How long non-finality could kill Ethereum

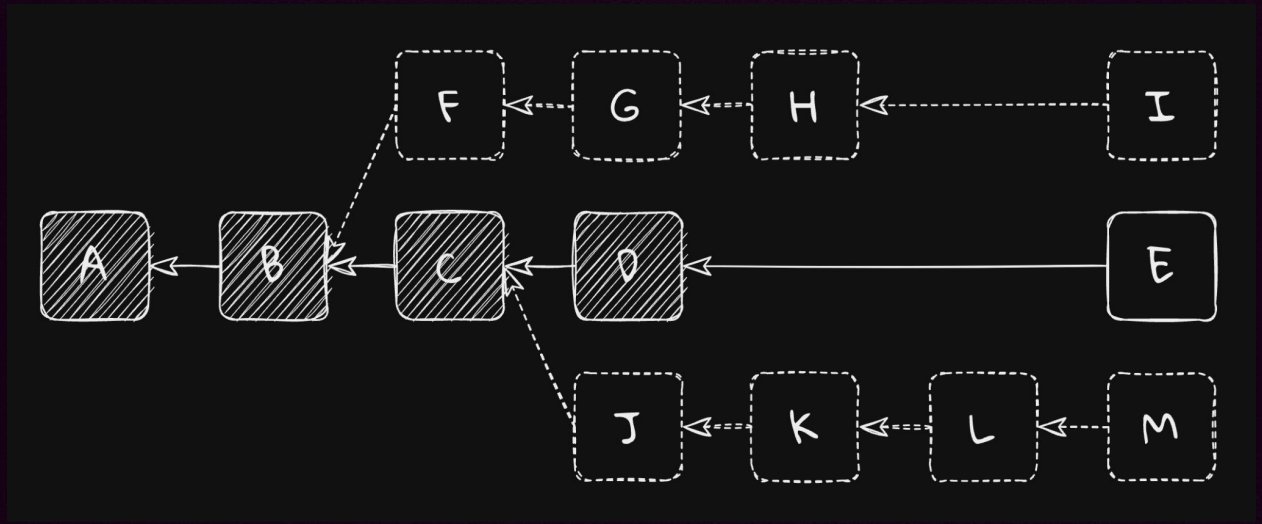
..or a healthy dose of fear-mongering to get people prioritize this

dapplion

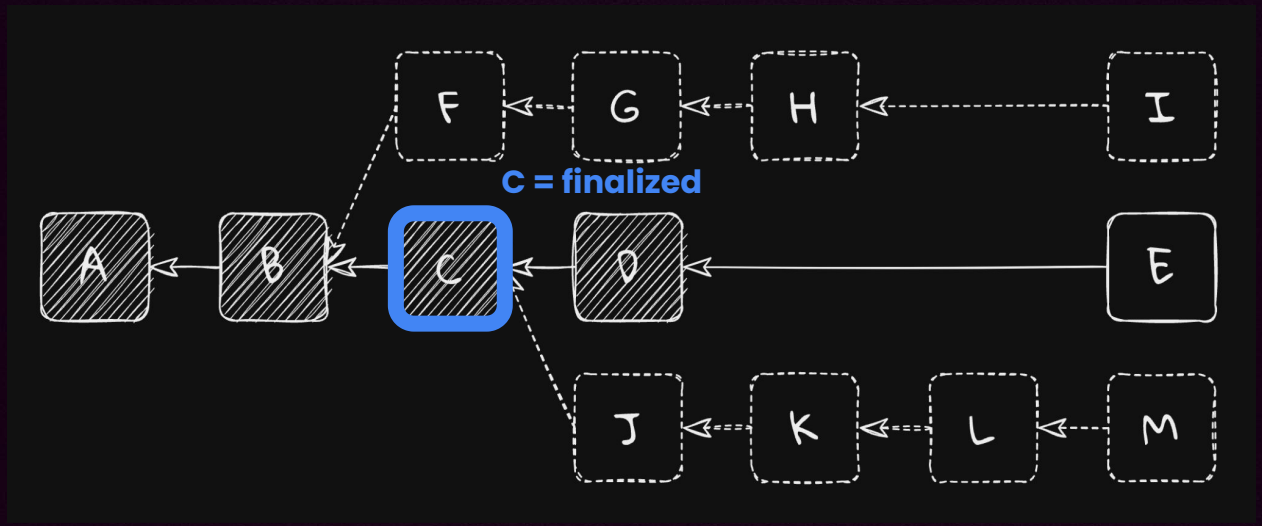
Core dev @ Sigma Prime

The failure mode

What's finality?



What's finality?

