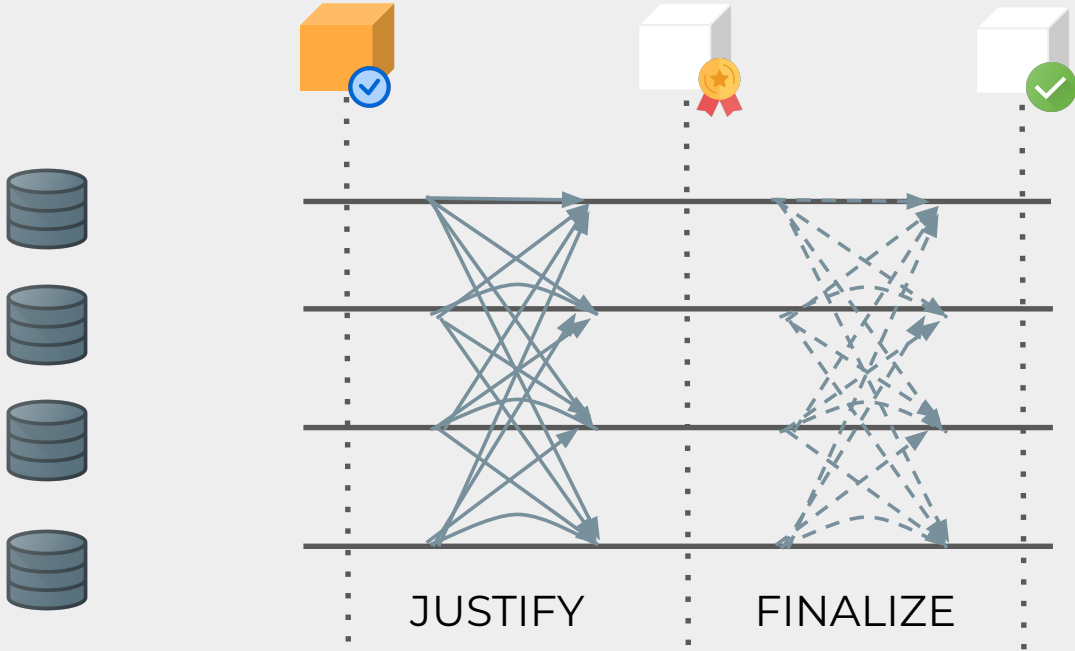


One-Round Finality



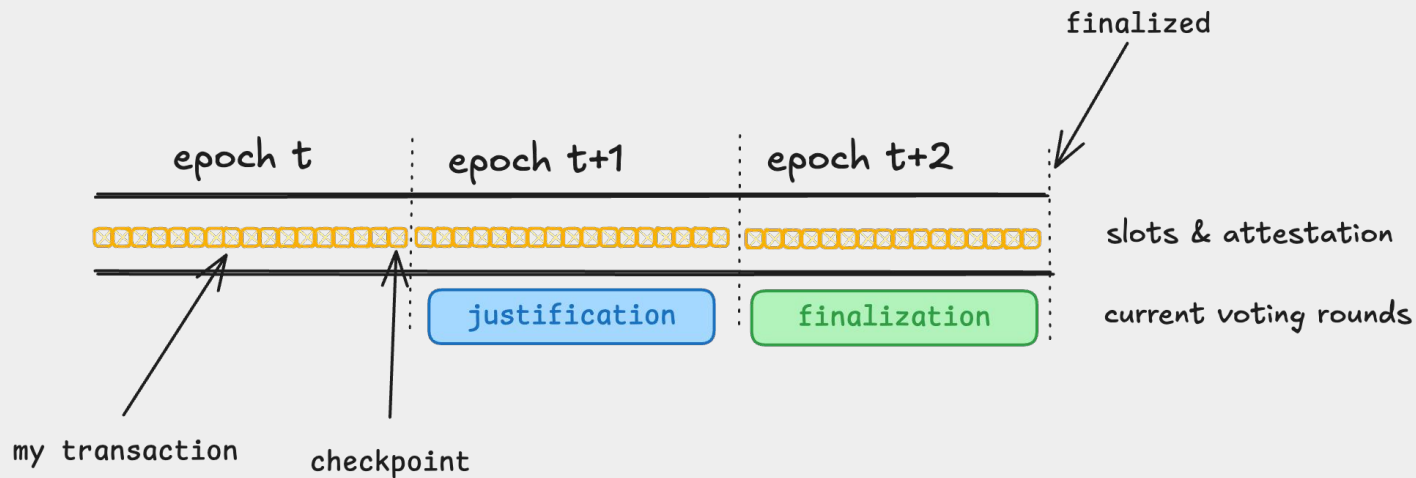




