



# Block Production and Finalization in Polkadot:

## Understanding the BABE and GRANDPA Protocols

Bill Laboon

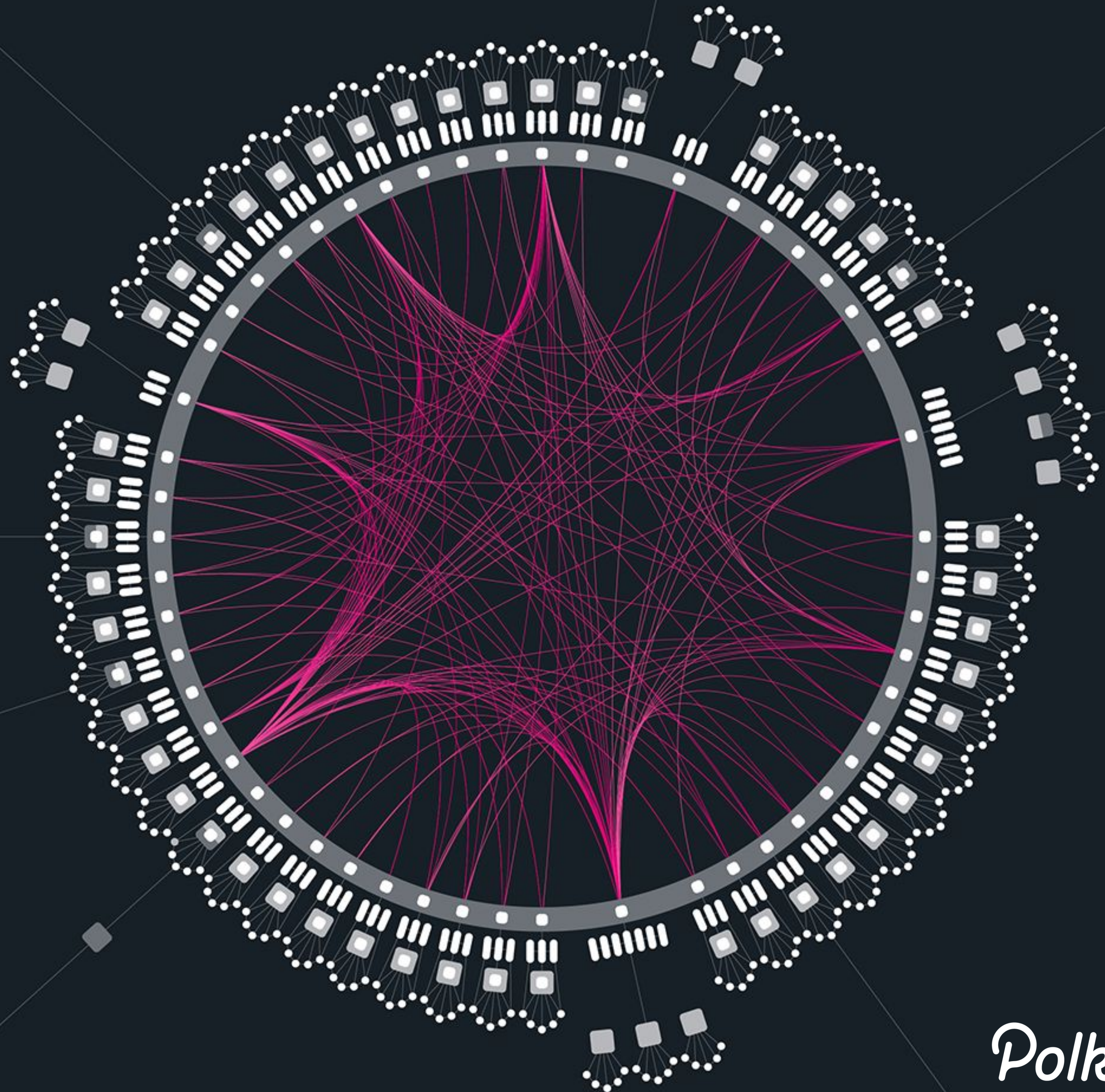
Technical Education Lead at Web3 Foundation



Web3 Foundation is a Switzerland-based foundation dedicated to "nurturing and stewarding cutting-edge technologies and applications in the fields of cryptographically-powered decentralized software protocols."



# POLKADOT BASICS










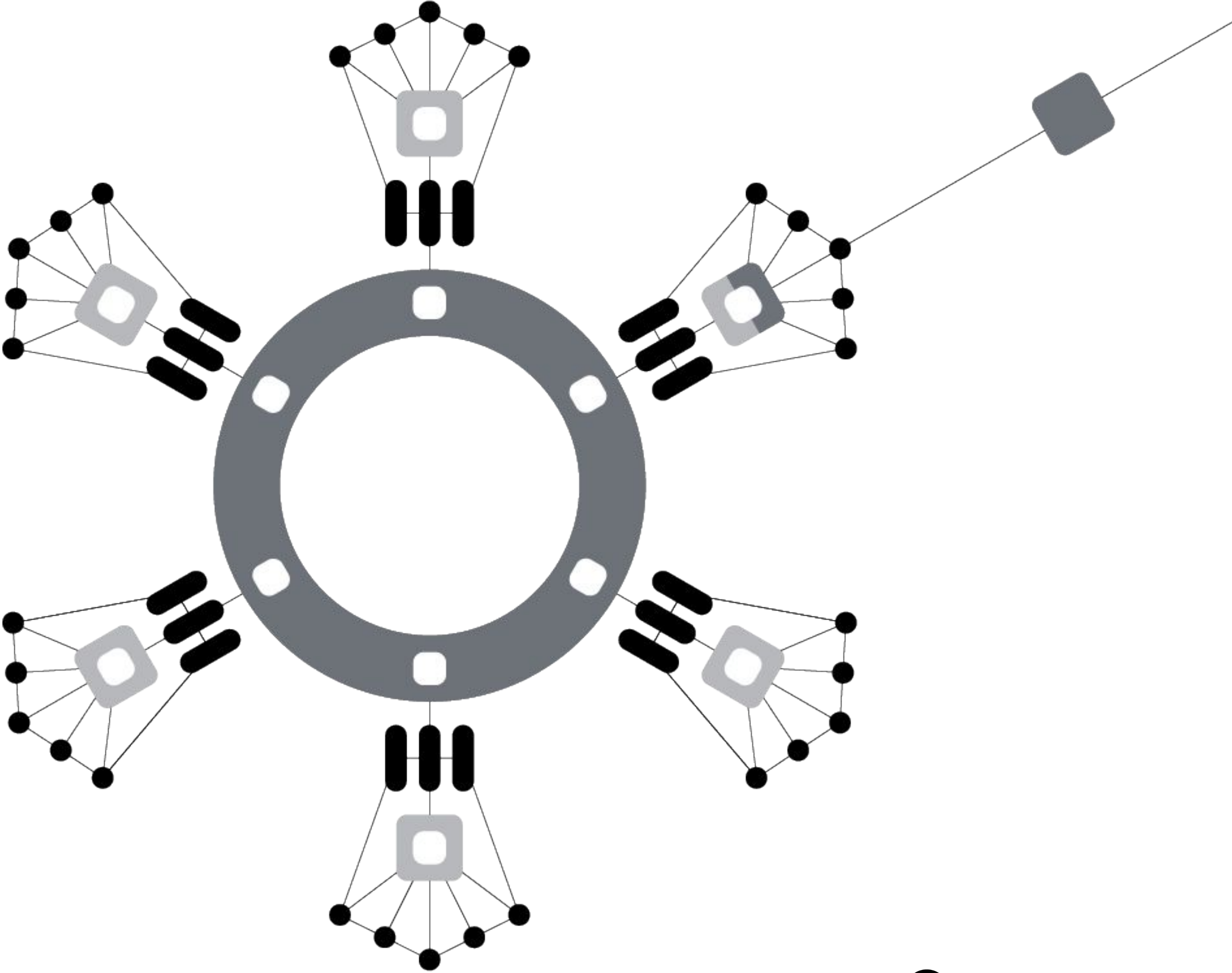
*Polkadot.*



# INTRODUCTION

## Polkadot

	Validators
	Collators
	Relaychain
	Parachains
	Parathreads
	Bridges
	Other Blockchains



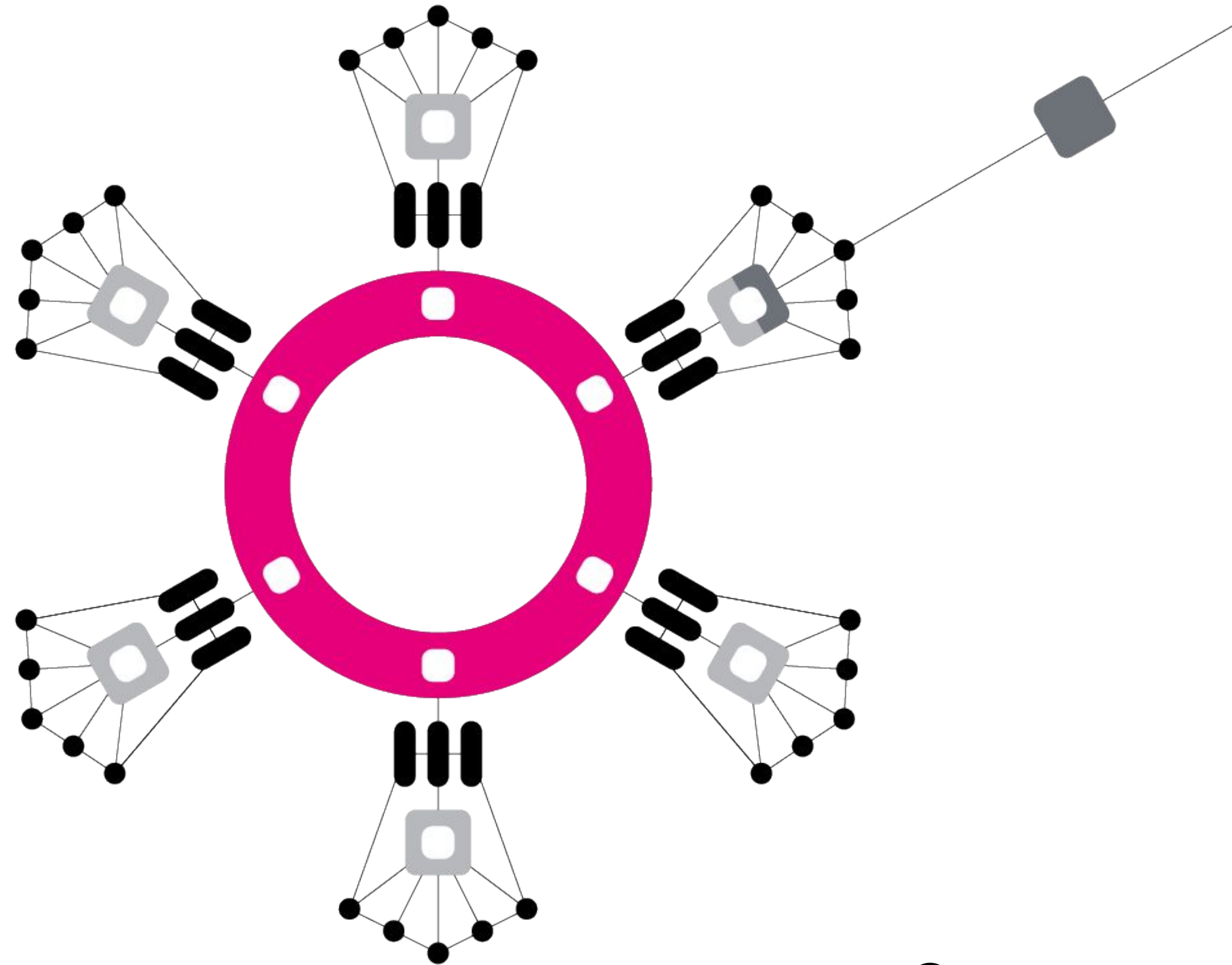
*Polkadot.*

# INTRODUCTION

## Relaychain

**The relay chain is the main chain of Polkadot.**

- *Other connected “parachains” are heterogenous shards - blockchains which share security and communicate with each other*
- *Relay chain holds the states of the parachains.*
- *It is secured via nominated proof of stake*