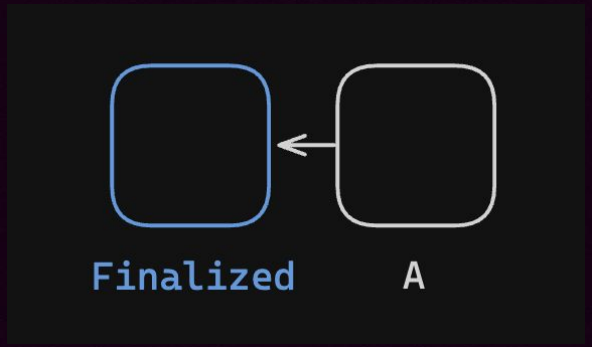


❌

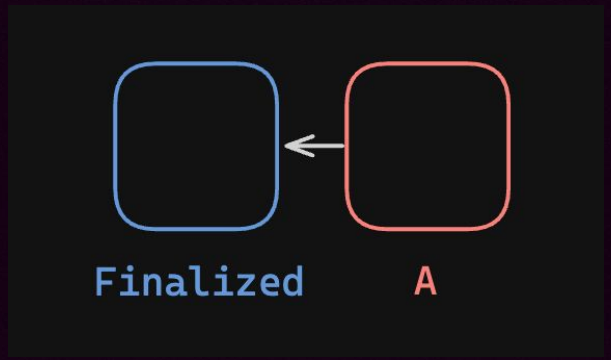
✅

✅

A non-finality case

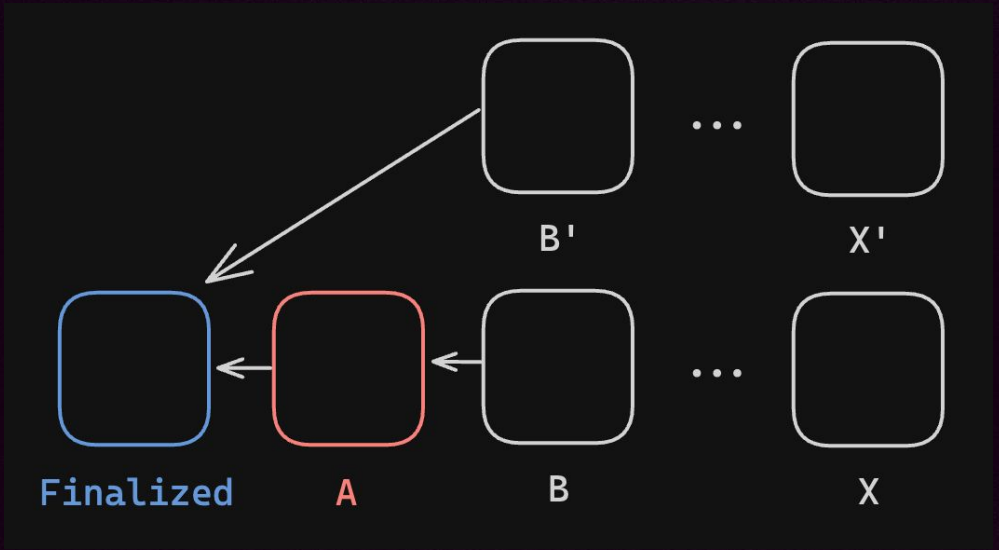


A non-finality case



Ups! Block A is rejected by a client that has >33% stake

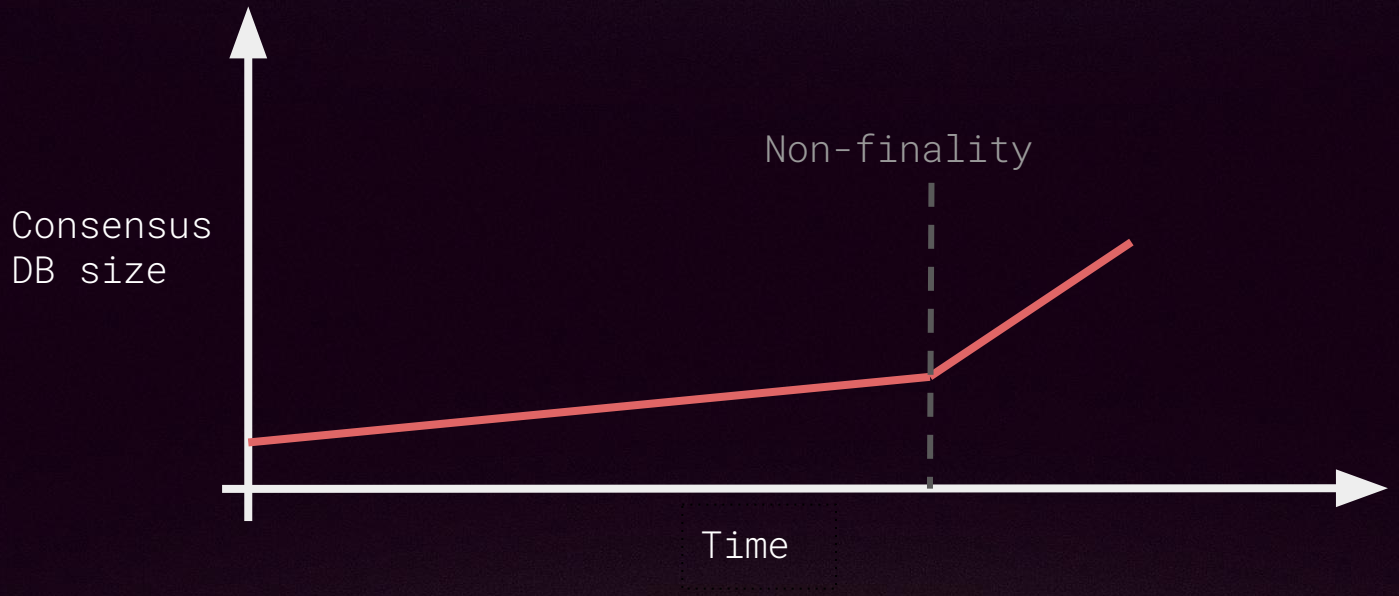
