Lecture 16: Longest Chain Protocol Meets BFT

https://web3.princeton.edu/principles-of-blockchains/

Professor Pramod Viswanath
Princeton University

This lecture:
Both finality and dynamic availability
Hybrid consensus featuring both longest chain and BFT methods
Design of Ethereum 2.0

The Story So Far

Blockchain Protocols

Safety: all parties have the same ledger

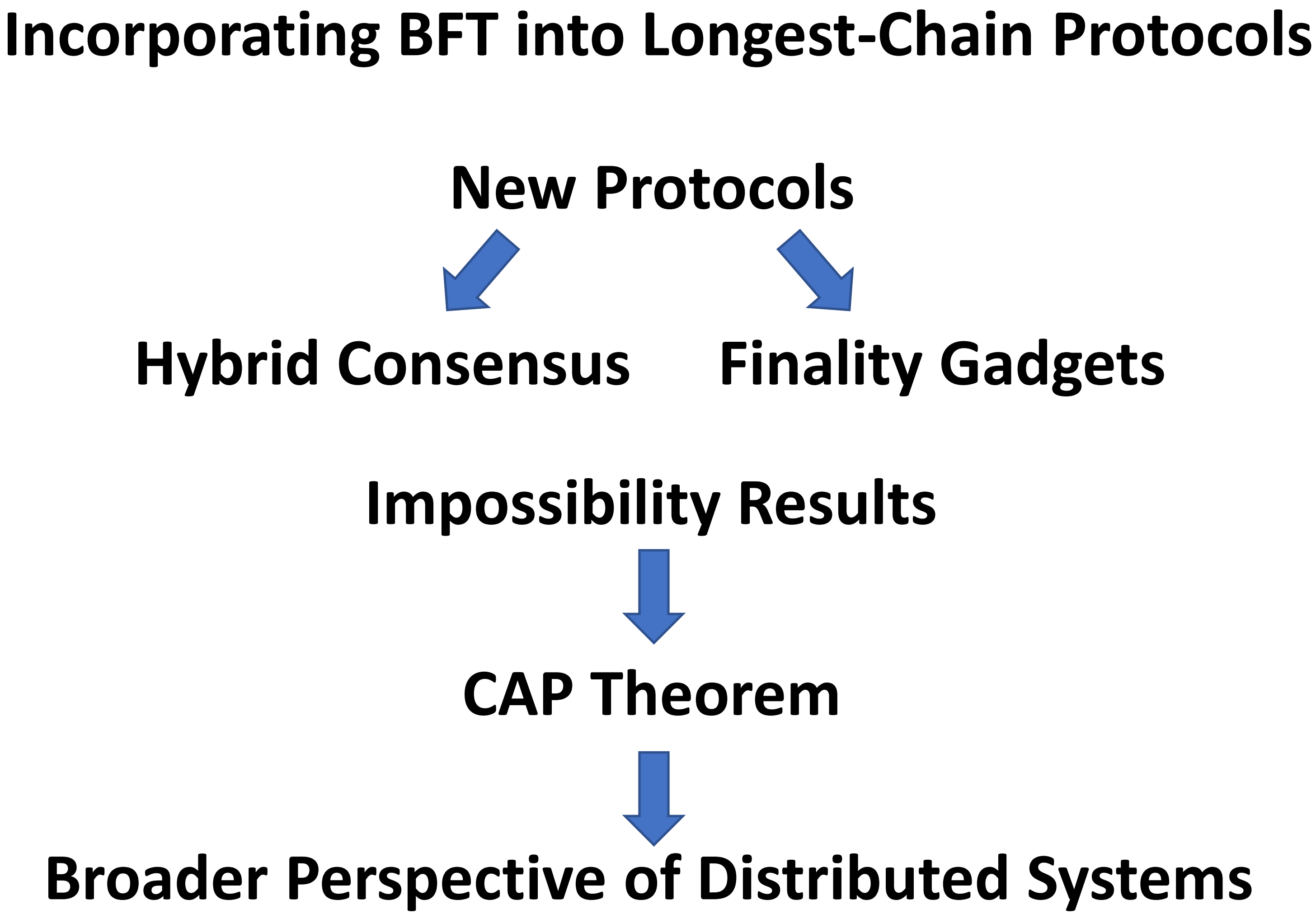
Liveness: the ledger keeps growing

Two families

Longest-chain (Bitcoin):	✓ permissionless	✗ unsafe in asynchrony
BFT-style (HotStuff):	✗ permissioned	✓ safe under asynchrony

TRADEOFFS!