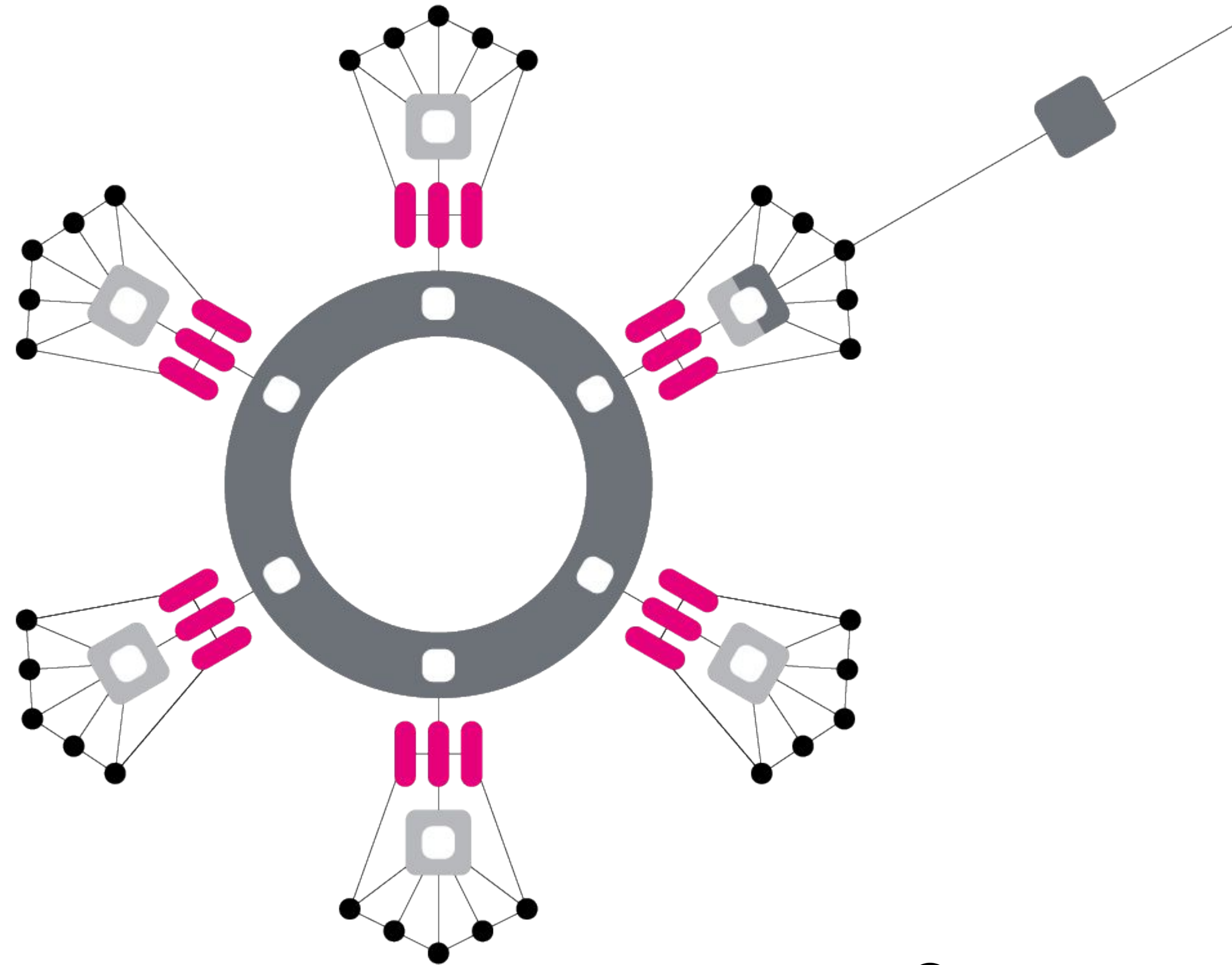
*Polkadot.*

# INTRODUCTION

## Validators

**Validators are the block producers of the relay chain.**

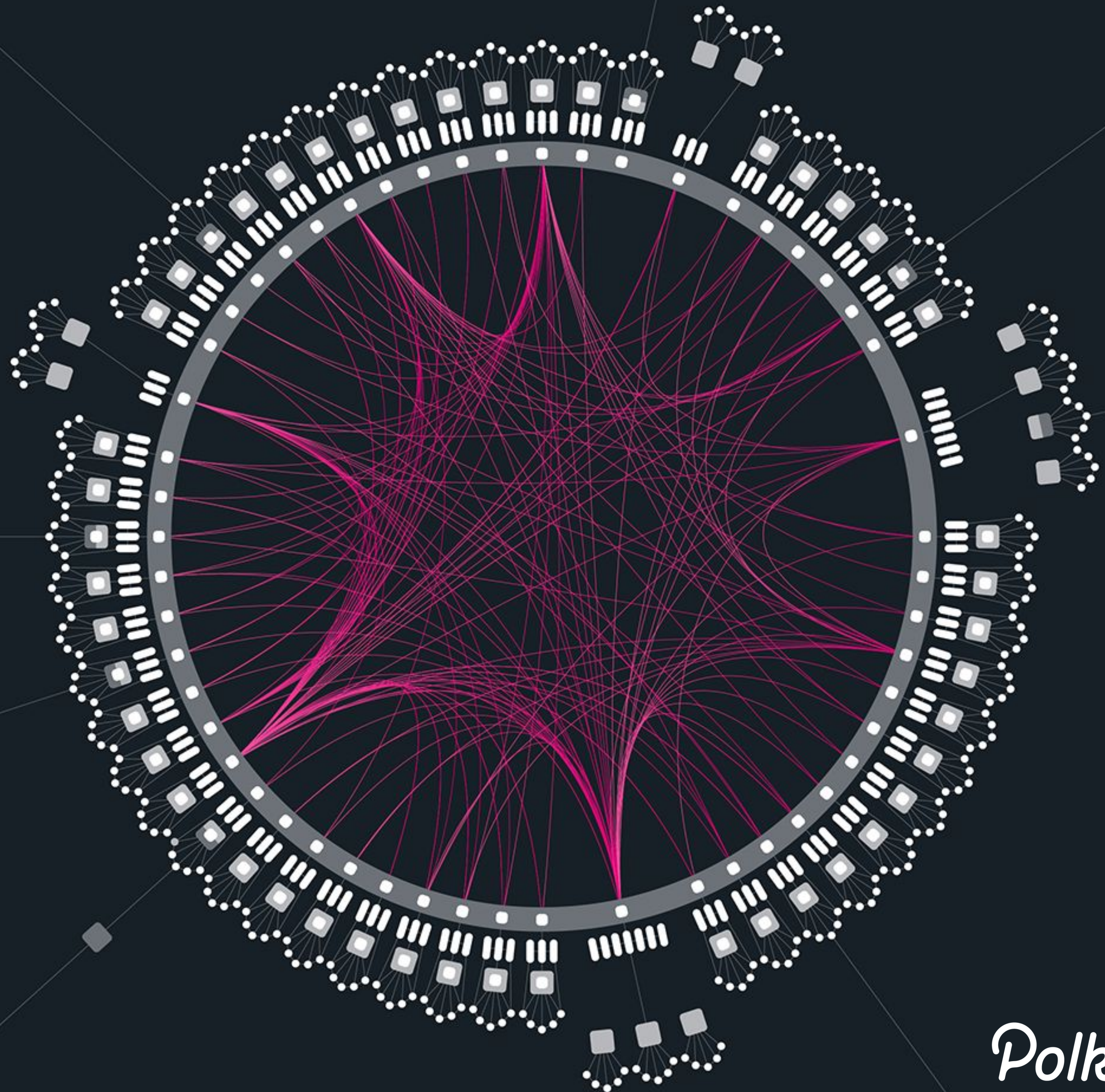
- *They produce and finalize blocks*
- *Other responsibilities such as communicating with parachains that we will not discuss in this talk*



*Polkadot.*



# HYBRID CONSENSUS



*Polkadot.*



# FINALITY

- Finality in classic blockchain model is probabilistic
- Other mechanisms (e.g. Tendermint) have provable finality
- BUT the big drawback is that they are vulnerable to stalling
- AND Polkadot may occasionally need to revert blocks that have been recently added (in case of conflicting information about parachains)
- ◆ This should be rare, so ideal situation is a situation where we generally have fast finalization, but block production can continue and we let finalization “catch up” when ready



# Why Do We Want Provable Finality?

- Allows us to prove to parties not involved in consensus that a block is final
- Provable finality makes bridges to other blockchains easier



# Hybrid Consensus

- Best of both worlds - block production can always continue as long as one validator is online
- Finalization is done via a separate finalization gadget
- In ideal circumstances, block finalization can be rather fast (a few blocks, empirically ~ 20-30 seconds on our canary network Kusama)
- With minor issues, can delay finalization while further checks are done
- In the event of severe network partitioning or malicious attack, block production can continue but we temporarily fall back to probabilistic finalization



# Impacts

- Polkadot messaging system speed is constrained by block production time, not time to finalize - allows faster message passing
- Allows liveness as long as one validator is producing blocks, albeit with probabilistic finality



# BABE vs GRANDPA

→ BABE (Blind Assignment for Blockchain Extension)

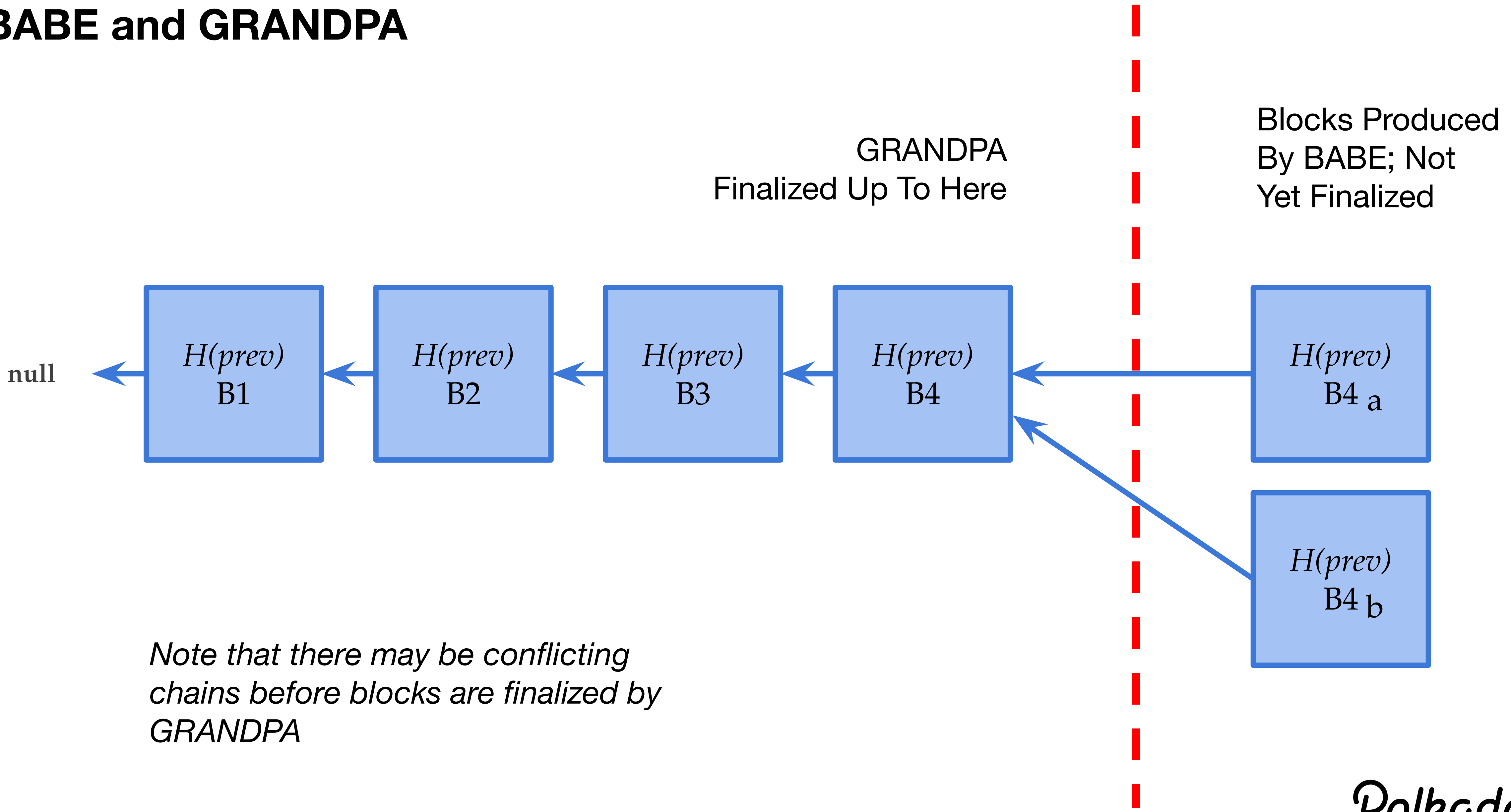
- ◆ Validators randomly select themselves to produce blocks
- ◆ Validators selected by amount of stake (tokens)
- ◆ For "young" blocks that are newly created

→ GRANDPA (GHOST-based Recursive ANcestor Deriving Prefix Agreement)

- ◆ Finalizes blocks separately
- ◆ For "old" blocks.. blocks must be produced before they are finalized