A non-finality case



Faulty client builds its own fork

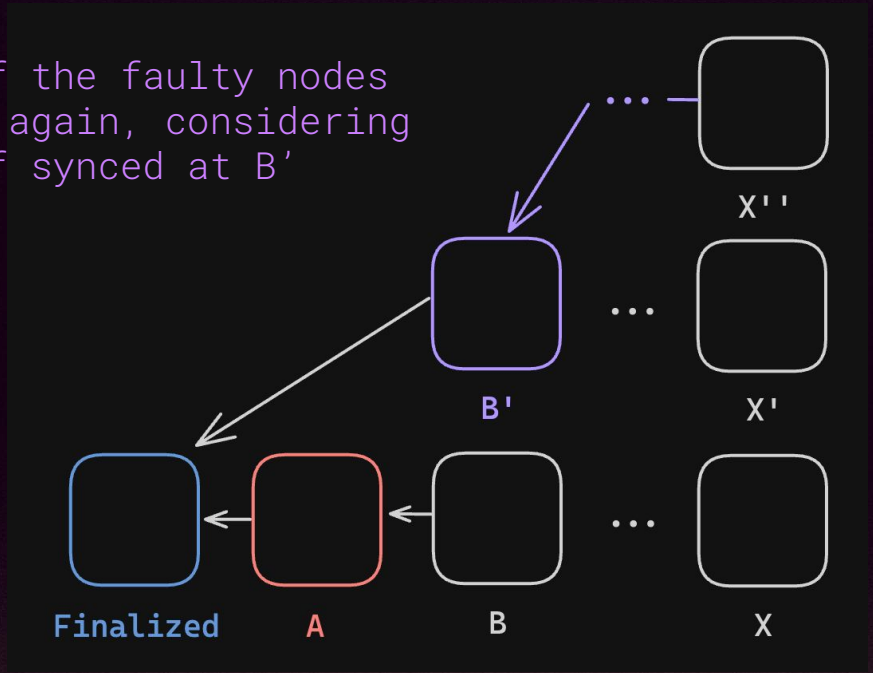
Other clients build on the majority fork

A non-finality case



A non-finality case

One of the faulty nodes fails again, considering itself synced at B'



Proposes a very expensive block X''