Today’s Lecture



Incorporating BFT into Longest-Chain Protocols

New Protocols

Hybrid Consensus

Finality Gadgets

Impossibility Results

CAP Theorem

Broader Perspective of Distributed Systems

Hybrid Consensus

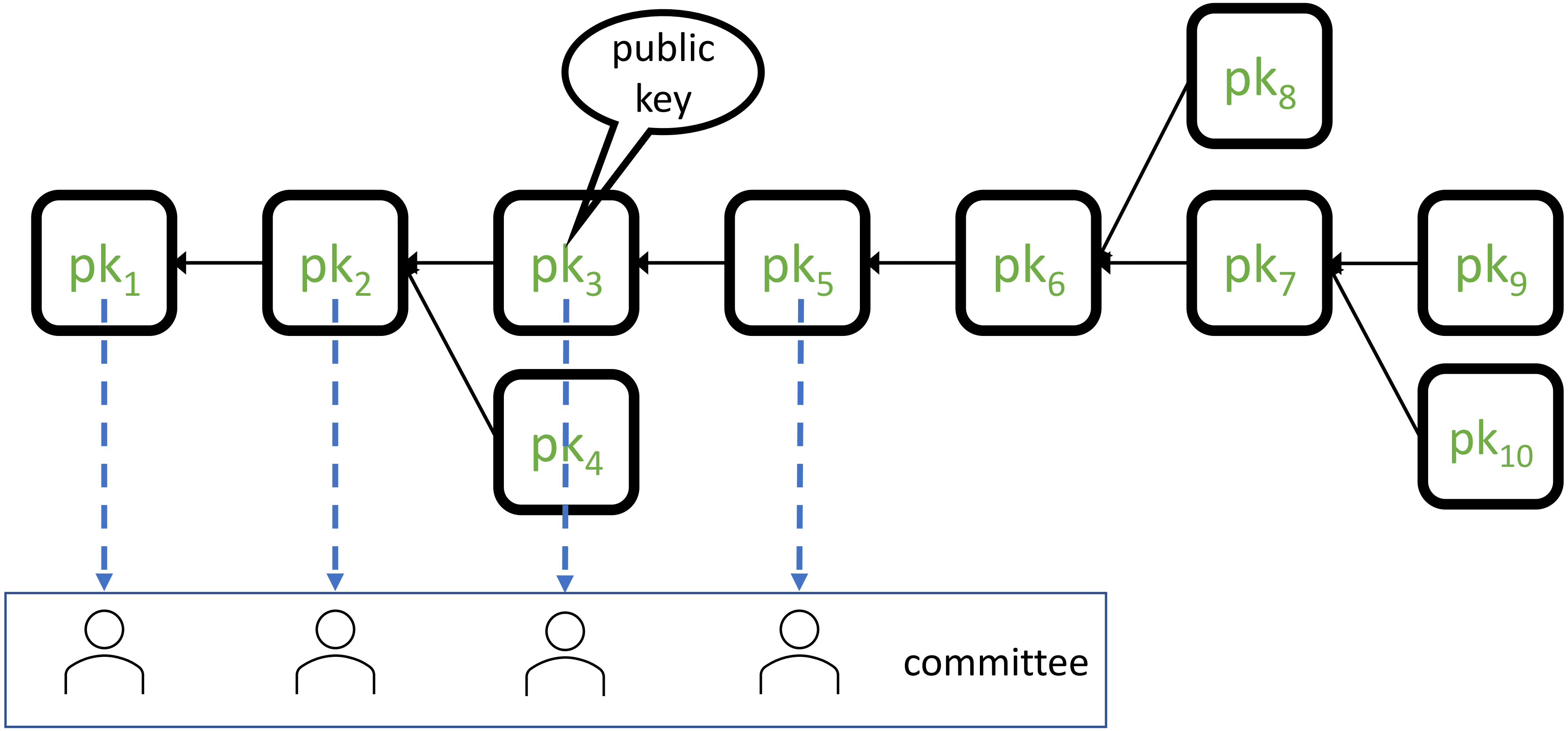
Longest-chain protocol is slow to confirm txns

Can we have fast confirmation in a PoW permissionless system?

Idea: Bring HotStuff to PoW for fast confirmation!

Need decentralized, fair committee election

Hybrid Consensus



Longest-chain protocol can serve as committee election mechanism

A fool-proof, fair, decentralized method!