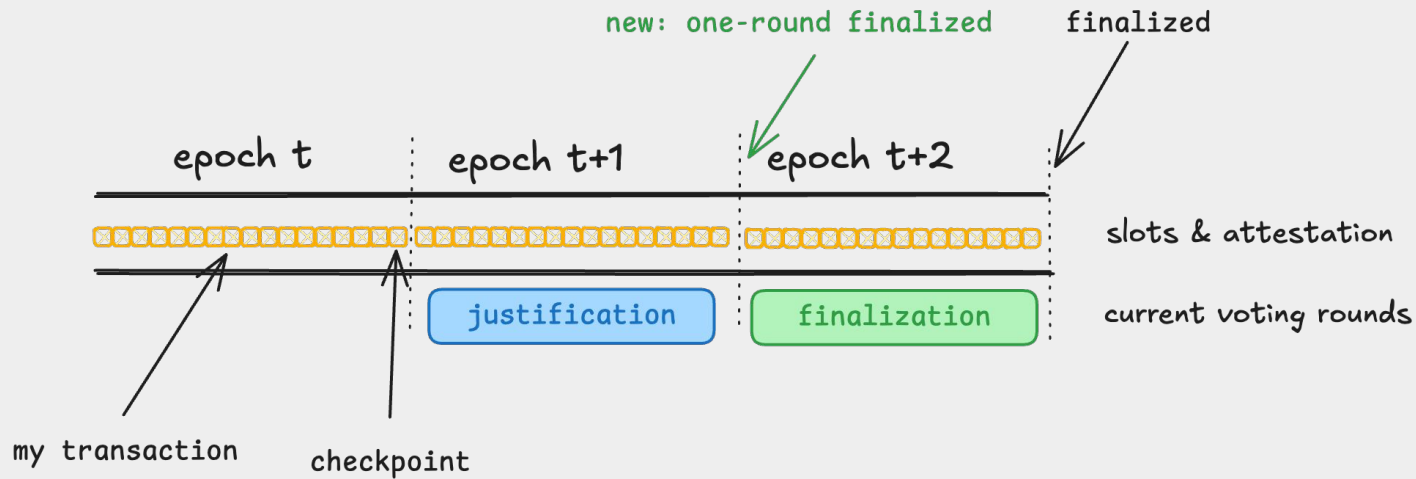


One-Round
Finality

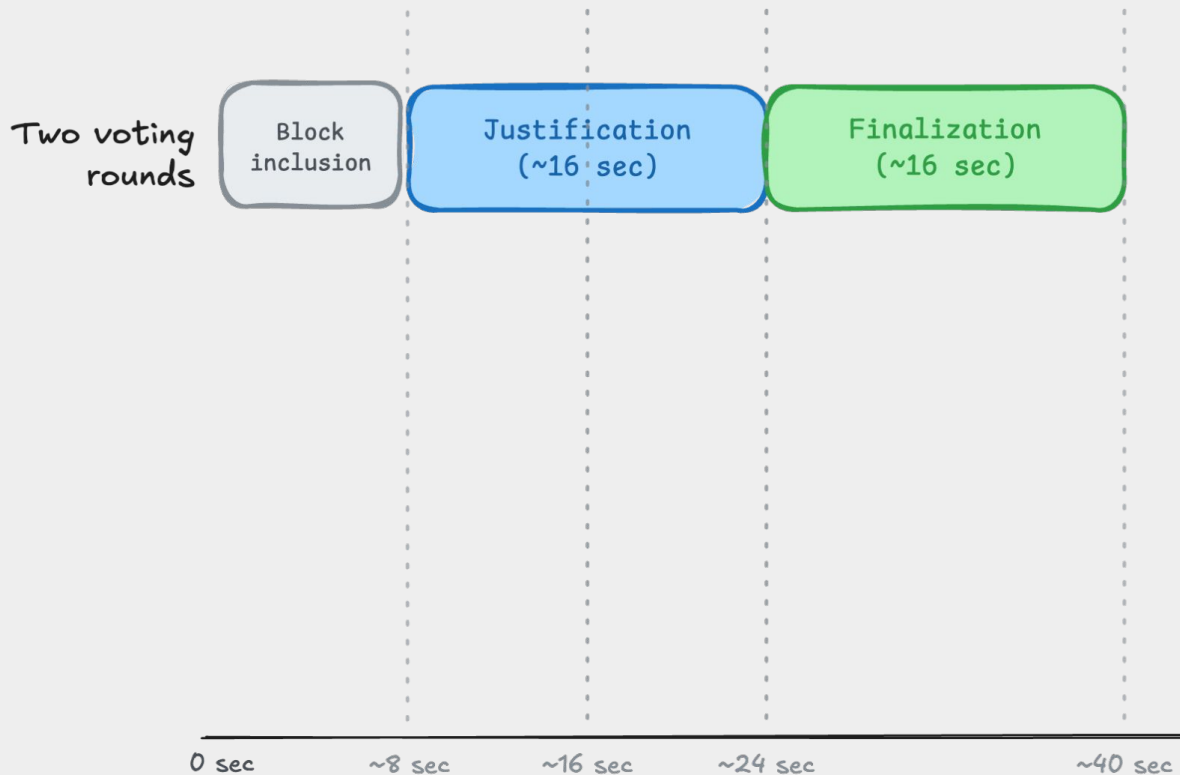
One-Round Casper FFG