# BABE and GRANDPA

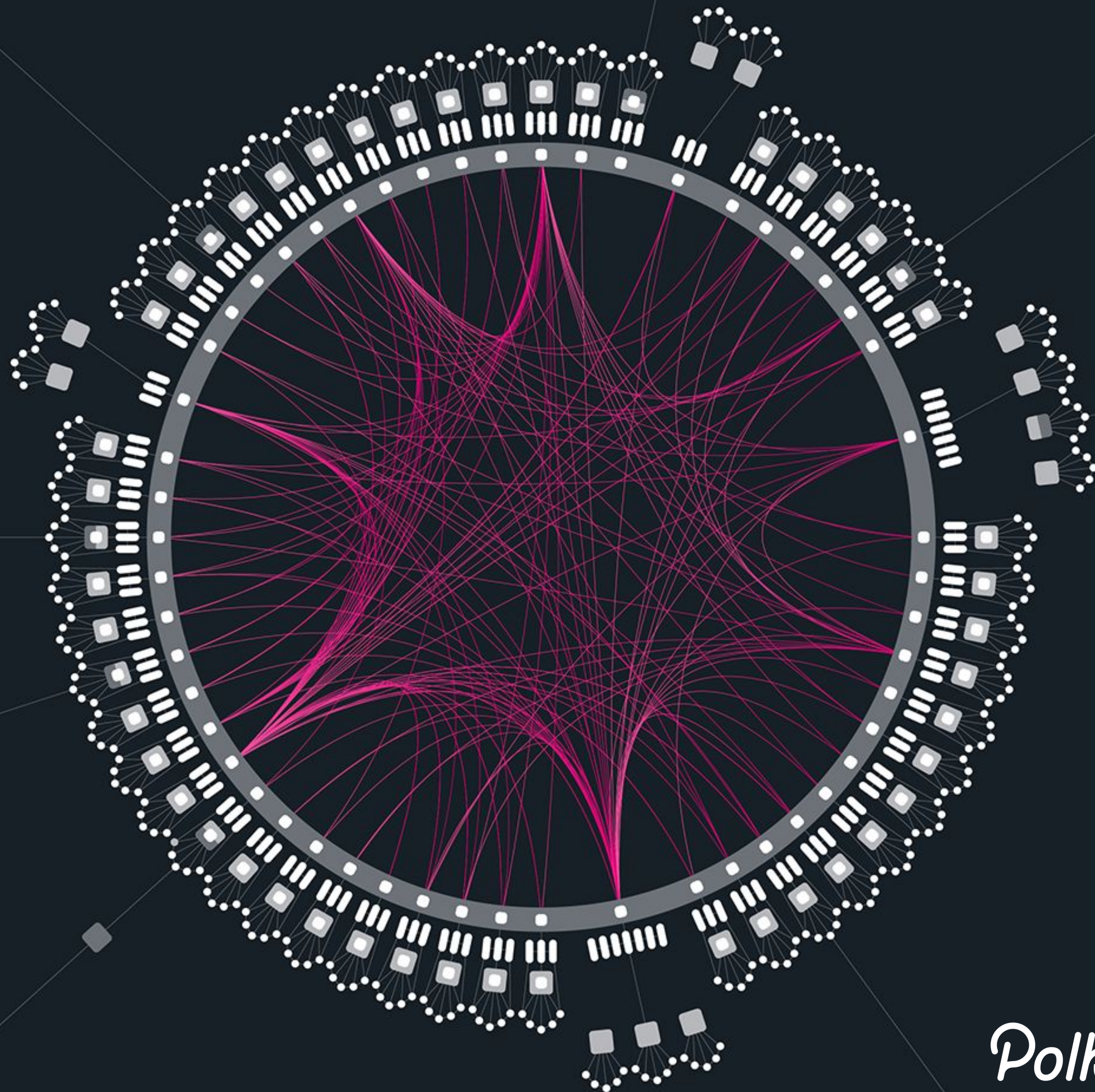


*Note that there may be conflicting chains before blocks are finalized by GRANDPA*



# BABE

Polkadot's Block  
Production Protocol



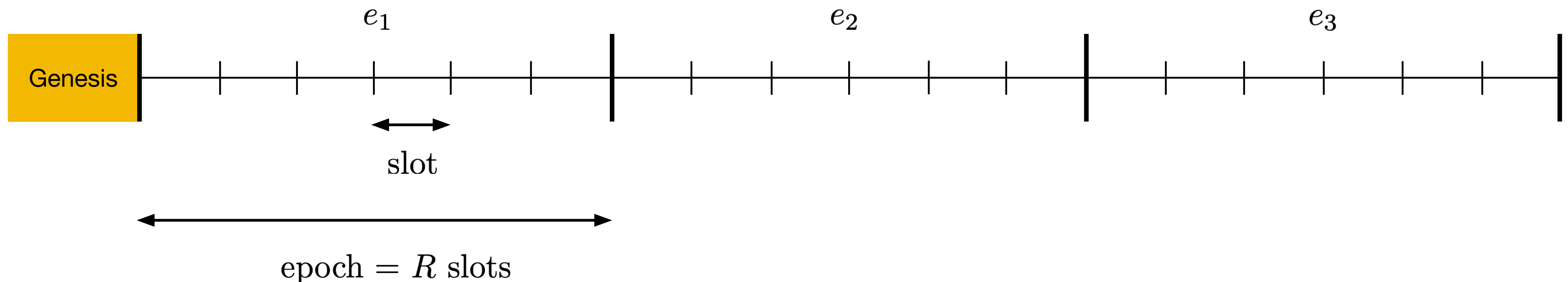
*Polkadot.*



# BABE (Blind Assignment for Blockchain Extension)

## Overview

- BABE [1] is a proof of stake (PoS) protocol
- BABE's design is similar to Ouroboros Praos [2]
- It does not depend on any central clock.

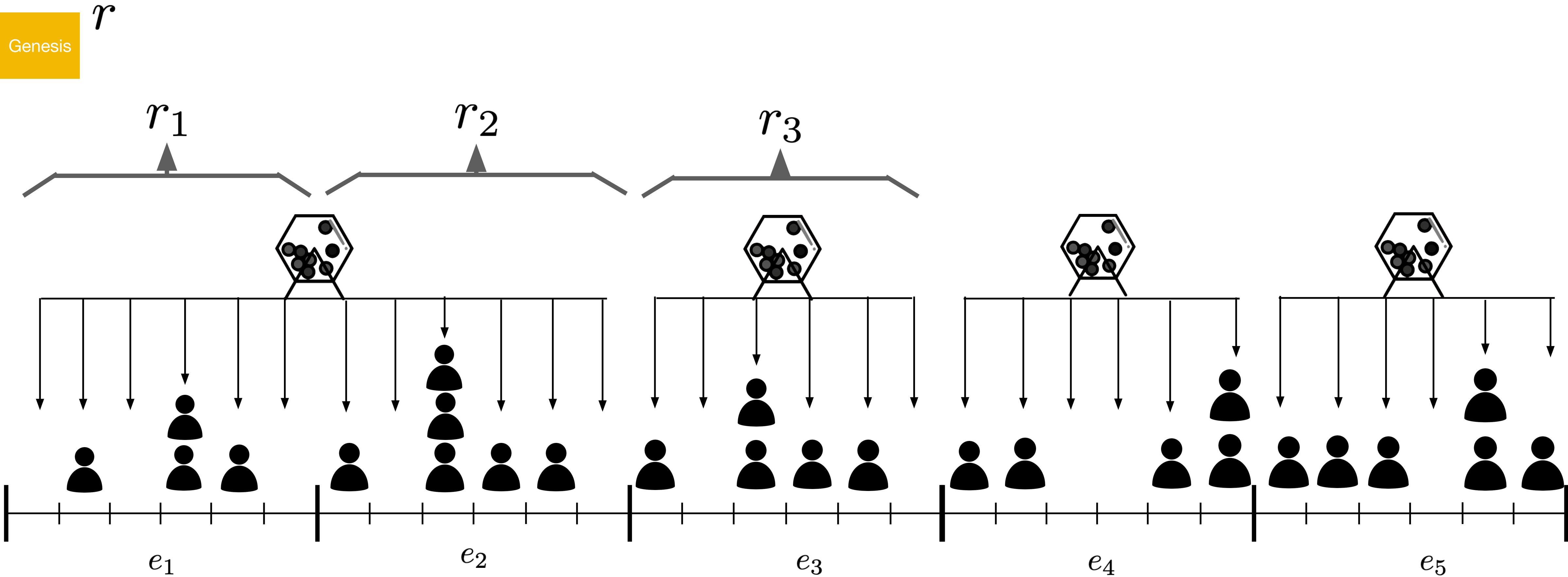


[1] Alper, Handan Kilinc. <https://research.web3.foundation/en/latest/polkadot/BABE/Babe/>

[2] David, Bernardo, et al. "Ouroboros Praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain." Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Cham, 2018.

# BABE (Blind Assignment for Blockchain Extension)

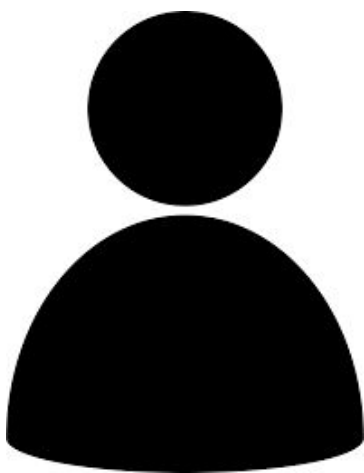
## Overview





# KEYS IN BABE

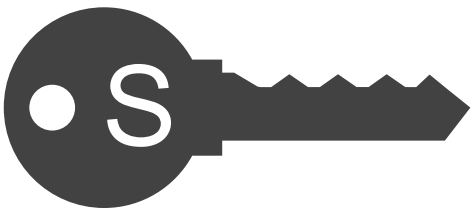
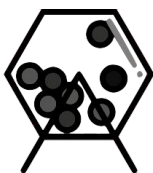
Validator



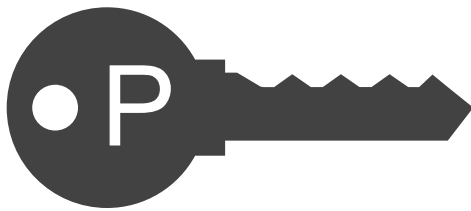
Block Signing  
Key



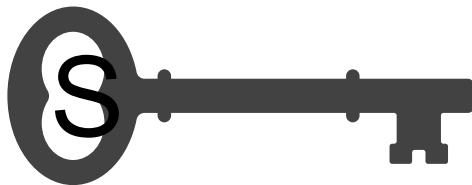
Lottery  
Key



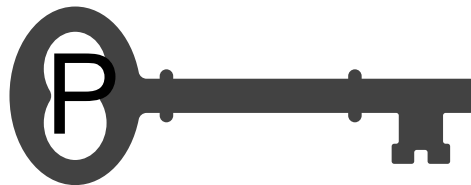
secret  
key



public  
key



secret  
key

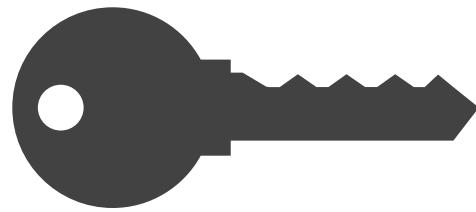


public  
key

VERIFIABLE RANDOM FUNCTION  
VRF

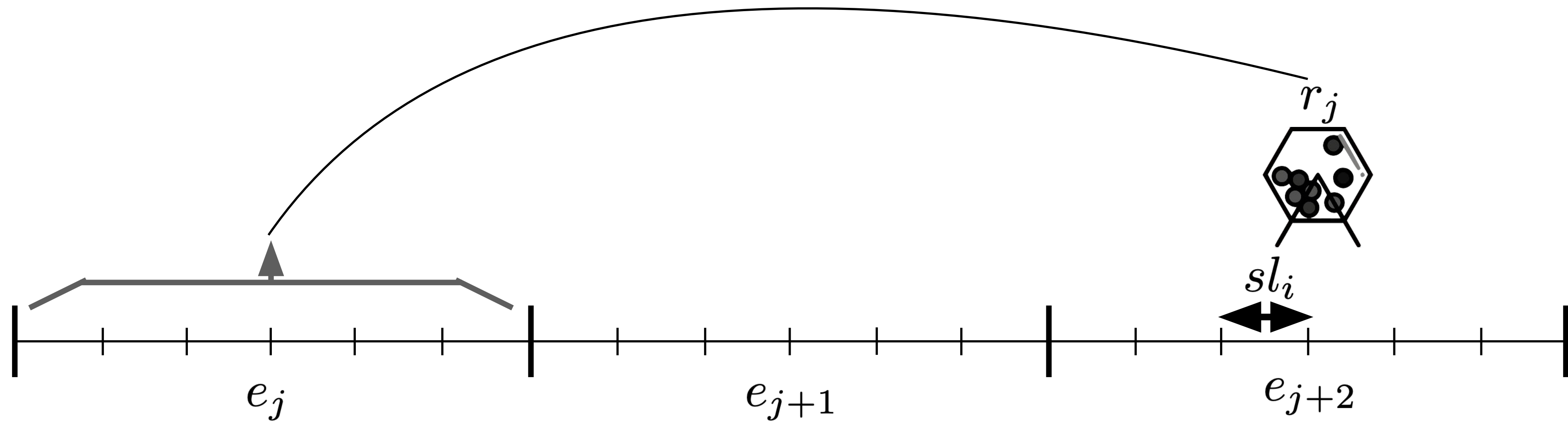
$F(\text{key}, x) \rightarrow 3819...2773$