Hybrid Consensus

Some finer details

- Can’t stop mining!
  - Adversary can upend longest chain if honest miners stop
  - Committee overturned ➔ insecure protocol
- Chain quality matters!
  - 1/3 mining adversary ➔ ½ adversary in committee. Cannot tolerate!
  - Need Fruitchains instead of Nakamoto consensus for ideal chain quality
- Susceptible to adaptive corruption
  - Committee is all-powerful; block proposers are no longer unpredictable!
  - Committee rotation protects against slow adaptive corruption

Hybrid Consensus

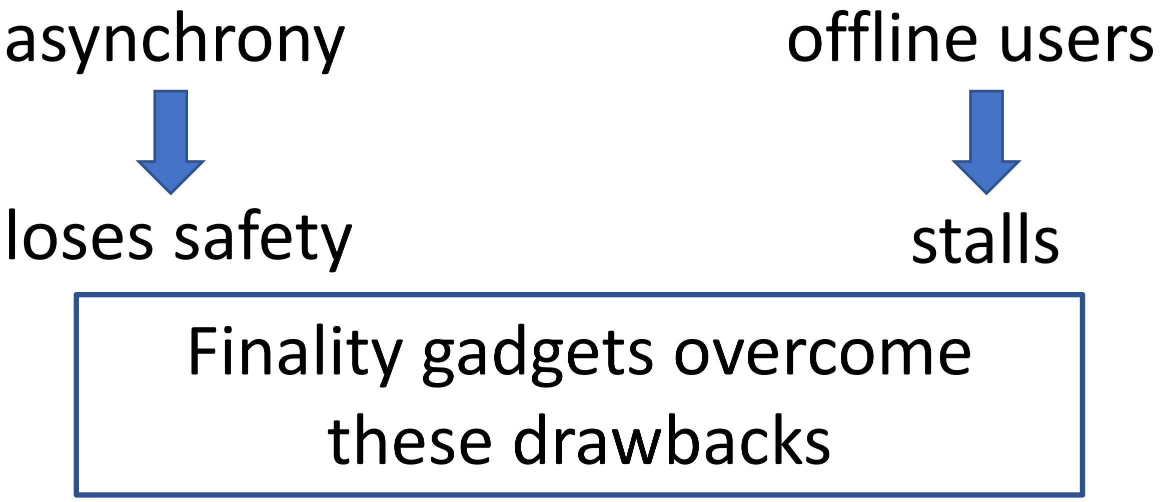
What it achieves

It achieves low confirmation latency in a PoW (permissionless) setting

Where it fails

$\beta > 1/3$

Needed for responsiveness



asynchrony

loses safety

offline users

stalls

Finality gadgets overcome these drawbacks

Finality and Availability

What we desire

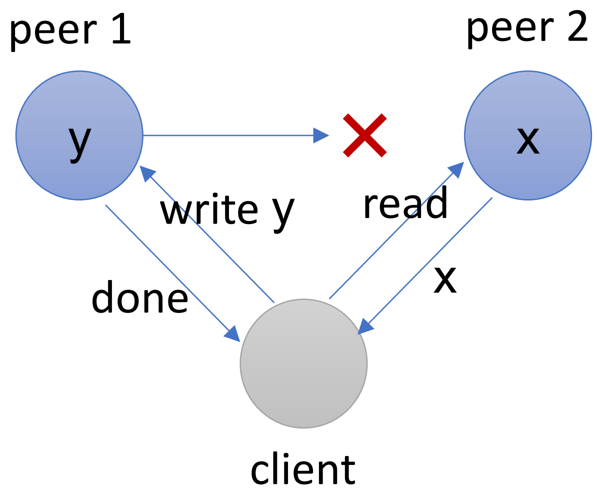
- Availability: a protocol that remains live and safe, despite variable participation
  - PoW longest chain has this property
- Finality: a protocol that remains safe, despite asynchrony
  - BFT protocol has this property

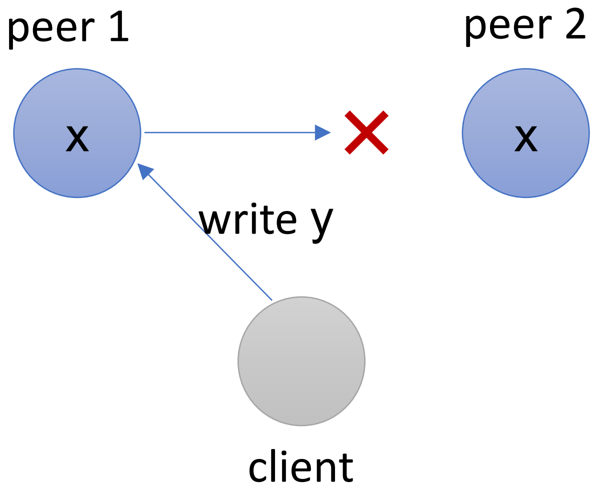
One protocol offering availability and finality?