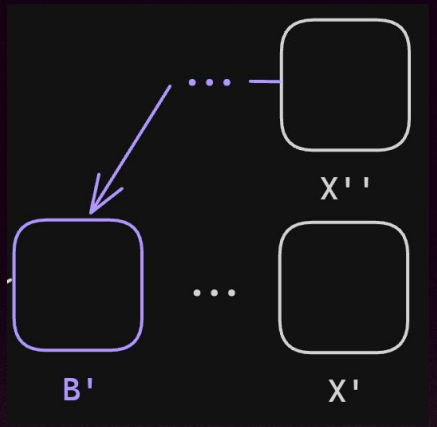
..that everyone else must process

(tangent) Skips slots and epoch transitions

2 - Advance post state to X’’ slot

3 - Apply X’’ block

1 - Grab B’ post state



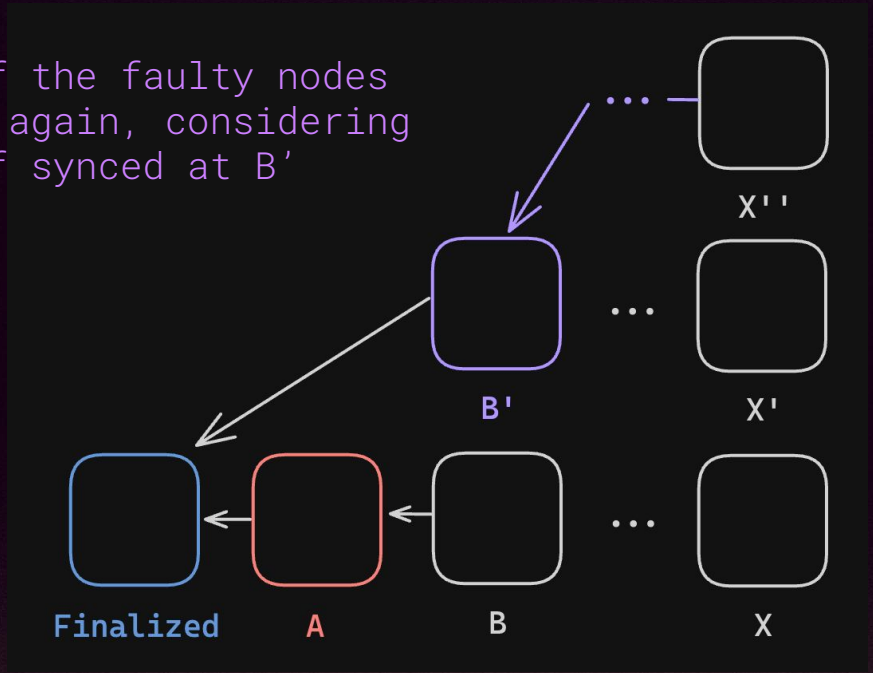
(tangent) Skips slots and epoch transitions

```python
def process_epoch(state: BeaconState) -> None:
    process_justification_and_finalization(state)
    process_inactivity_updates(state)
    process_rewards_and_penalties(state)
    process_registry_updates(state)  # [Modified in Electra:EIP7251]
    process_slashings(state)  # [Modified in Electra:EIP7251]
    process_eth1_data_reset(state)
    process_pending_deposits(state)  # [New in Electra:EIP7251]
    process_pending_consolidations(state)  # [New in Electra:EIP7251]
    process_effective_balance_updates(state)  # [Modified in Electra:EIP7251]
    process_slashings_reset(state)
    process_randao_mixes_reset(state)
    process_historical_summaries_update(state)
    process_participation_flag_updates(state)
    process_sync_committee_updates(state)
```

x 225 times + hashing!

A non-finality case

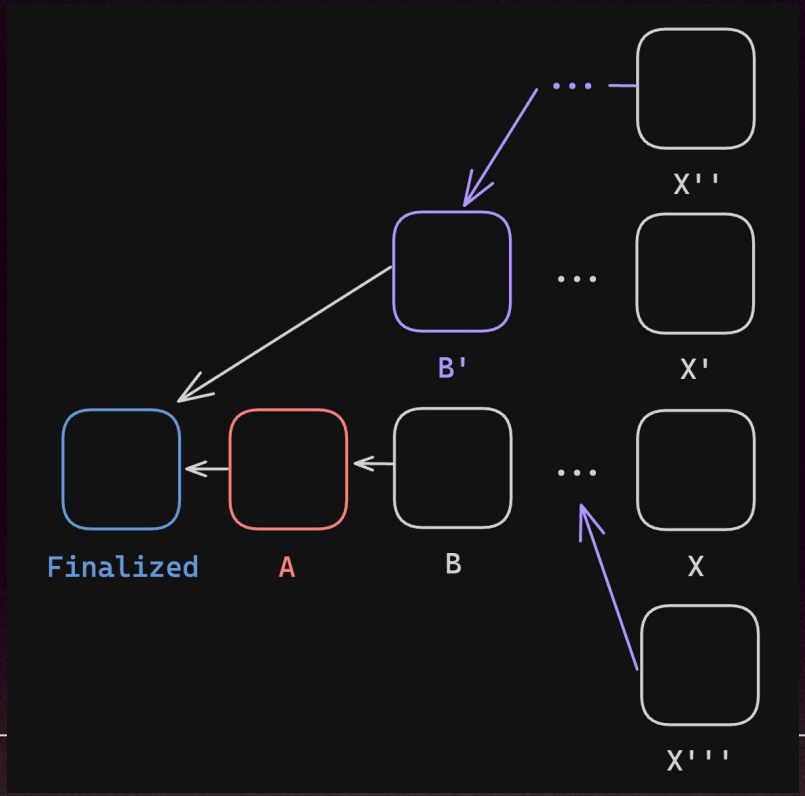
One of the faulty nodes fails again, considering itself synced at B'