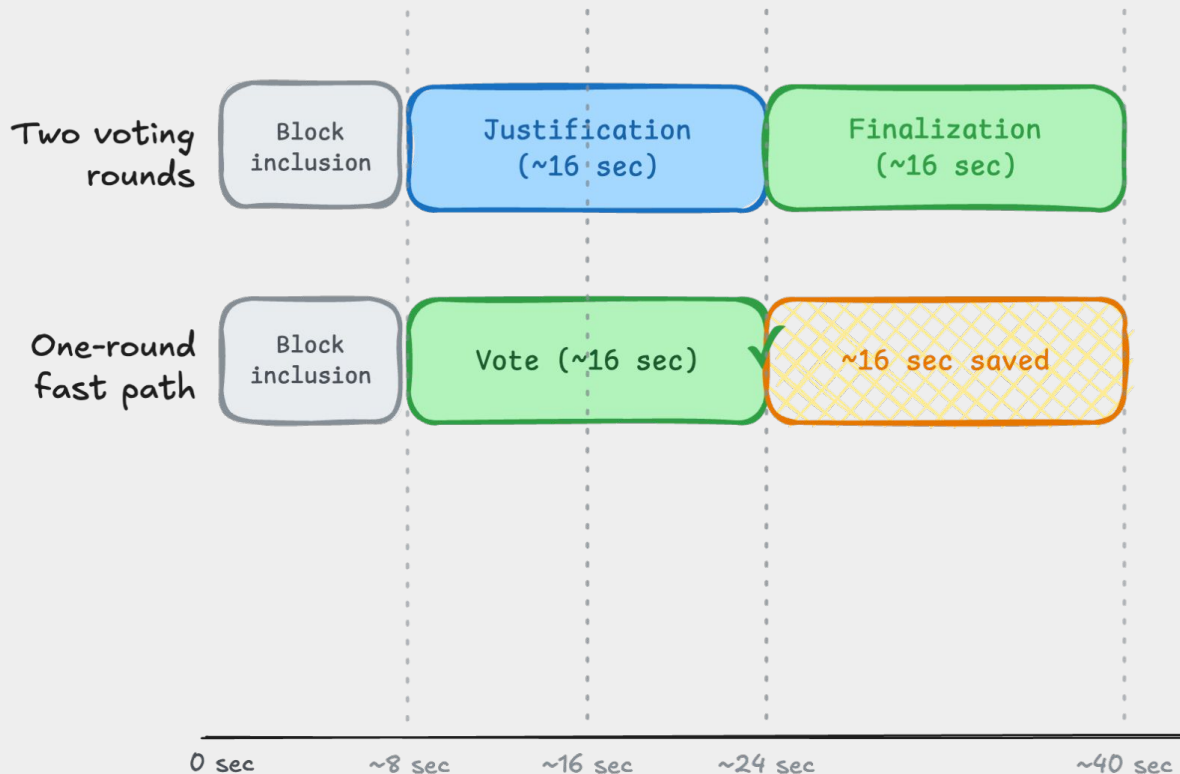
One-Round Casper FFG



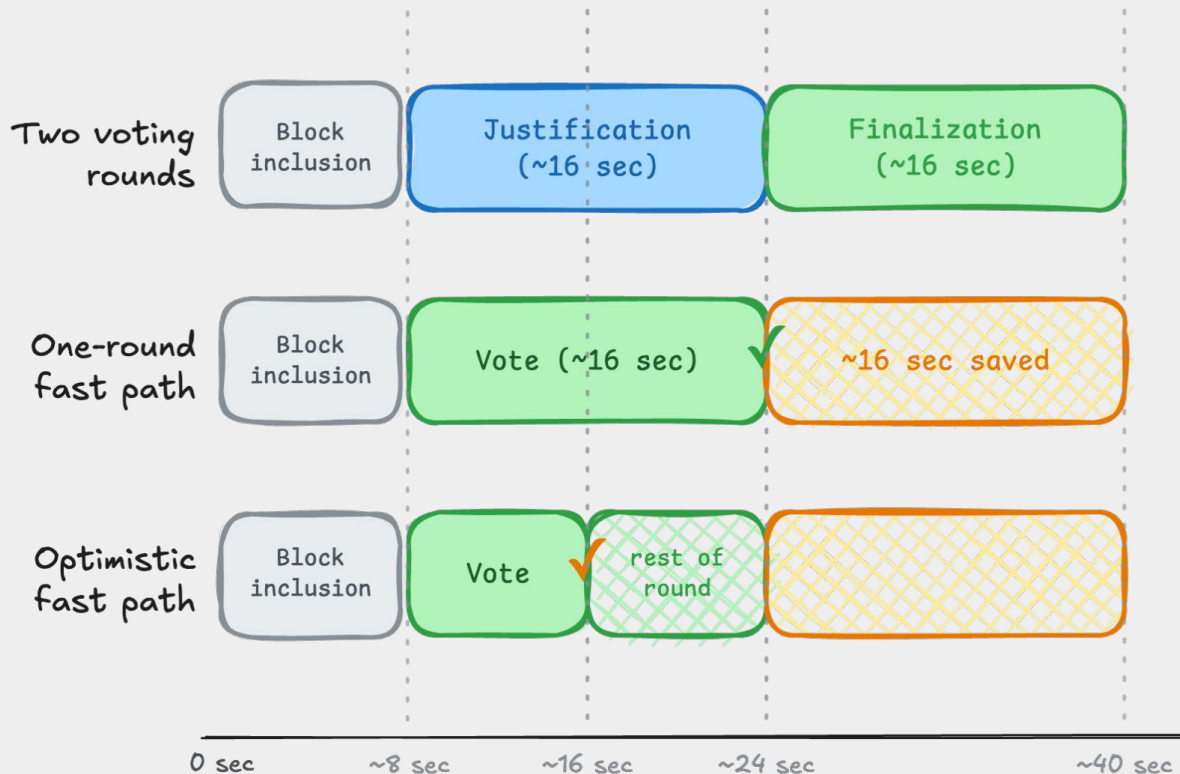
One-Round Decoupled Consensus