Blockchain CAP Theorem says NO

The CAP Theorem in Distributed Systems

Theorem: A distributed system cannot be both **C**onsistent and **A**vailable during network **P**artitions (Brewer 2000, Gilbert & Lynch 2002)

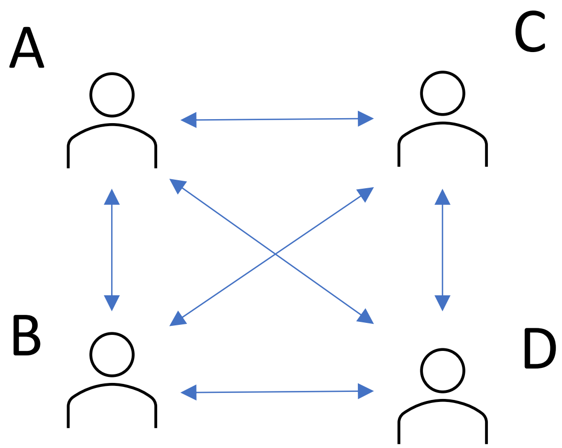


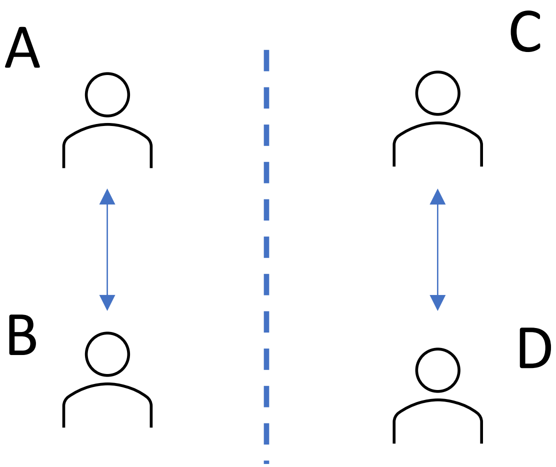


Choose liveness or safety during network partition!

The Blockchain CAP Theorem

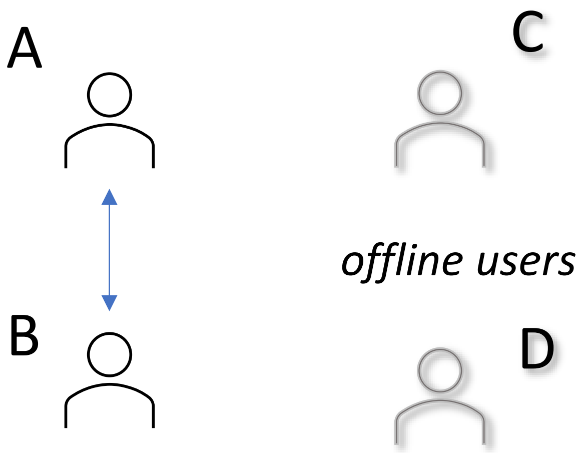
*No blockchain protocol can offer both a*vailability and *finality. [LR, 2020]*