Proposes a very expensive block X''

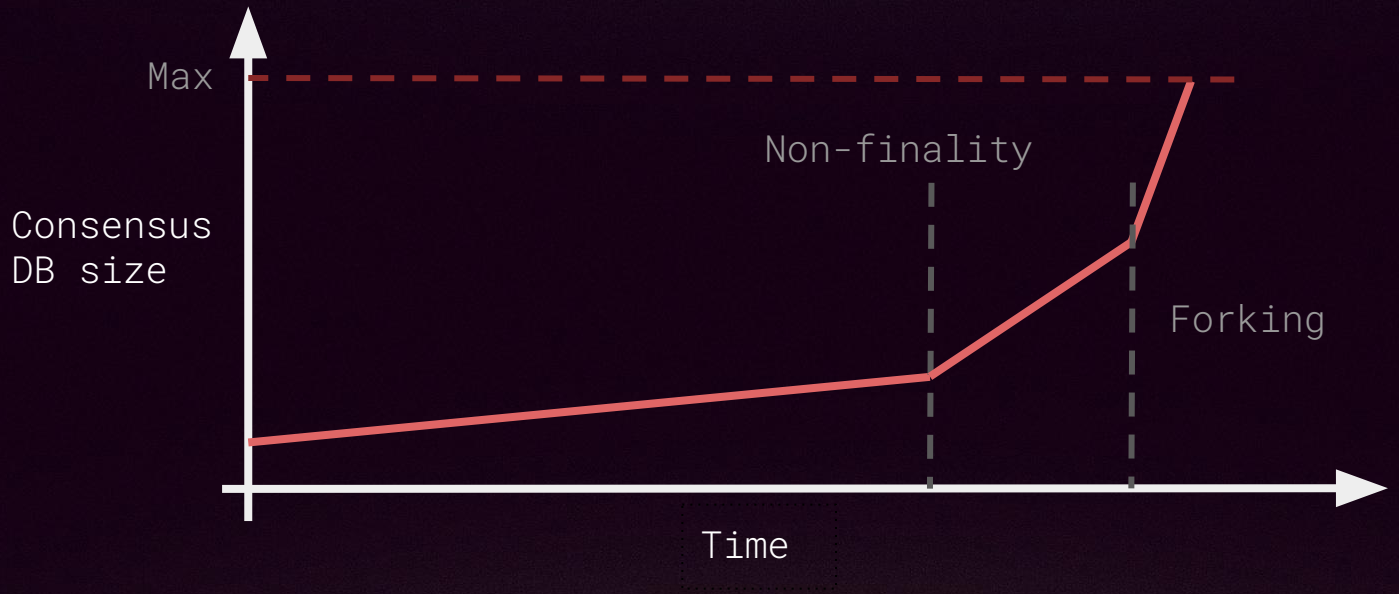
..that everyone else must process

A non-finality case

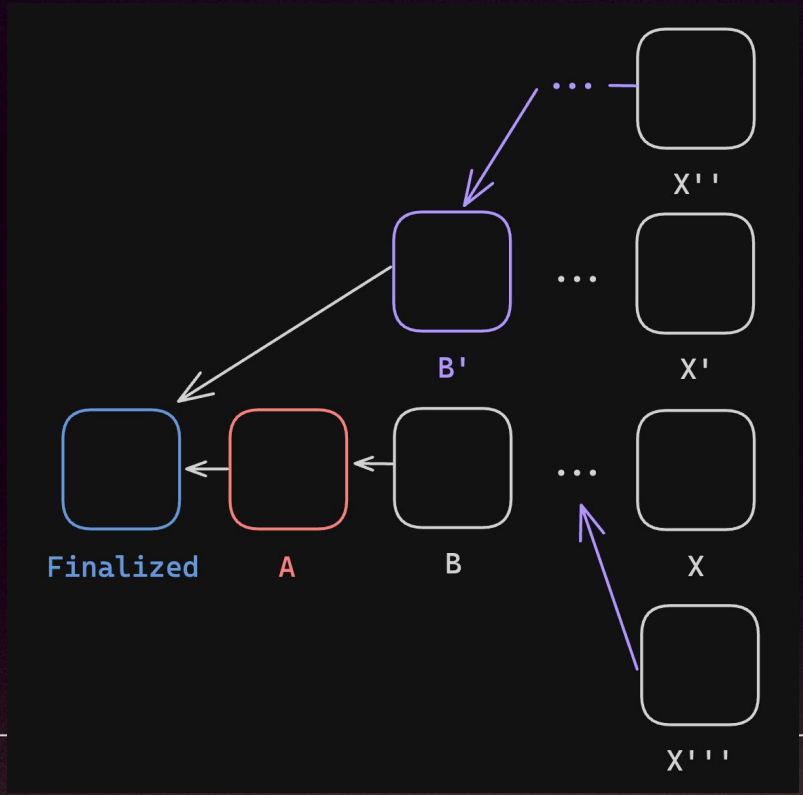


The expensive X'' cause other nodes to fall out of sync and start building on new forks