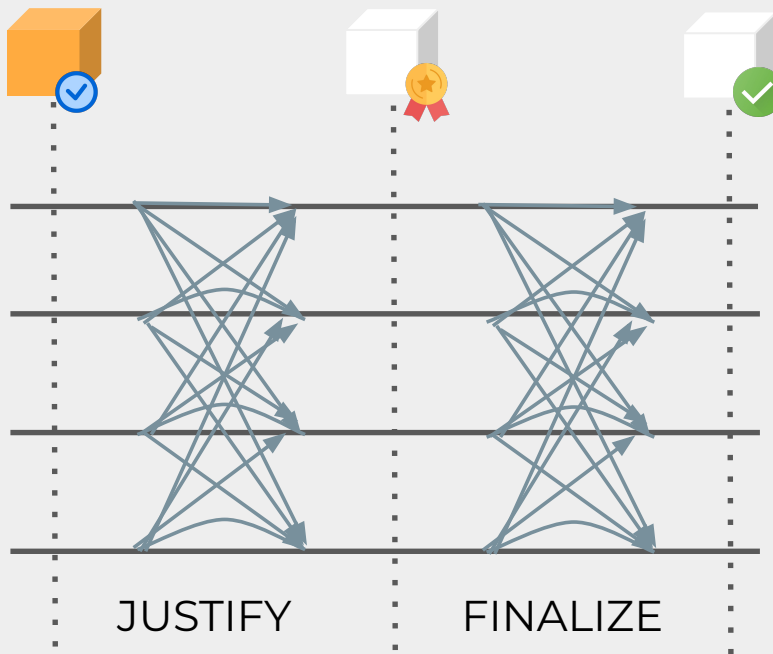
One-Round Decoupled Consensus



One-Round Decoupled Consensus



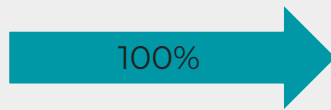