network partition

protocol should stall

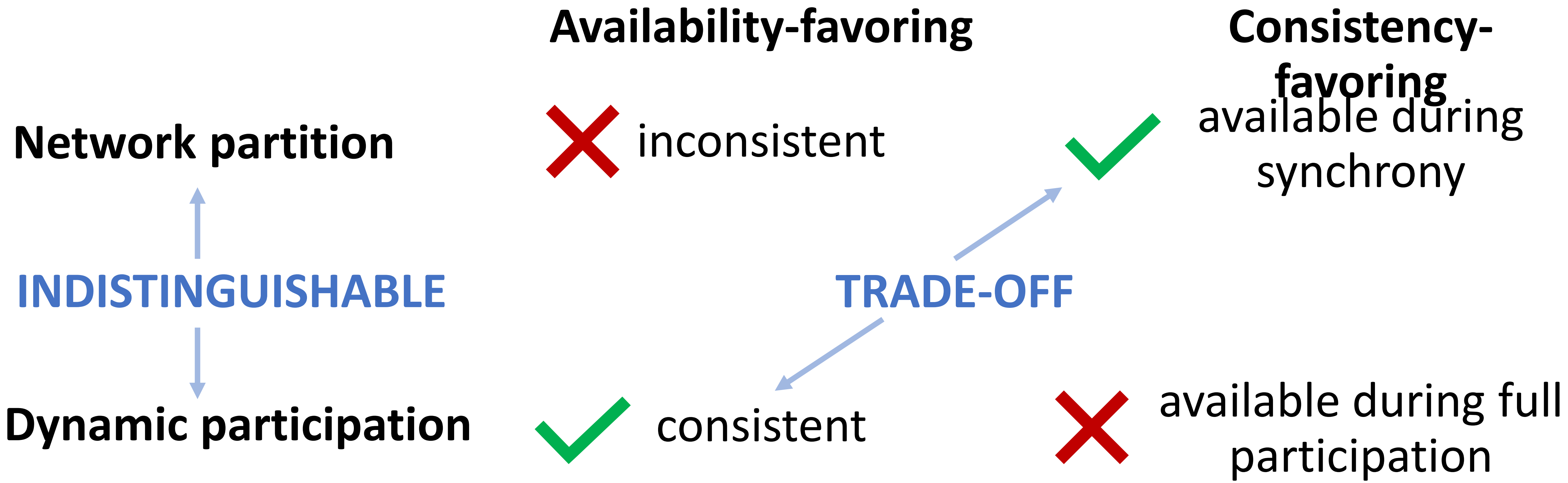


offline users

online users should continue

A decentralized protocol cannot distinguish between offline users and network partition

CAP Theorem in Blockchains



	Availability-favoring	**Consistency-favoring**
Network partition	✗ inconsistent	✓ available during synchrony
Dynamic participation	✓ consistent	✗ available during full participation

Solution: Two Confirmation Rules

- **Availability-preserving rule**
  - Remains live and safe under variable participation
  - Requires synchrony for liveness and safety

- **Finality-preserving rule**
  - Remains safe under all conditions
  - Is live only under synchrony and fixed participation

Each rule generates its own ledger!