A non-finality case



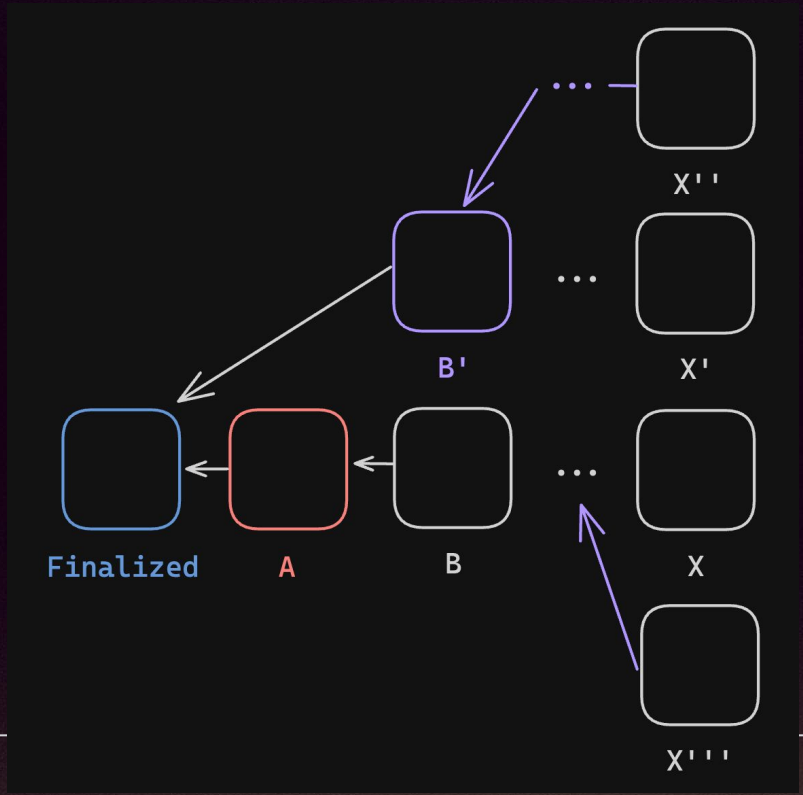
A non-finality case



Branches may lose LMD-Ghost weight due to nodes going offline (full disk)

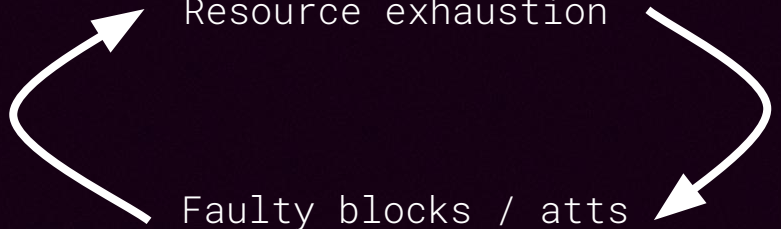
..eventually causing a re-org (too deep for EL)

A non-finality case

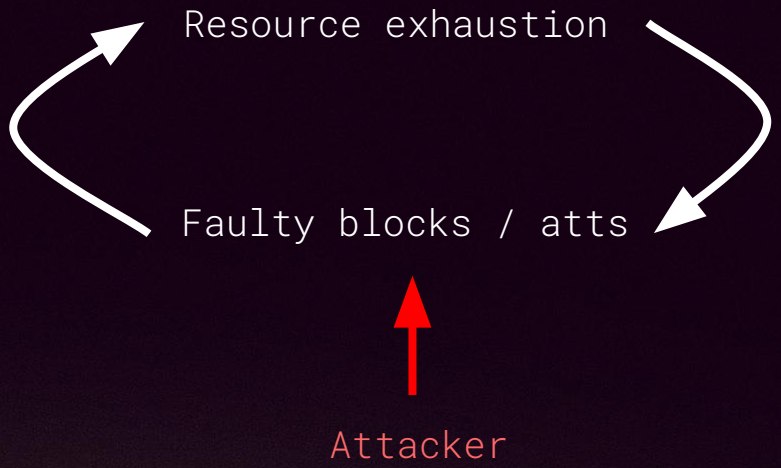


Crashing clients (i.e. OOM) have to sync all forks, from already overwhelmed nodes. Sync issues cause more rogue blocks like X''