$\text{Prove}(\text{secret key}, x, 3819...2773) \rightarrow \text{PROOF}$

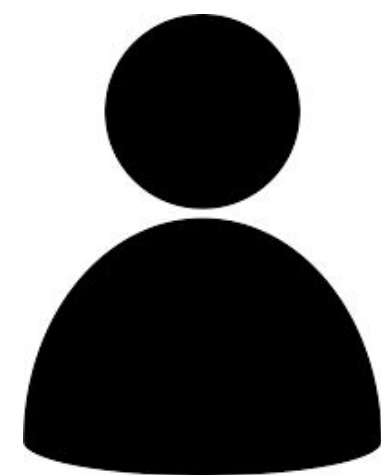




**LOTTERY**  
Slot Leader Selection



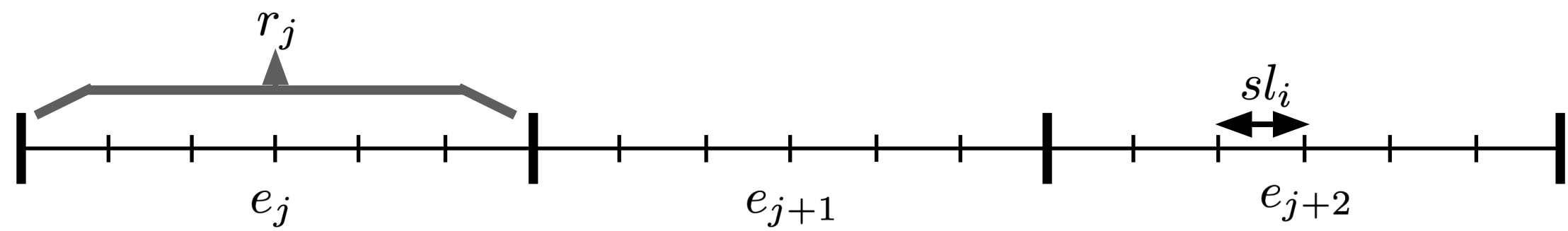
Validator



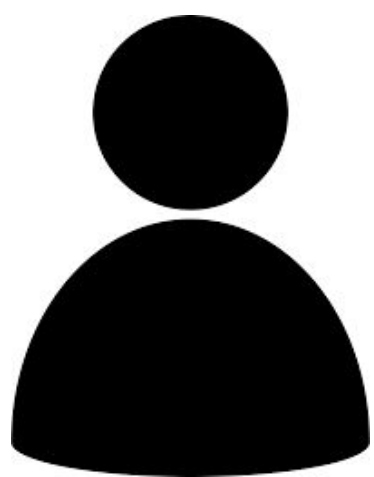
$$\mathbb{F} \left( \text{key}, r_j || sl_j || e_{j+2} \right) \longrightarrow v$$

$$v < \tau$$

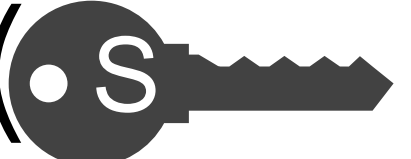
# BLOCK PRODUCTION



Validator

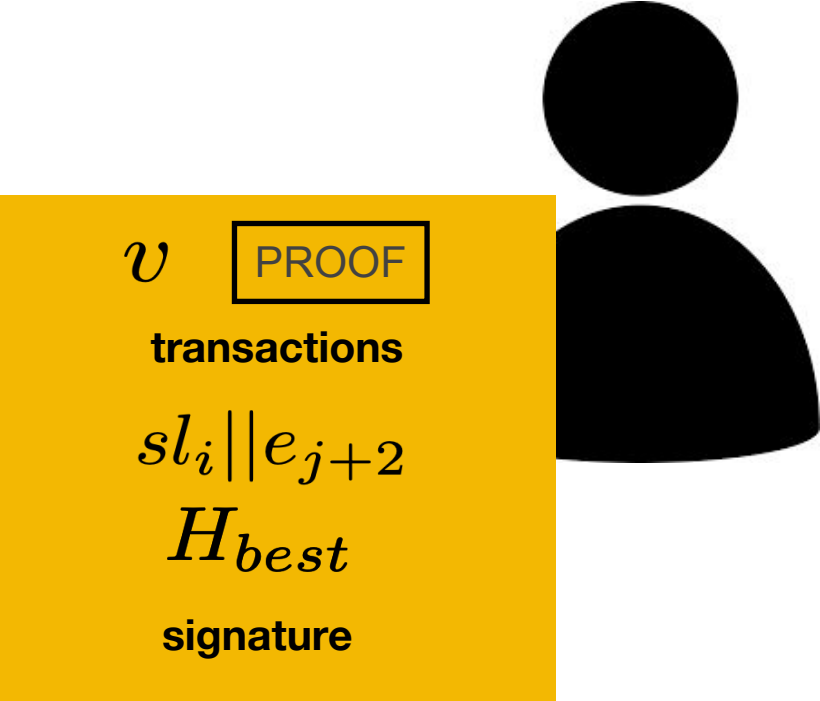
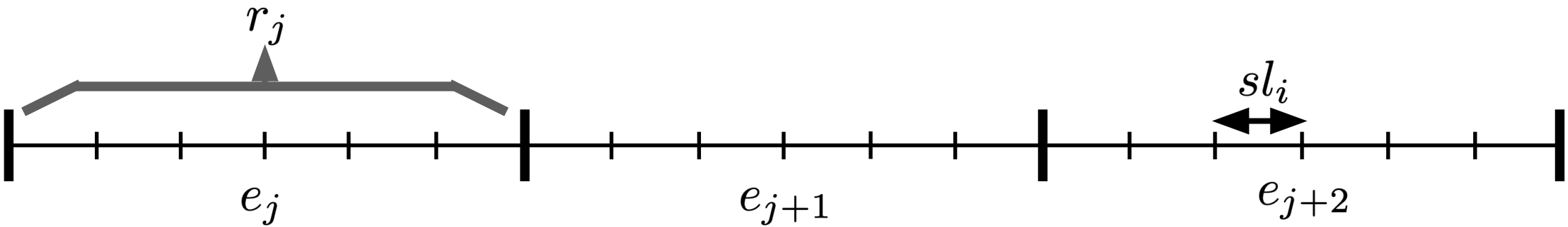


- if  $v < \tau$ , the validator produces a block for the slot  $sl_i$  in epoch  $e_{j+2}$

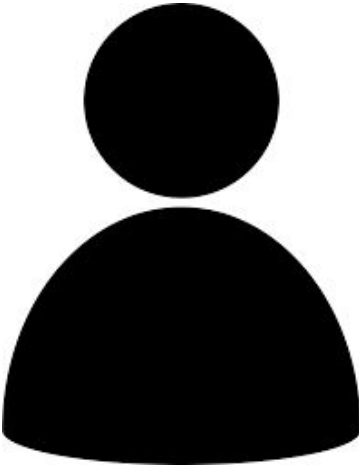
Prove (  ,  $r_j || sl_j || e_{j+2}$  ,  $v$  )  $\longrightarrow$  PROOF



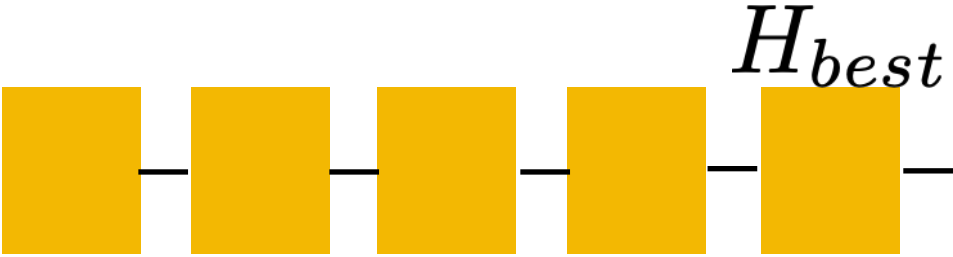
# CHAIN EXTENSION



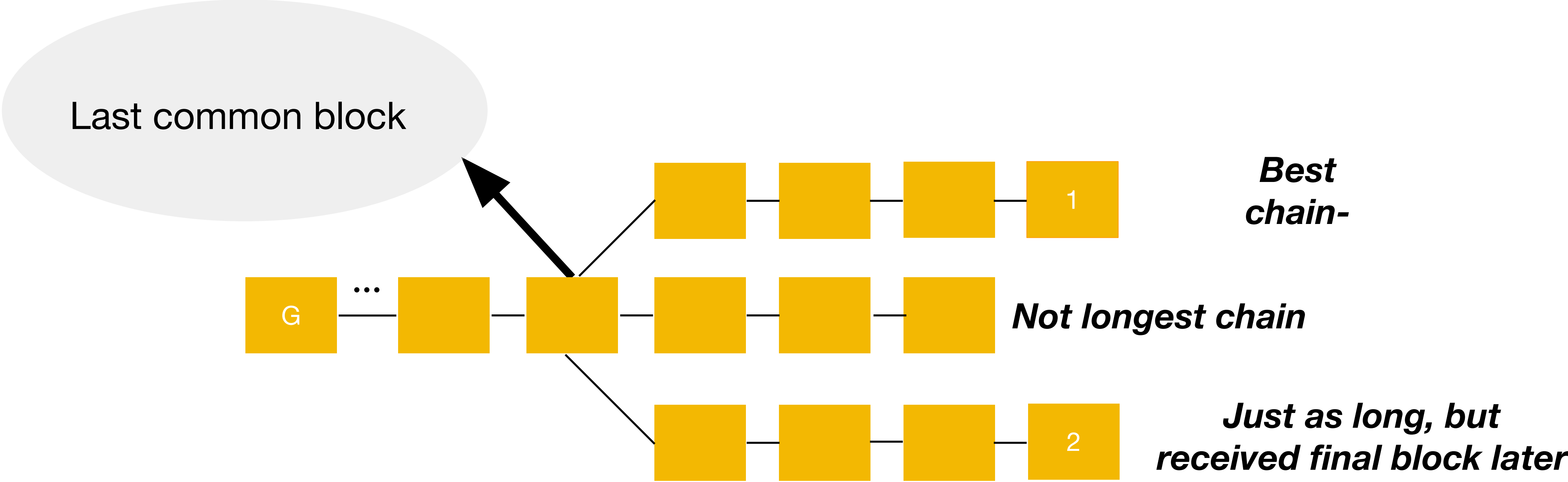
Validator



- if  $v \geq \tau$ , the validator collects blocks
  - Check if there is any chain with the hash
  - Verify the signature
  - Check if the validator is a slot leader
  - Check if there is any other block by the same validator for the same slot



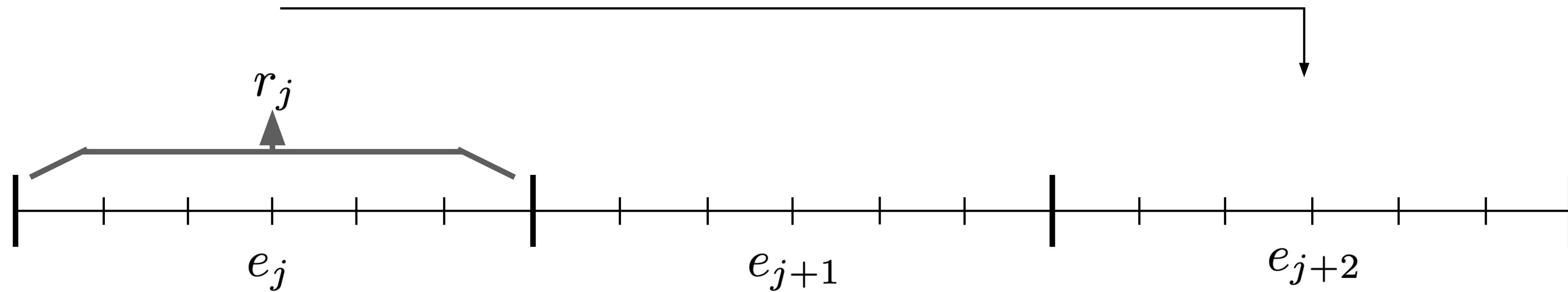
# BEST CHAIN SELECTION



Longest chain rule, with ties broken by first block received

# EPOCH RANDOMNESS

How to obtain epoch randomness



- Decide the best chain
- Retrieve all VRF values that belongs to the blocks of epoch  $e_j$