Finality Gadget

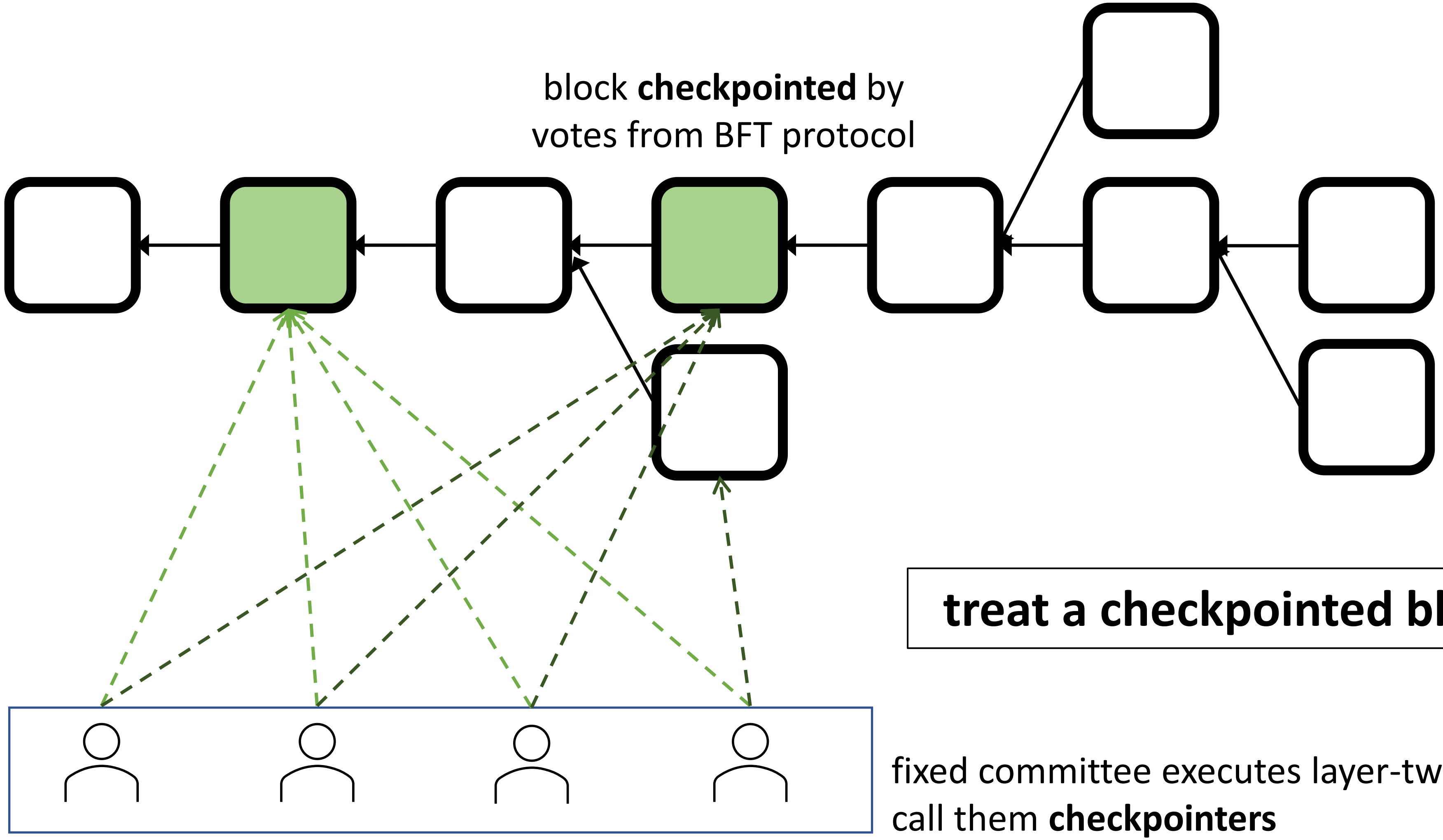
- Two-layer design

  Layer-one: Proof-of-Work Longest Chain

  Layer-two: Committee-based BFT protocol

- Longest chain protocol produces and confirms blocks
  - Works with variable participation
  - *k*-deep rule remains viable
- BFT protocol independently confirms blocks
  - Confirms the same set of blocks as produced by PoW!
  - Switches on or off based on participation level

Finality Gadget – Checkpoints



blocks produced by PoW mining

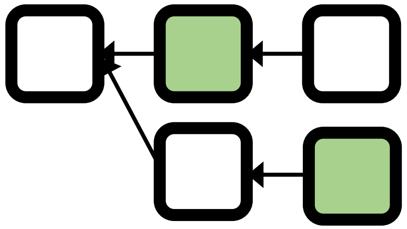
block **checkpointed** by votes from BFT protocol

treat a checkpointed block as final

fixed committee executes layer-two BFT protocol
call them **checkpointers**

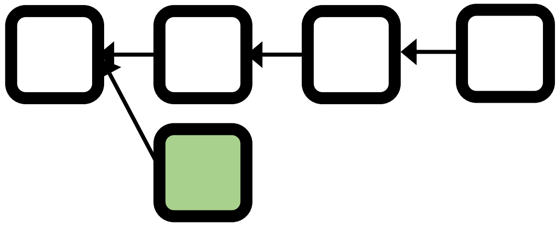
Rules of Checkpointing

- Checkpoint blocks on the same chain



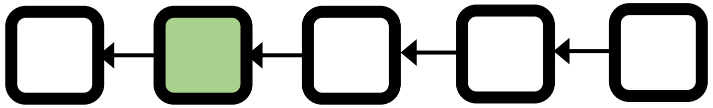
If not, safety violation!

- Checkpoint blocks on the longest chain



If not, liveness violation!