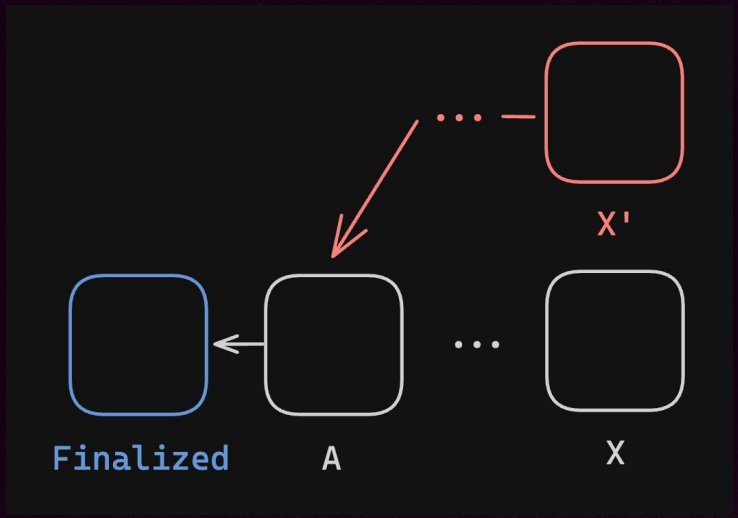
When proposing a block, the set of possible pre-states increases quadratically with time to last finalized checkpoint



When proposing a block, the set of possible pre-states increases quadratically with time to last finalized checkpoint



Malicious attacks



An attacker can produce this very expensive blocks / attestations and DOS the network

Malicious attacks

Blocks

- Require stake
- Must bruteforce a valid proposal slot

Aggregated attestations

- Require stake
- Must bruteforce a valid aggregator duty

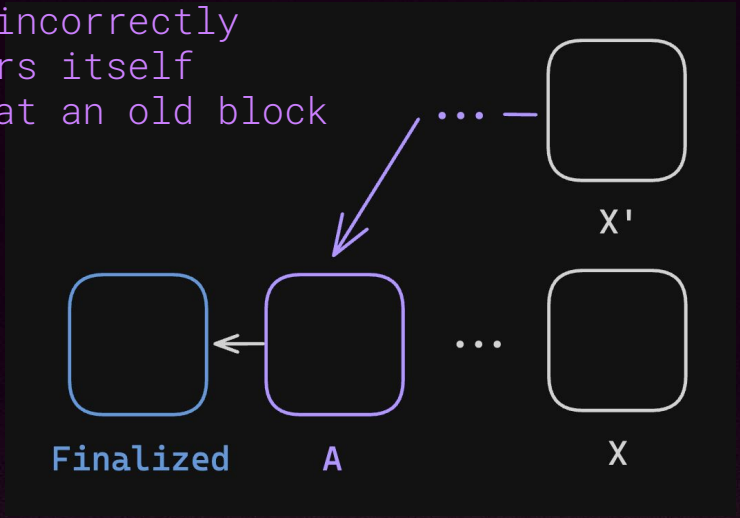
Unaggregated attestations

- Do not require stake
- If invalid, must target each node

Past incidents: Mainnet 2023

Past incidents: Mainnet 2023

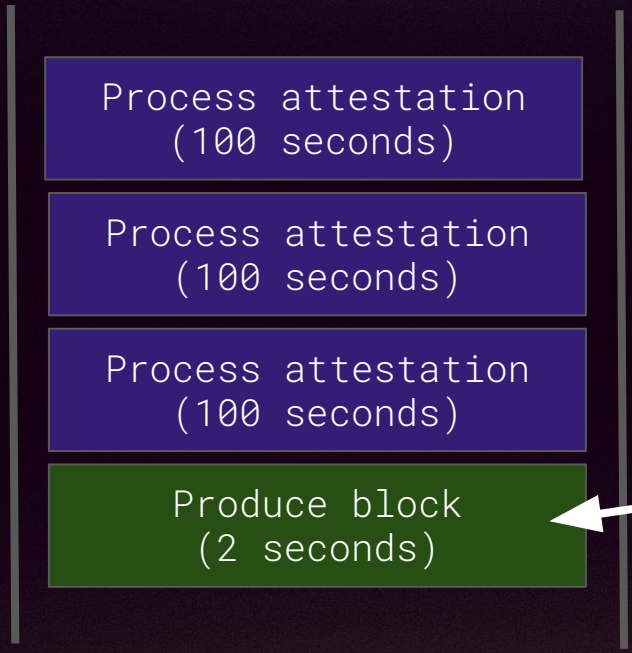
A node incorrectly considers itself synced at an old block