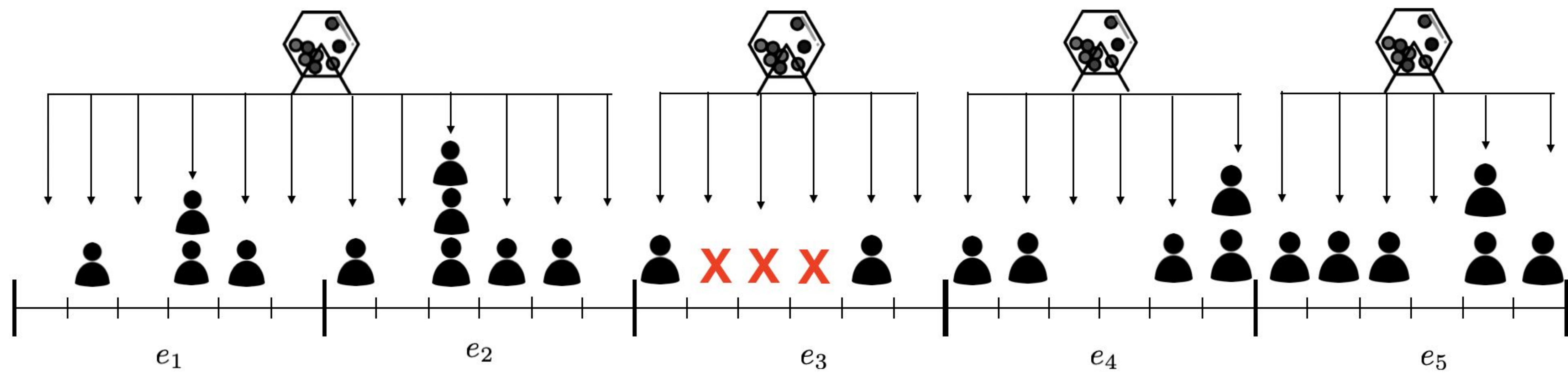
$v_1 - v_2 - v_3 - v_4 - v_5 - v_6 - v_7$

$$r_j = H(v_1 || v_2 || v_3 || v_4 || v_5 || v_6 || v_7)$$



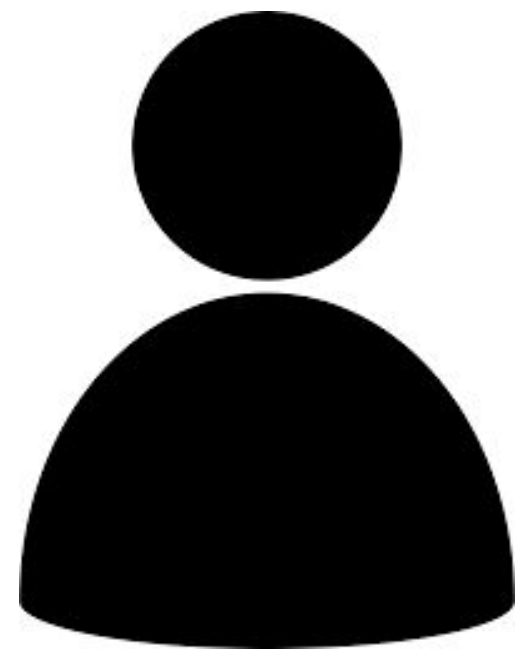
# ADJUSTMENT FOR CONSTANT BLOCK TIME

*Note long period of time (red X's) with no blocks!*



## Secondary Blocks

### Validator

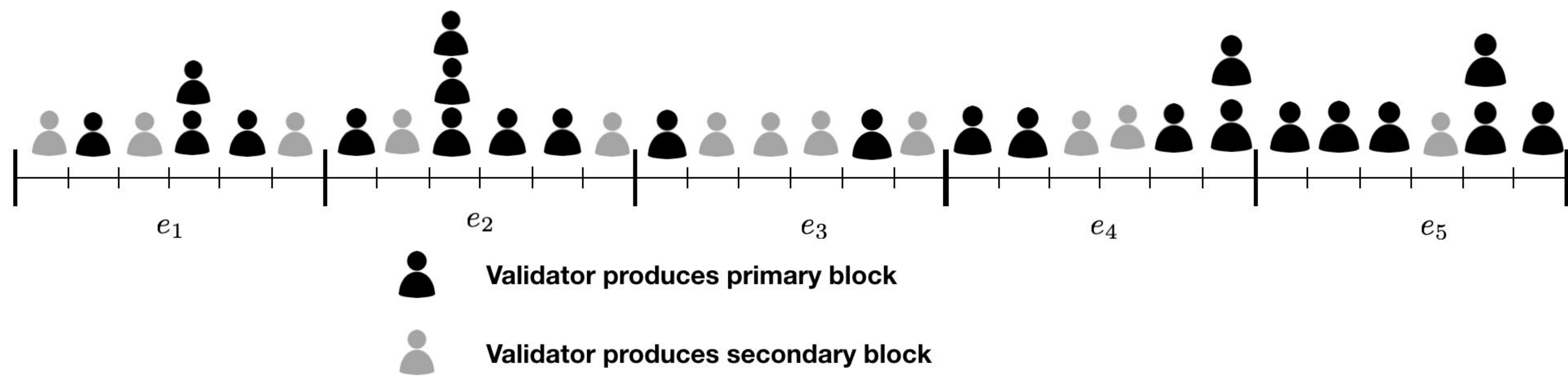


**Validator will produce a secondary block if it is their "turn" (determined by calculating if their authority number matches the one produced by the above algorithm).**

**Secondary blocks are "outweighed" by primary blocks if one is produced in that same slot. Secondary blocks provide less security than BABE, although everyone can verify that they did “take their turn” since all the information is public.**

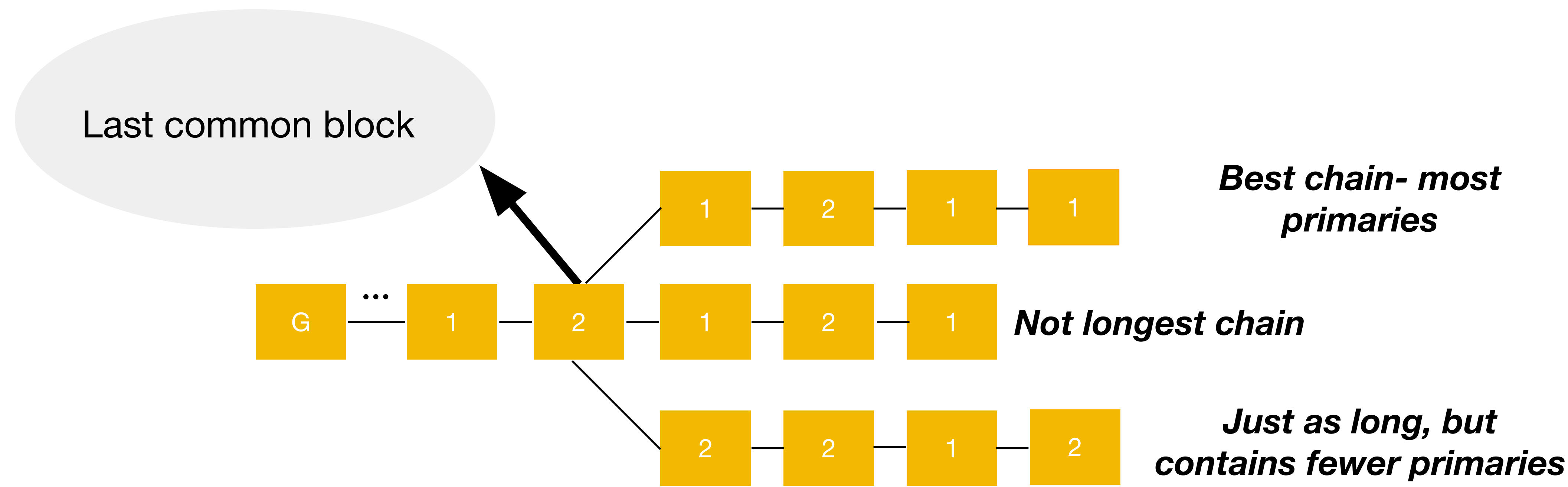
# ADJUSTMENT FOR CONSTANT BLOCK TIME

*Note there is now a block for every slot,  
although some are primary and some secondary.*





# BEST CHAIN SELECTION WITH SECONDARY BLOCKS

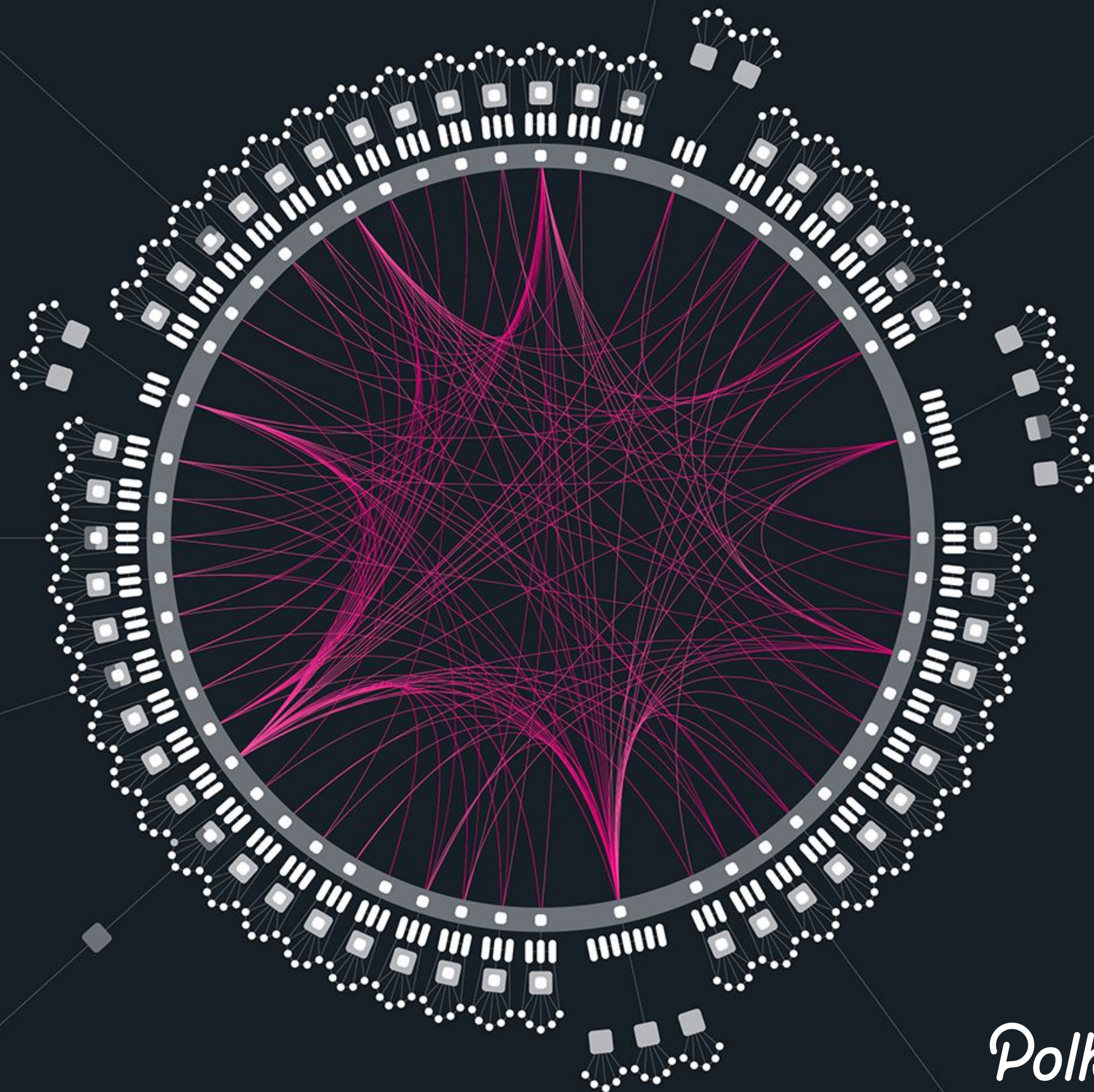


The longest chain as measured by number of primaries



# GRANDPA

Polkadot's Block  
Finalization Protocol



*Polkadot.*

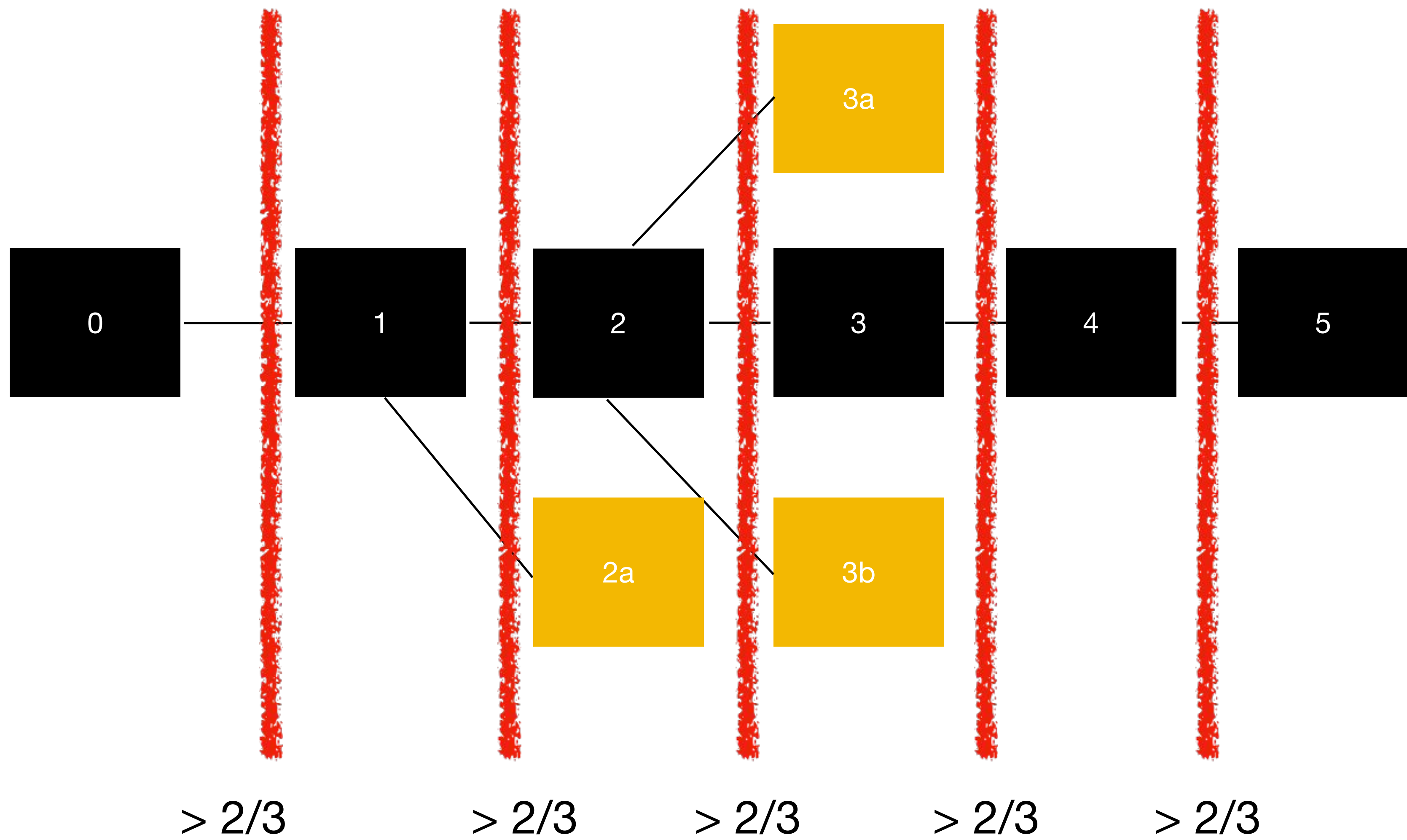


# Provable Deterministic Finalization

- We know the size of the validator set and that they should have roughly equal stake (using a weighted Phragmen algorithm)
- One way is to have a supermajority ( $2/3$ ) of validators "sign off" on a block
  - ◆ All validators considered equal, even if stake is not
- This is vulnerable to stalling in a Byzantine environment