- Checkpoint blocks close to the tip



If not, checkpointing not of much use

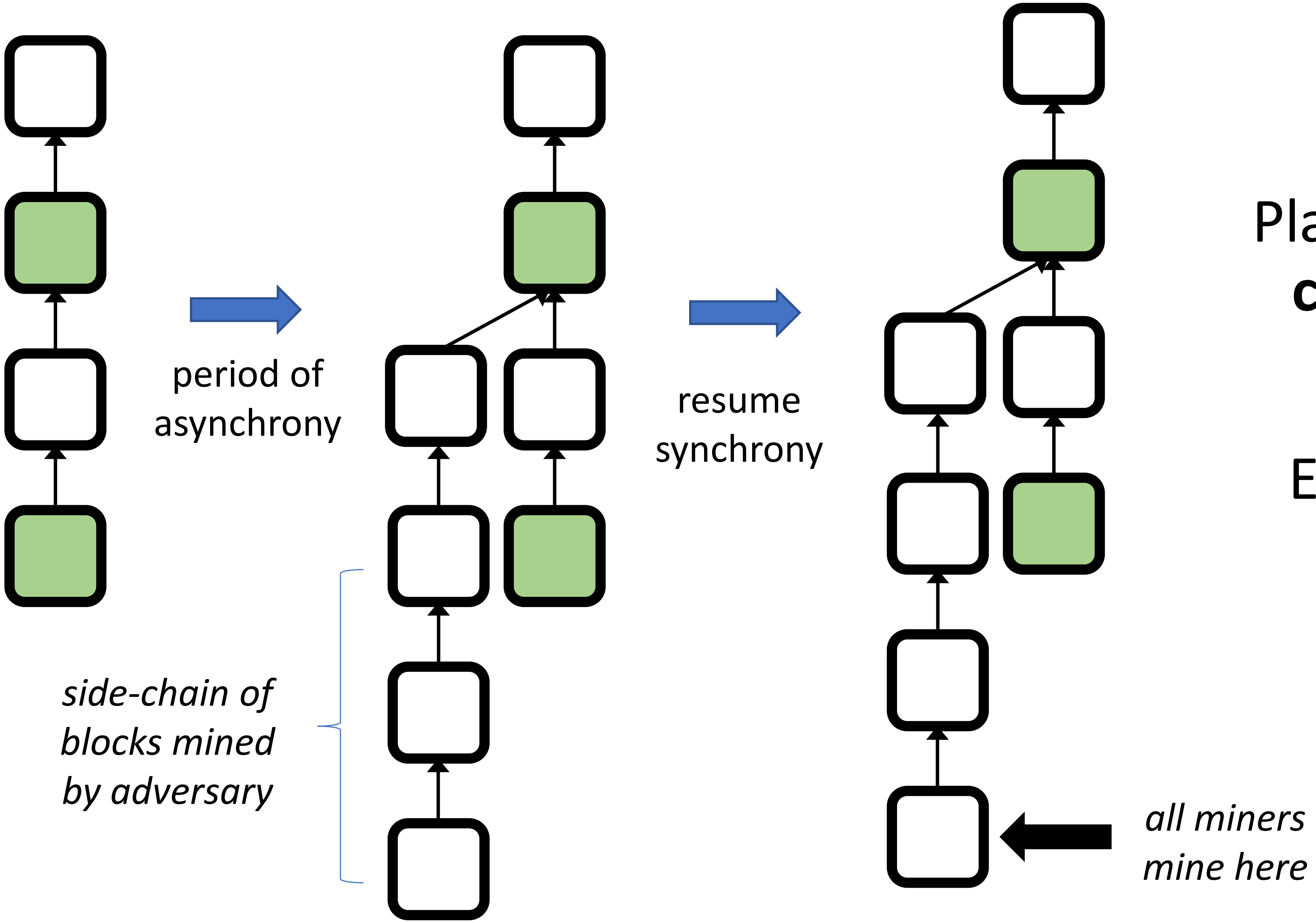
More about Checkpointing

Checkpointing protocol is a *consensus engine*

- Input values
  - **In theory:** entire chain leading up to prospective checkpoint block
  - **In practice:** hash of prospective checkpoint block

- Validity conditions
  - **Classical:** if all honest users have same input, that input is finalized
  - **For gadgets:** if all honest users have chains with a *k*-common prefix, then finalized block is on common prefix