Optimistic



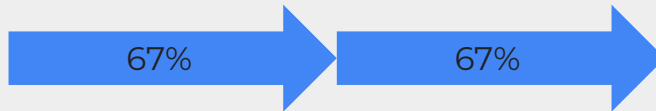
f = 33%



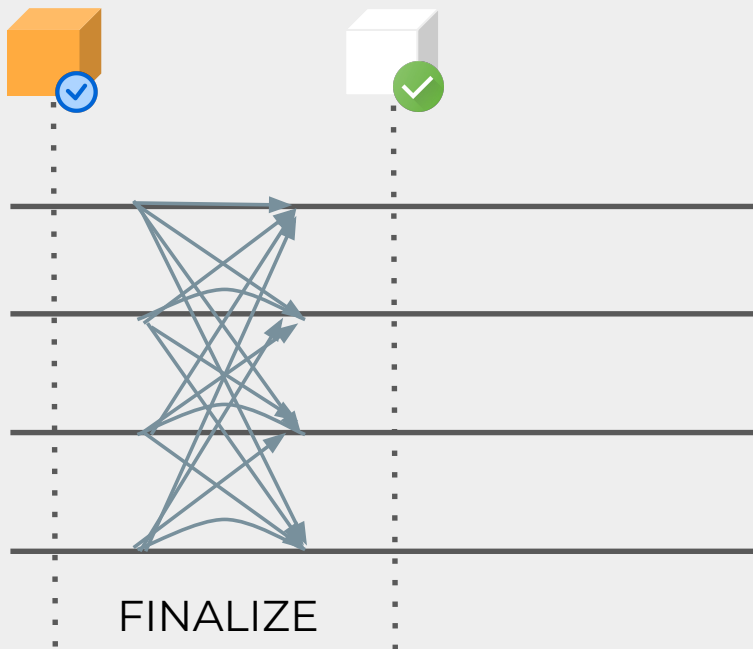
Fast Path:



Slow Path:



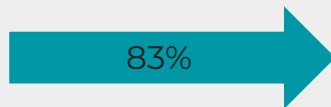
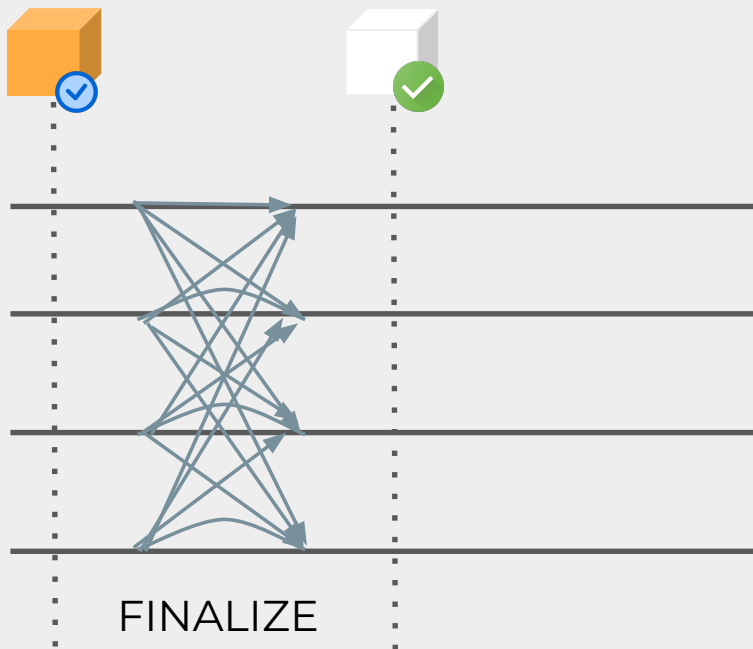
Pure



f = 20%



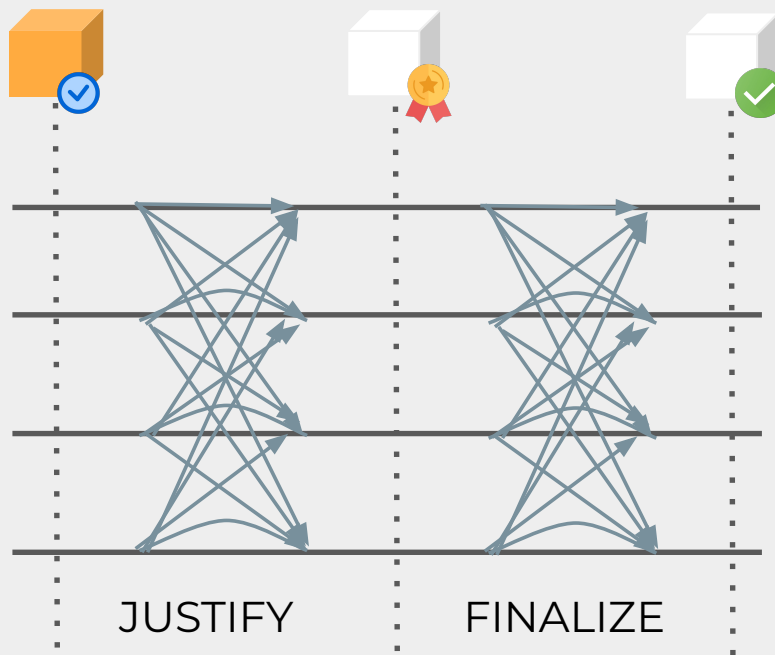
Pure



f = 25%



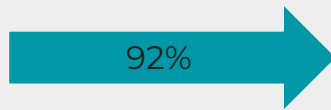
Hybrid



f = 33%



Fast Path:



Slow Path:



Why Ethereum Is Uniquely Positioned

Large Validator Set

→ removing a round is meaningful

Total active validators

932,205

100%

High Participation

→ we can benefit from optimistic conditions

Participation %

99.81%

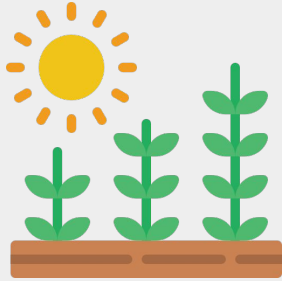
99.57%

Synergies with High Finality Threshold

→ we benefit from raising the finality threshold from 67% to e.g., 83%

83%

Synergies with High Finality Threshold



CROPS



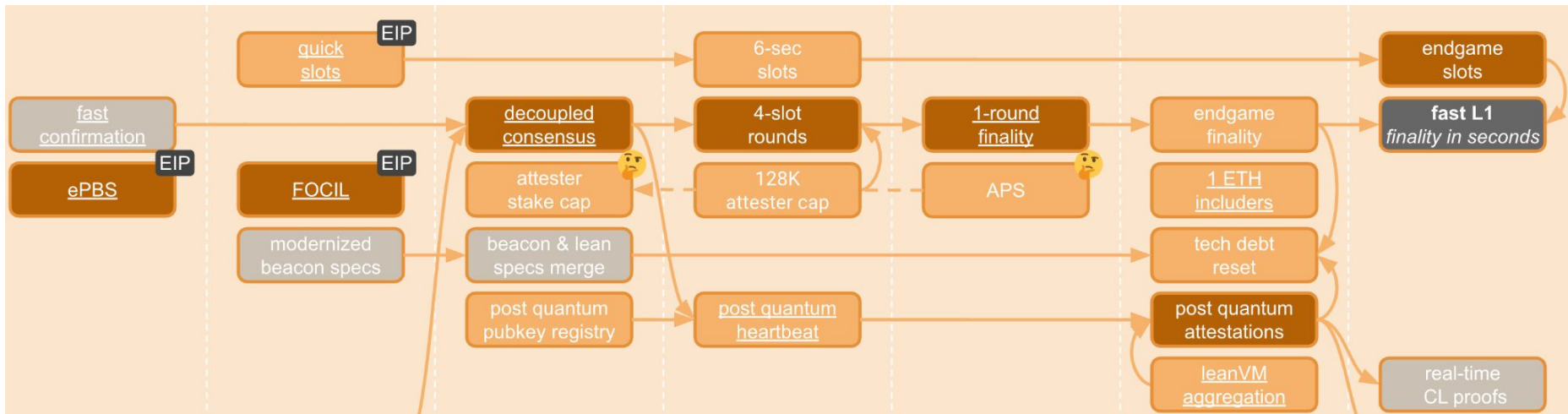
Hardness

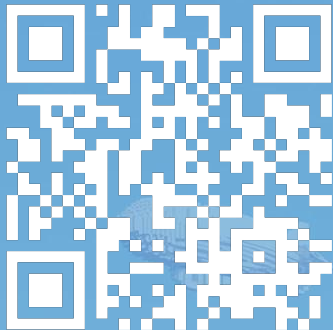


Stronger Finality

Consequences:

1. Longer periods of non-finality
2. Add economic safety on top?





Thank you!



@yannvon



yann.vonlanthen@ethereum.org