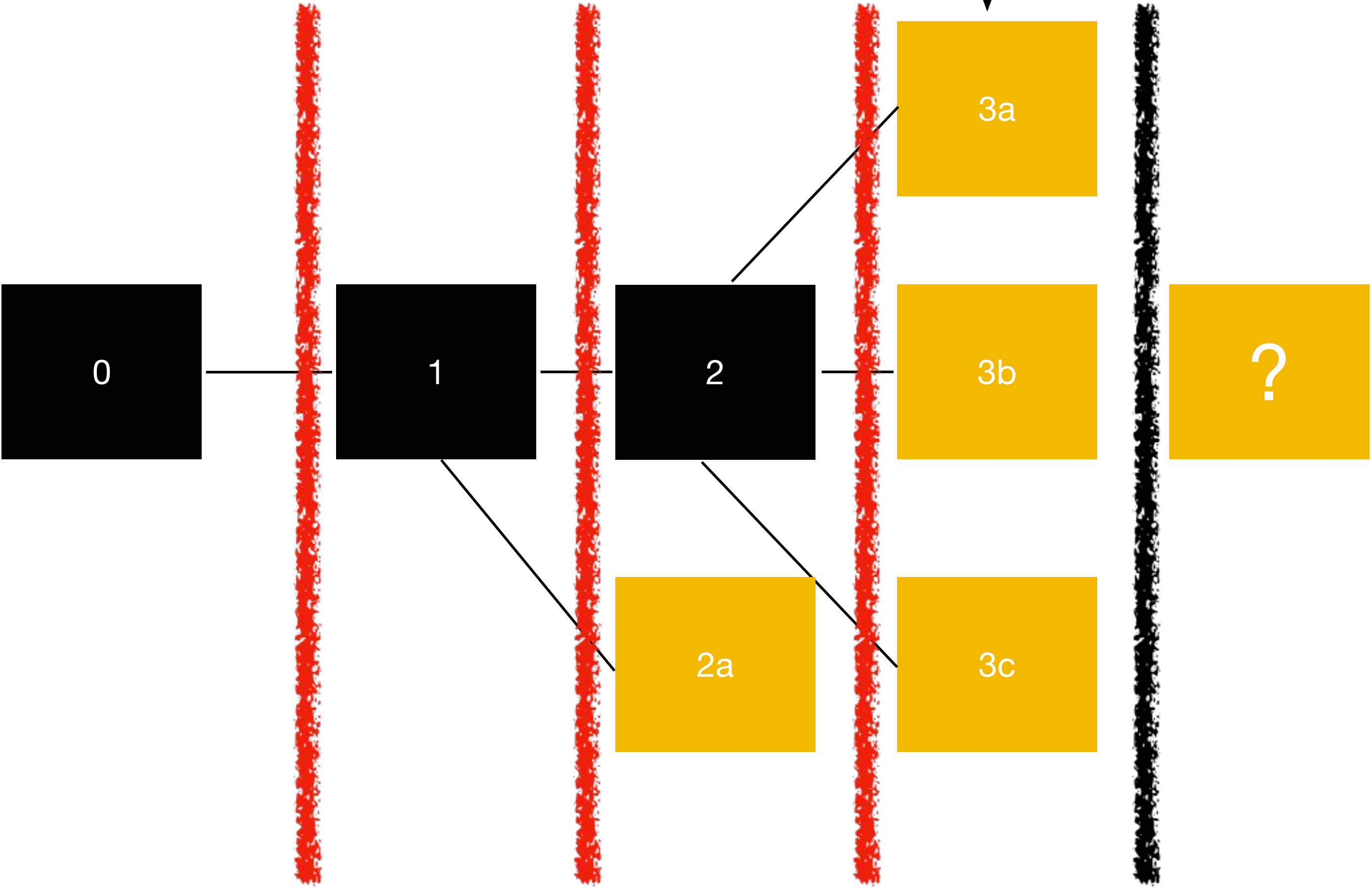


# CLASSIC BLOCK-BY-BLOCK BFT (SIMPLIFIED)



# Stalled Due to Partition

*40% of validators  
get separated or go offline here*



> 2/3

> 2/3

> 2/ 3

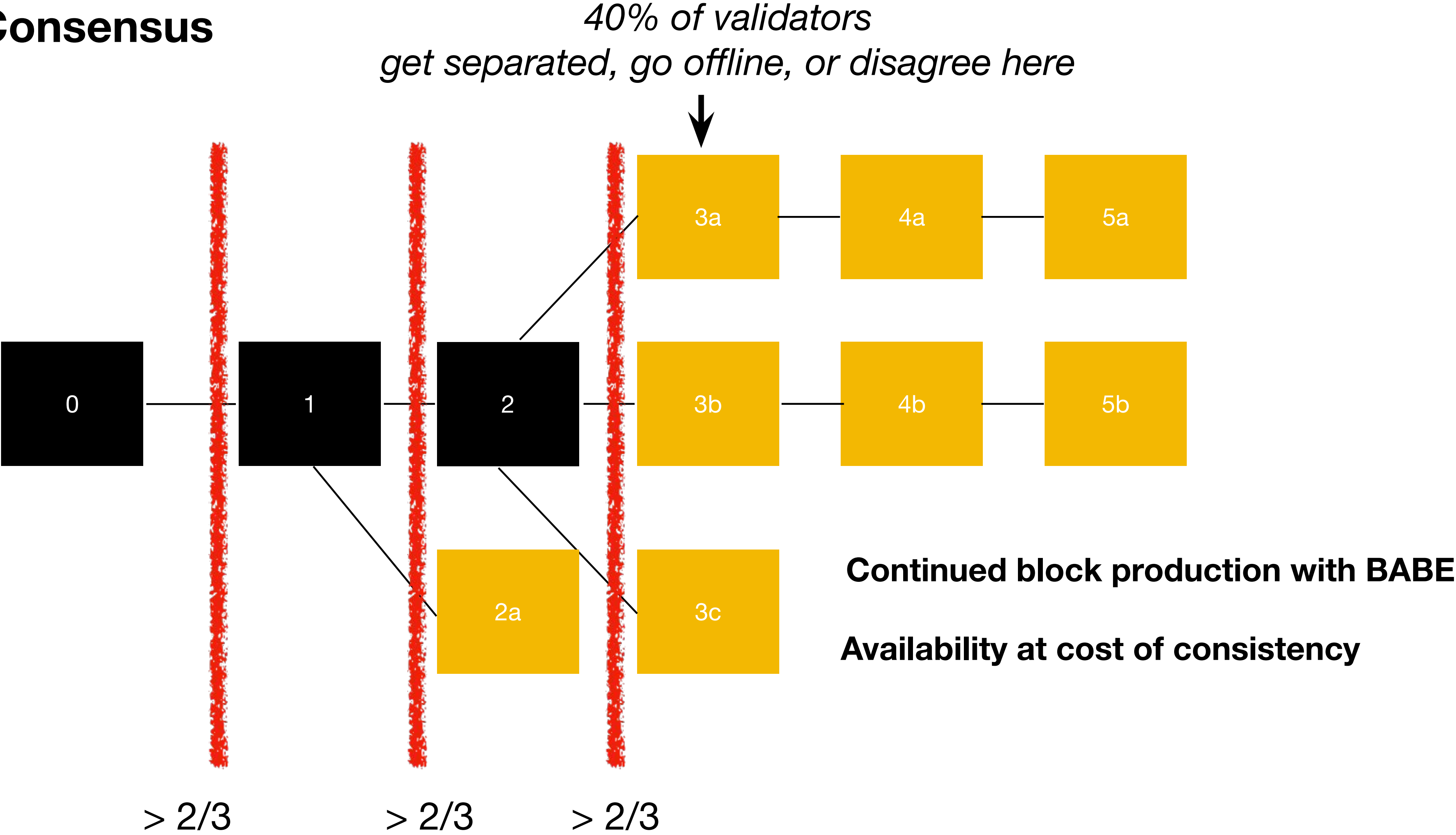
Impossible!  
> 2/3

# GRANDPA

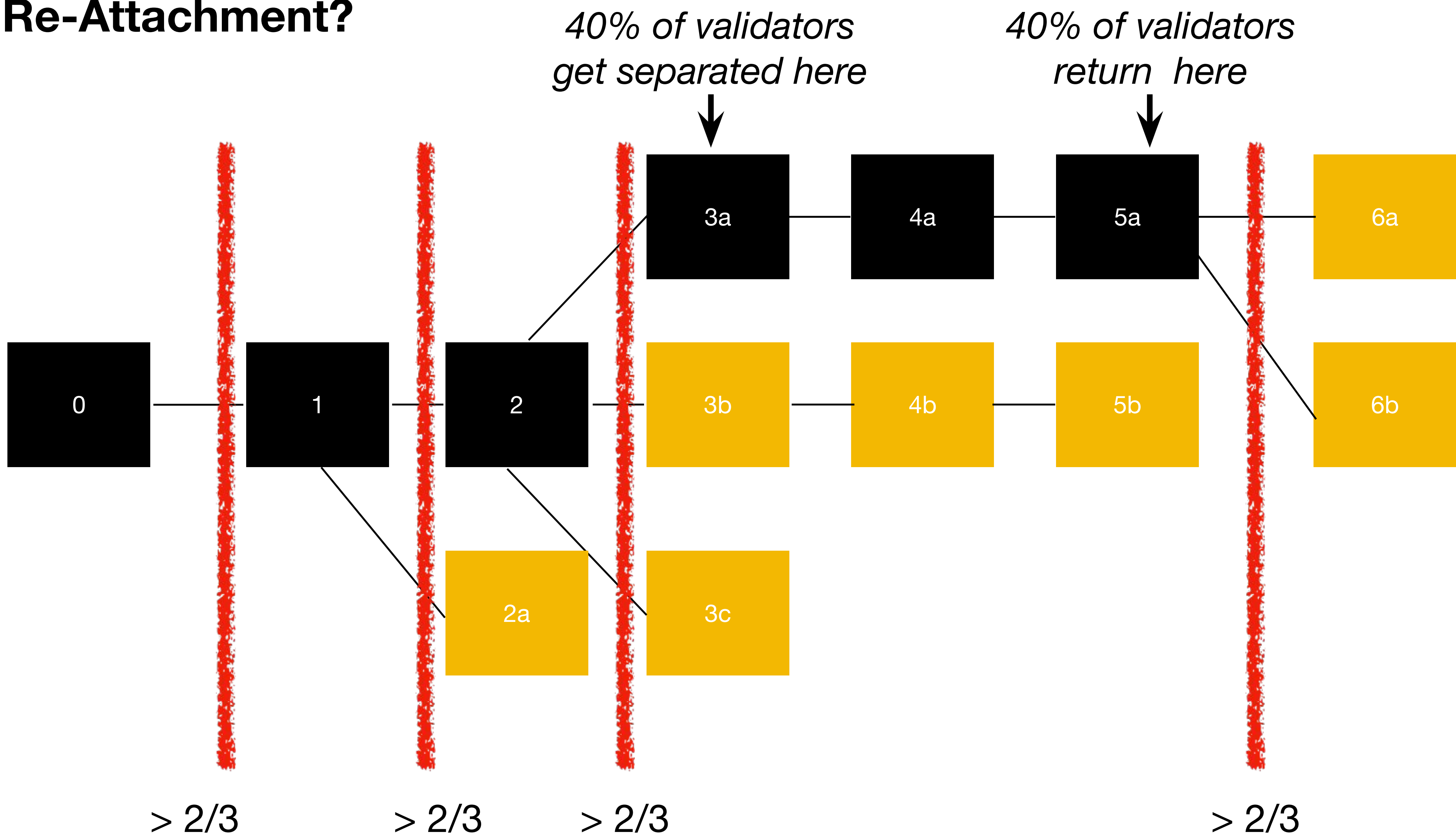
- GHOST-based Recursive ANcestor Deriving Prefix Agreement
- Byzantine agreement proposal
- Allows everyone to agree on a chain by following a simple fork rule choice
- Can work with many different block production mechanisms
- Goal: agree on many blocks at once, as opposed to single-block Byzantine agreement protocols
  - ◆ Since probabilistic finality block production may continue for a long time without GRANDPA, it needs to “catch up”