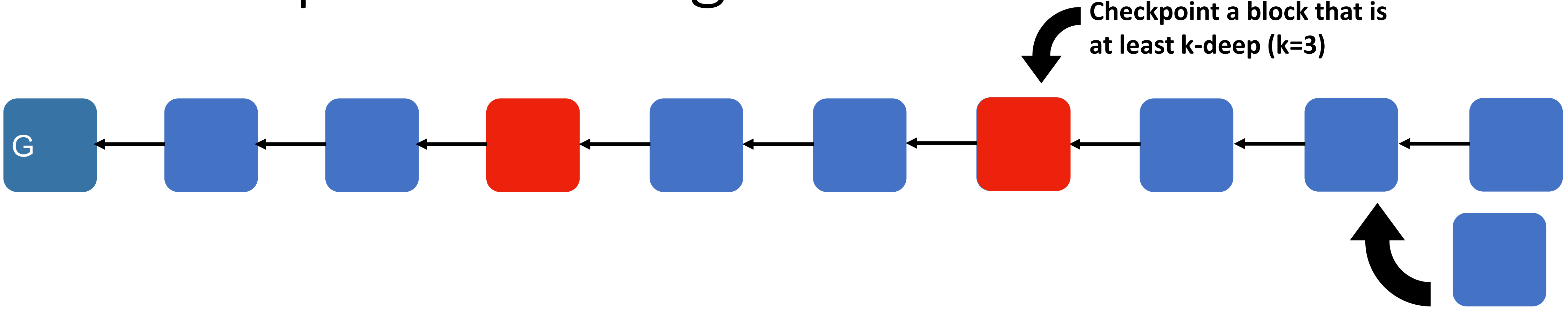
Second Look at the Two-layer design



Players must follow **longest checkpointed chain** rule

Extend the longest chain below the latest checkpoint block

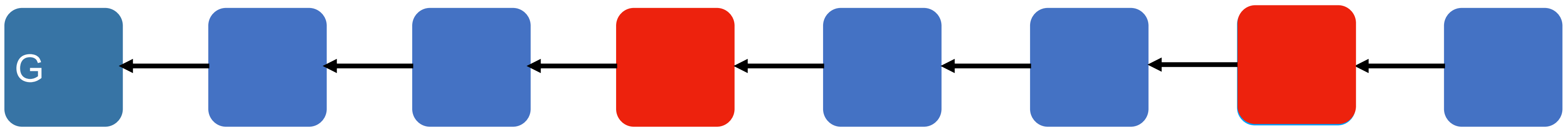
Checkpointed Longest Chain



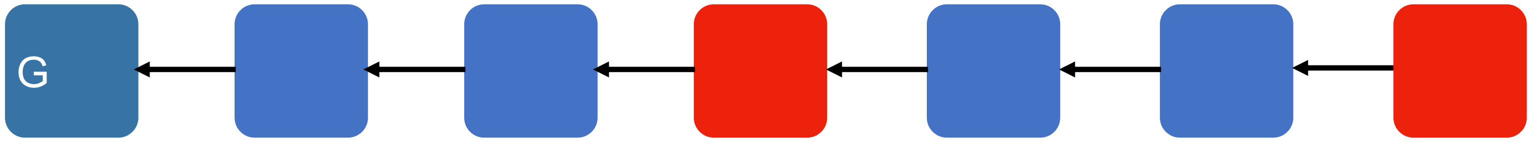
Checkpoint a block that is
at least k-deep (k=3)

Mine a new block extending
the longest chain that contains the latest checkpoint

Availability-preserving rule (k-deep)



Finality-preserving rule (checkpoints)



Checkpointed Longest Chain

- **Availability-preserving rule** (*k*-deep rule)
  - Remains live and safe under variable participation
  - Requires synchrony for liveness and safety

- **Finality-preserving rule** (checkpoint-based rule)
  - Remains safe under all conditions
  - Is live only under synchrony and fixed participation

Each rule generates its own ledger!

Finality Gadget

What it achieves

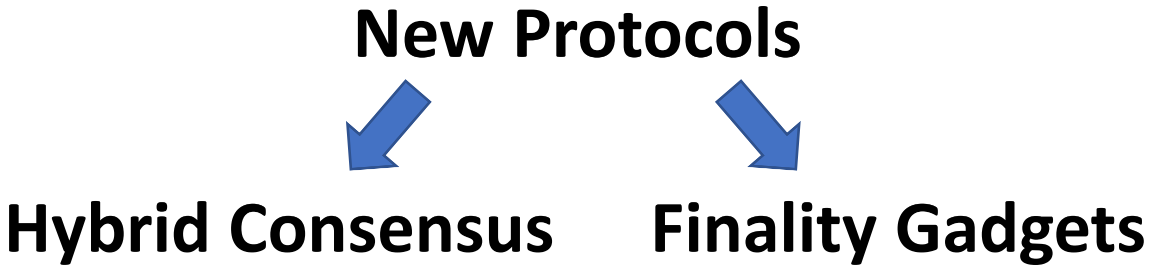
- Safety under asynchrony

- Safety and liveness under variable participation? Requires confirming *k-deep blocks*. [***Checkpointed Longest Chain, Ebb-and-Flow]***

- Faster confirmation? Requires confirming blocks *at tip*. [***GRANDPA]***

Summary

Incorporating BFT into Longest-Chain Protocols



New Protocols

Hybrid Consensus **Finality Gadgets**

Design of Ethereum 2.0

GHOST + Checkpointing

Going around the CAP Theorem

- Best-effort availability
  - files in a data center

Typically uses a consensus protocol in the back-end

- Best-effort consistency
  - Web content

No guarantee that the content retrieved is the latest

Attendance : NFT Drop

https://poap.website/main-skill-school

- Mint token to Metamask.
- Submit tx hash for attendance claim.

- Instructions in Ed pinned posts.