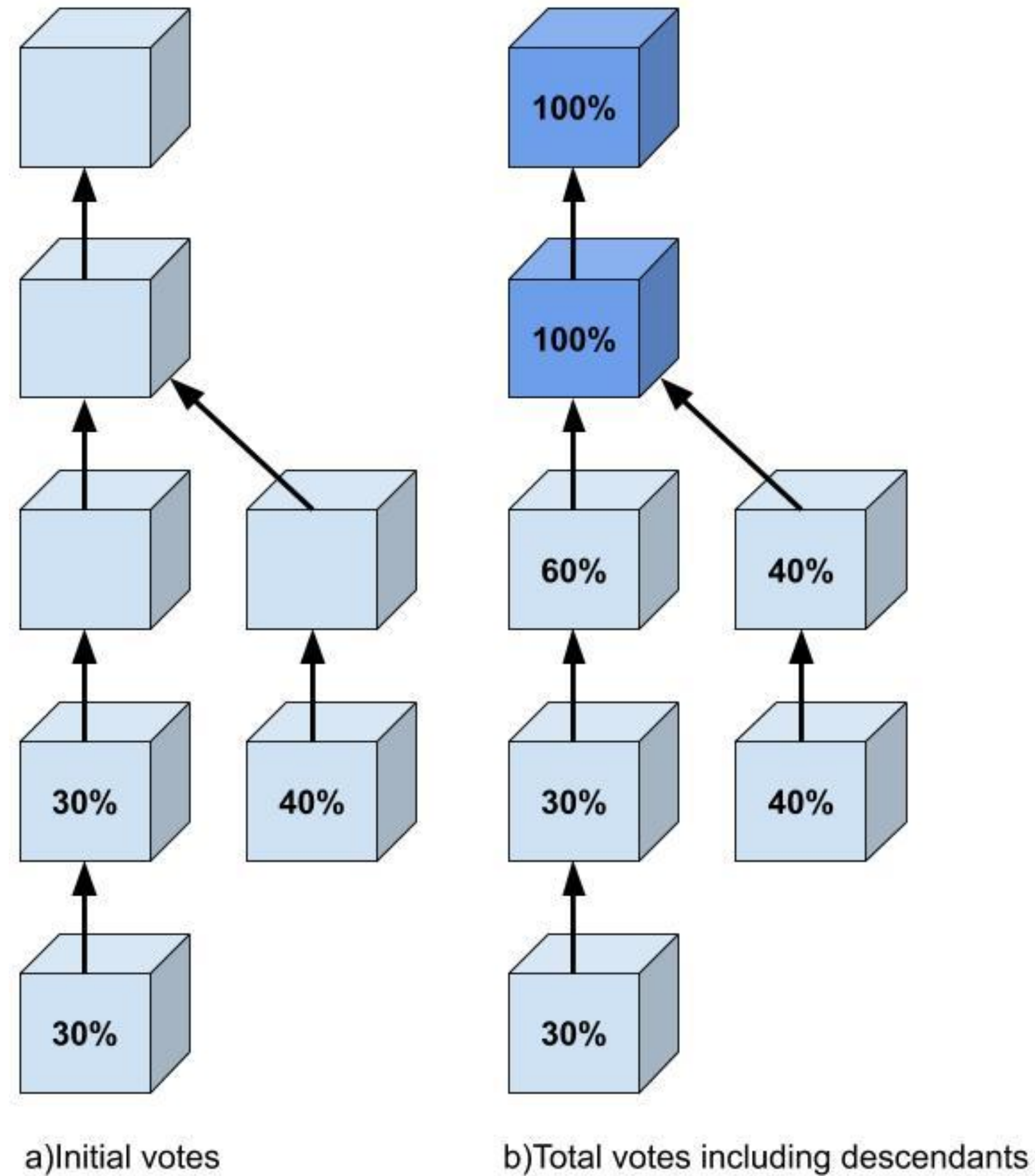
# Hybrid Consensus



# Re-Attachment?



# GRANDPA Voting Overview

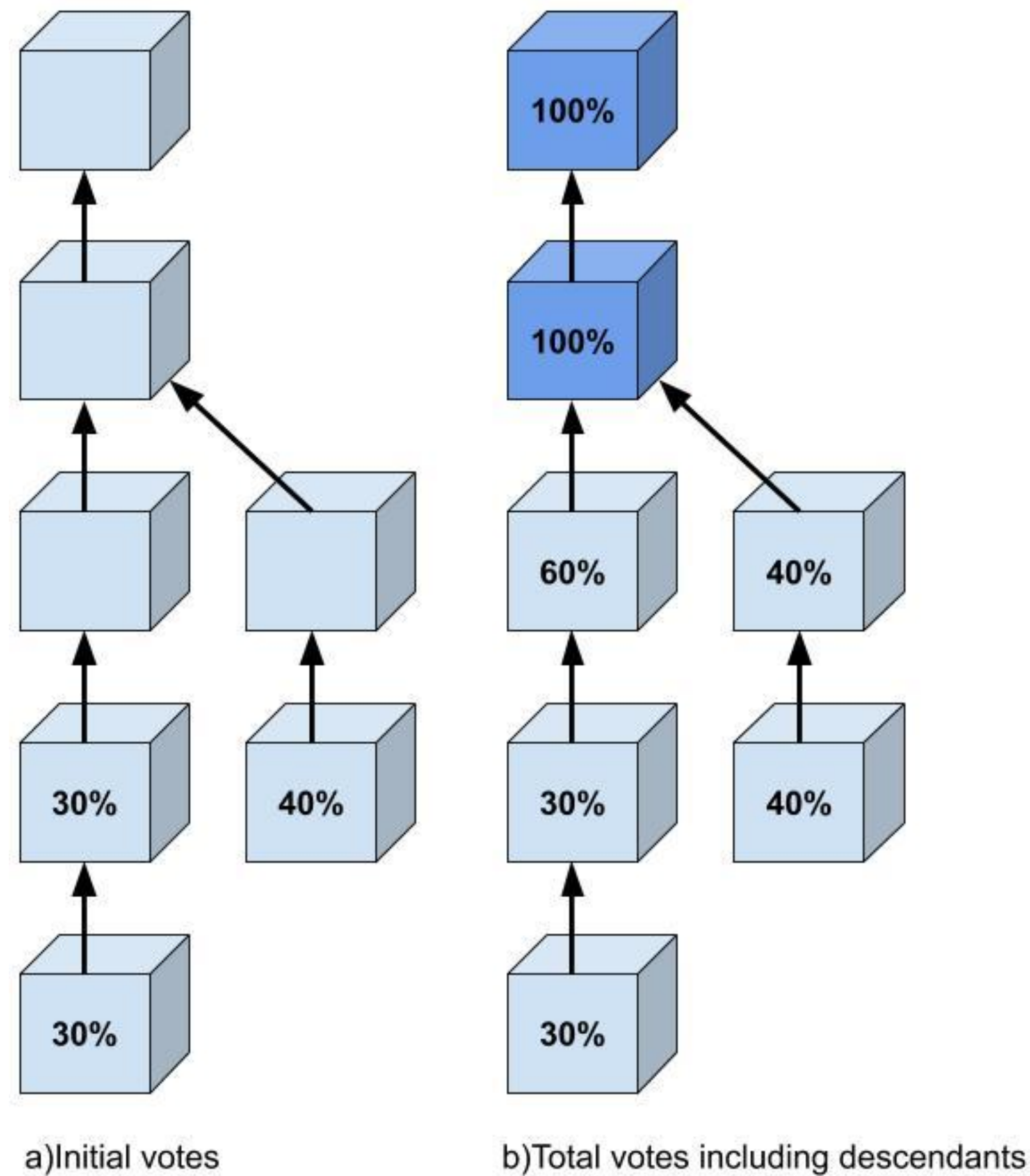




## 2/3 GHOST Rule

1. Start at genesis block
2. Include the child of that block that  $> \frac{2}{3}$  of voters voted for descendants of
3. The head of this chain is  $g(V)$ , where  $V$  is the set of all votes

# GRANDPA Voting Overview



# GRANDPA VOTING PHASES

## 1. Primary Selection

One validator is selected (in rotation) to be primary

They broadcast their estimate for the last round

## 2. Prevote

All validators prevote

Each validator applies  $\frac{2}{3}$ -GHOST rule,  $g$ , to set of prevotes they see,  $V$

## 3. Precommit

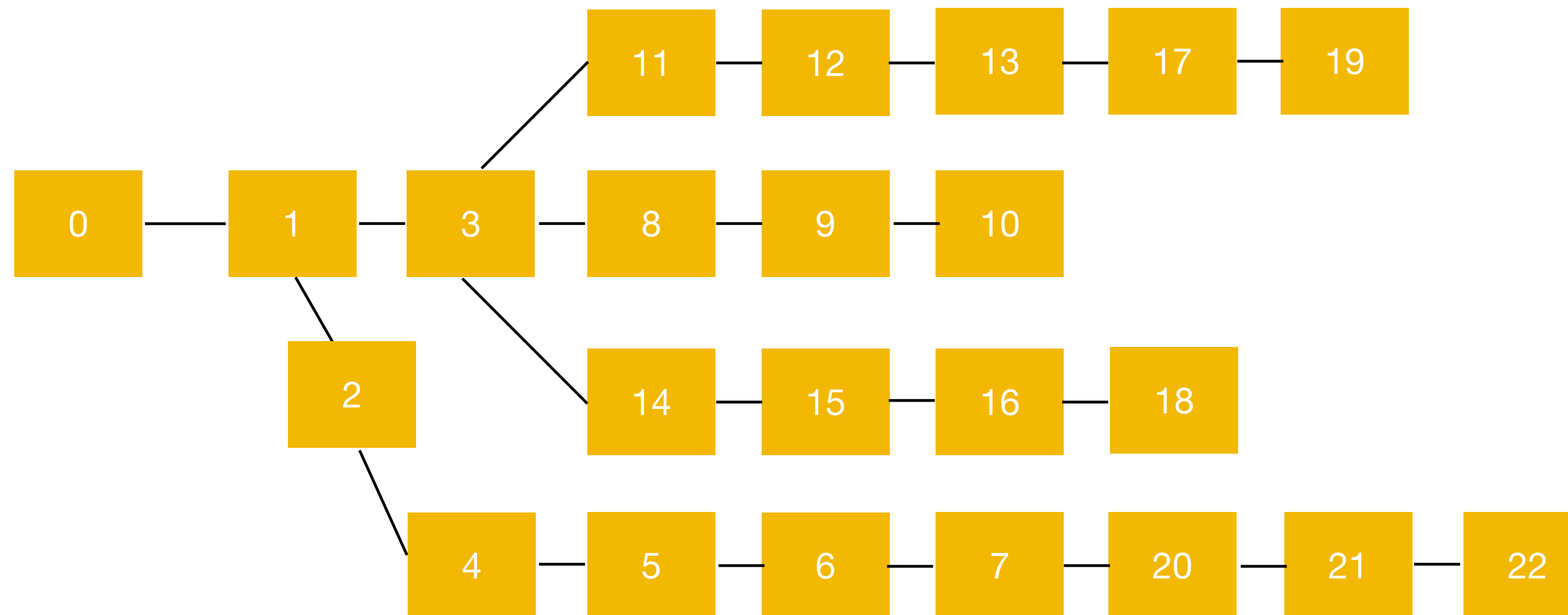
Each validator precommits and broadcasts  $g(V)$

Validators gather precommits that they see,  $C$

Validators finalize block indicated by  $g(C)$



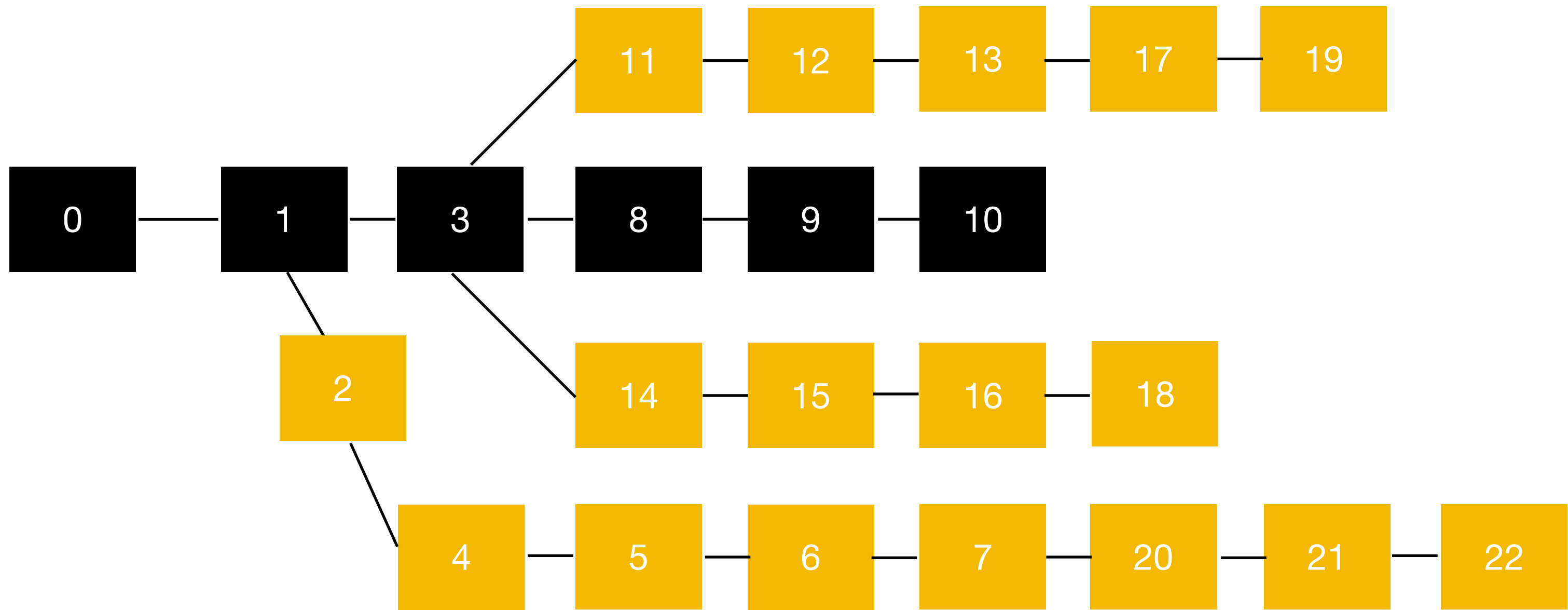
# VOTES ARE ON BLOCKS



All chains have at least one common ancestor;  
No conflicting chains will have a common  
descendant

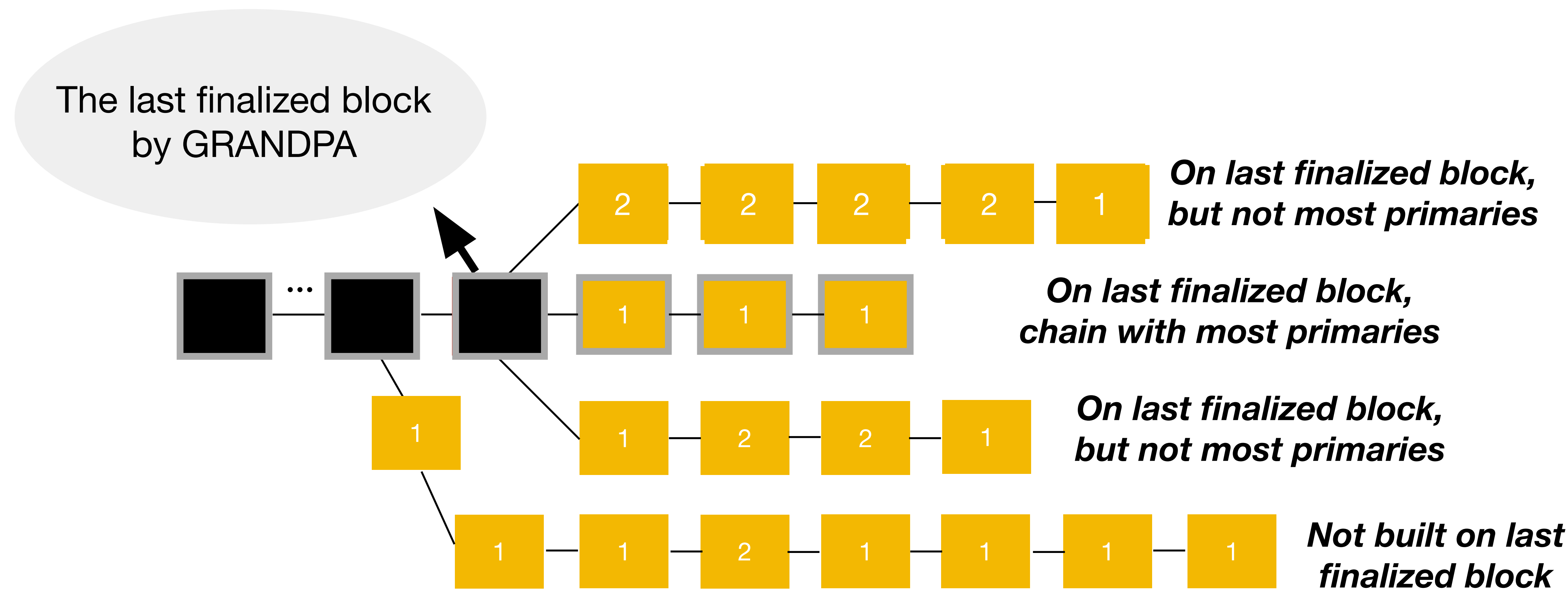
... WHICH ALLOWS FAST FINALIZATION OF CHAINS

Finalizing 10 implies a canonical chain  
0-1-3-8-9-10



Future BABE blocks should be built on top of the 0..10 chain

# FINAL BEST CHAIN SELECTION

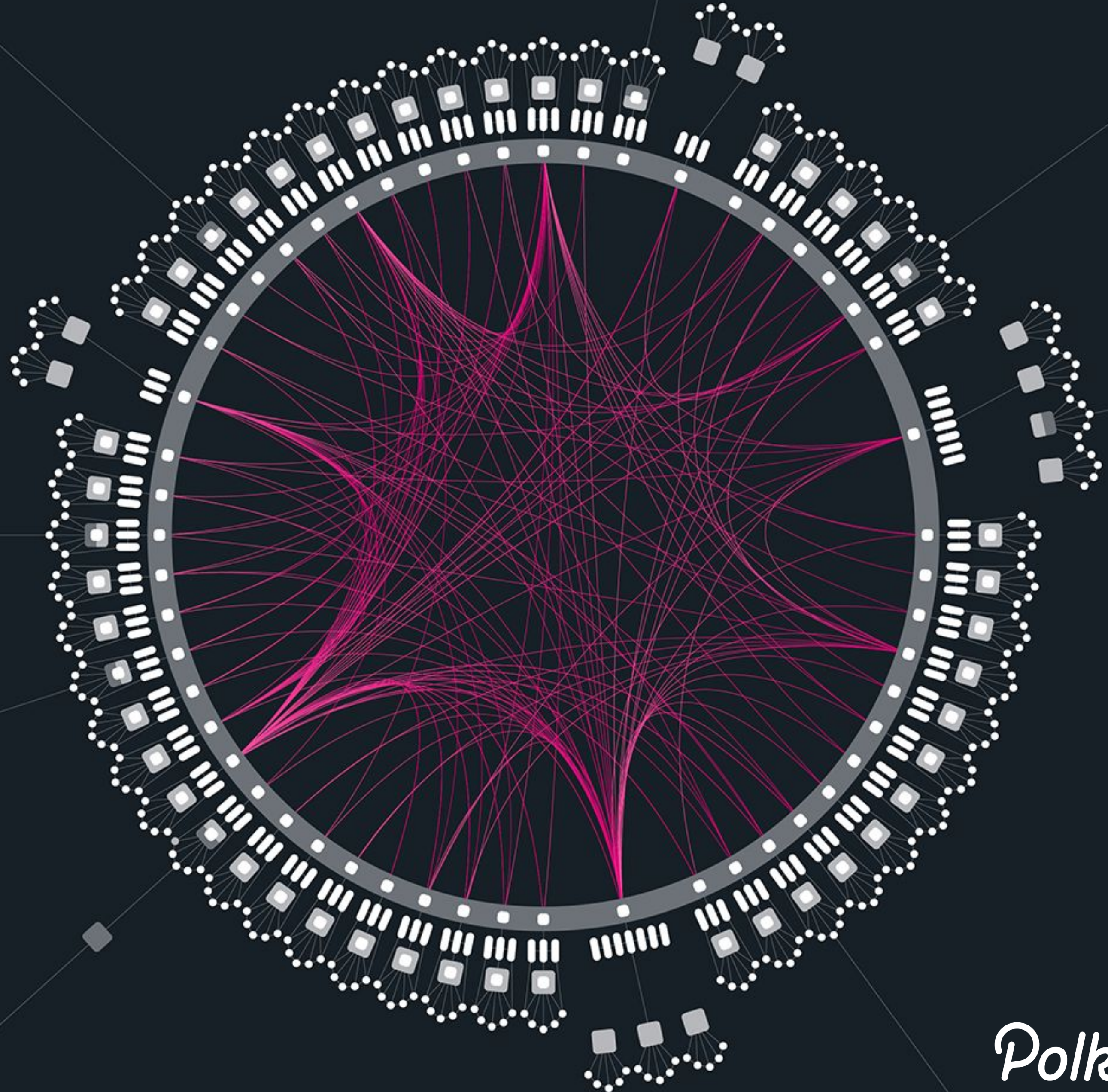


Longest chain (measured by primaries) on last finalized GRANDPA block



## PENALTIES

What happens if someone  
tries to cheat?



*Polkadot.*

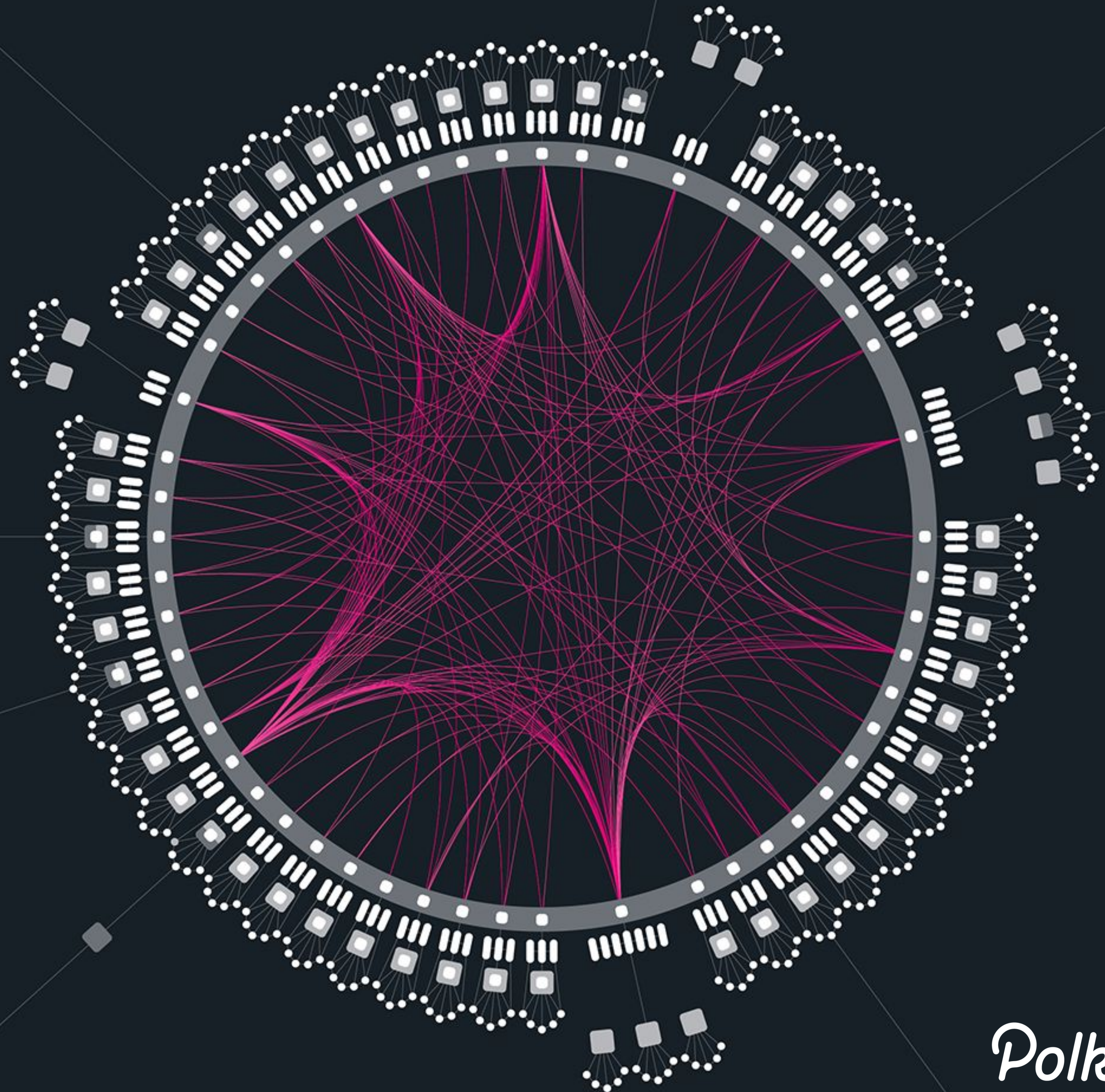


# Slashing

- A percentage of stake is deducted if you do something malicious or otherwise detrimental to the network
- Slashing penalties increase as number of participants increase
- Information about potential slashing penalties is kept off-chain by validators
- No matter what chain ends up being finalized, malicious or incompetent validators can be slashed on that chain



CONCLUSION



*Polkadot.*



# HYBRID CONSENSUS

- Block production (via BABE) and finalization (via GRANDPA) are separate mechanisms
- Allows provable finality without the drawbacks of stalling, with fallback to probabilistic finality in cases of network partition or malicious validator behavior
- Used in Polkadot (mainnet coming soon) and canary network, Kusama, running since August 2019



Polkadot.

## **RESEARCH**

<https://research.web3.foundation/>

## **GET IN TOUCH**

**Bill Laboon**

Technical Education Lead at Web3 Foundation

Twitter: [@BillLaboon](#)

Email: [bill@web3.foundation](mailto:bill@web3.foundation)

### **Join us**

<https://polkadot.network>

<https://web3.foundation>

### **Connect**