Produces (a lot of) attestations that are expensive to process

..other clients choke attempting to process all of them

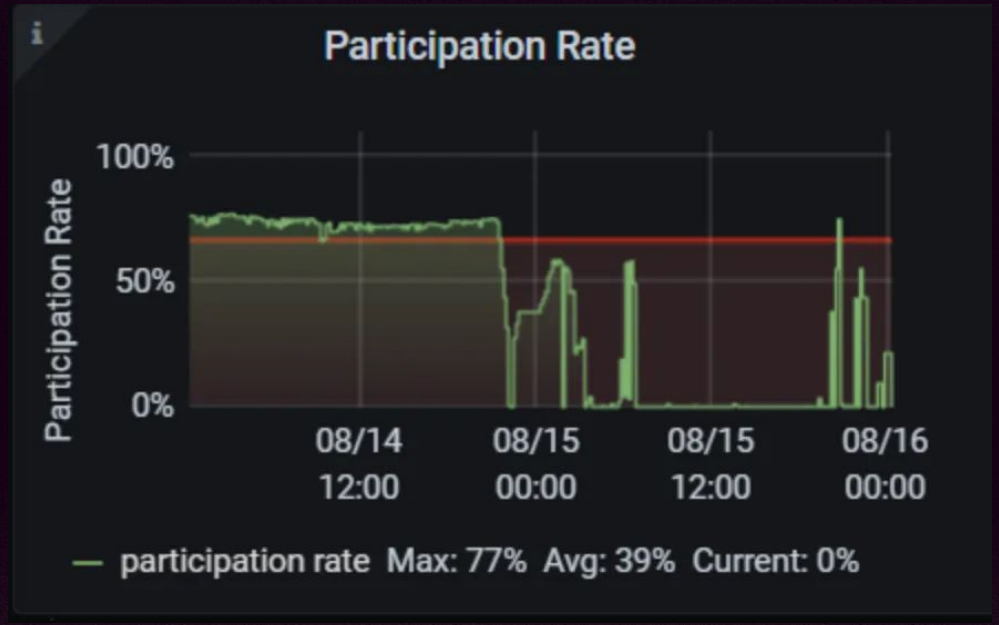
Past incidents: Mainnet 2023



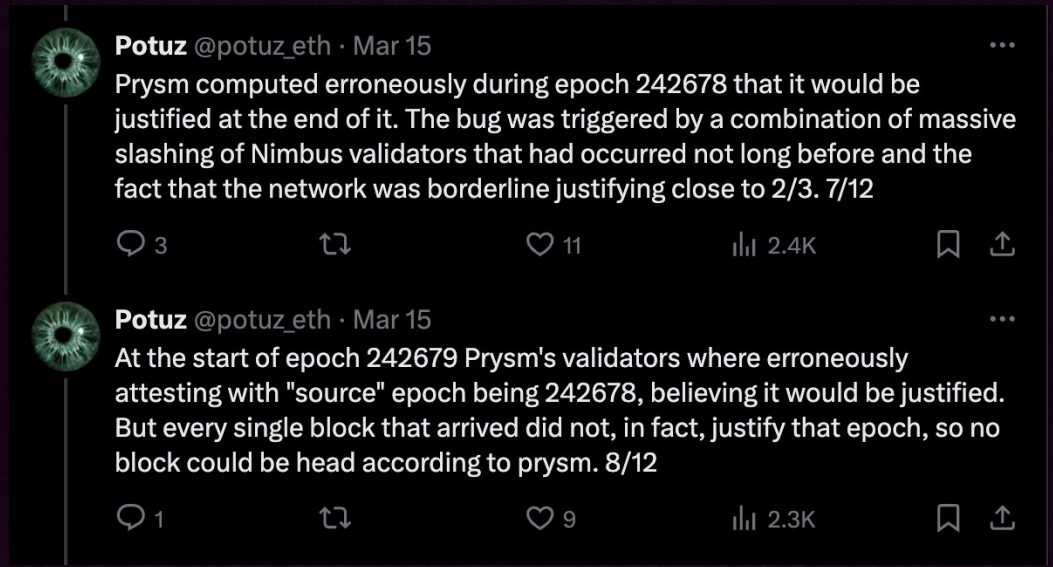
Work queue is clogged
with “garbage”

No blocks / attestations
get produced in time
= drop in participation

Past incidents: Medalla event



Past incidents: Goerli



Potuz @potuz_eth · Mar 15

Prysm computed erroneously during epoch 242678 that it would be justified at the end of it. The bug was triggered by a combination of massive slashing of Nimbus validators that had occurred not long before and the fact that the network was borderline justifying close to 2/3. 7/12

3 11 2.4K

Potuz @potuz_eth · Mar 15

At the start of epoch 242679 Prysm's validators where erroneously attesting with "source" epoch being 242678, believing it would be justified. But every single block that arrived did not, in fact, justify that epoch, so no block could be head according to prysm. 8/12

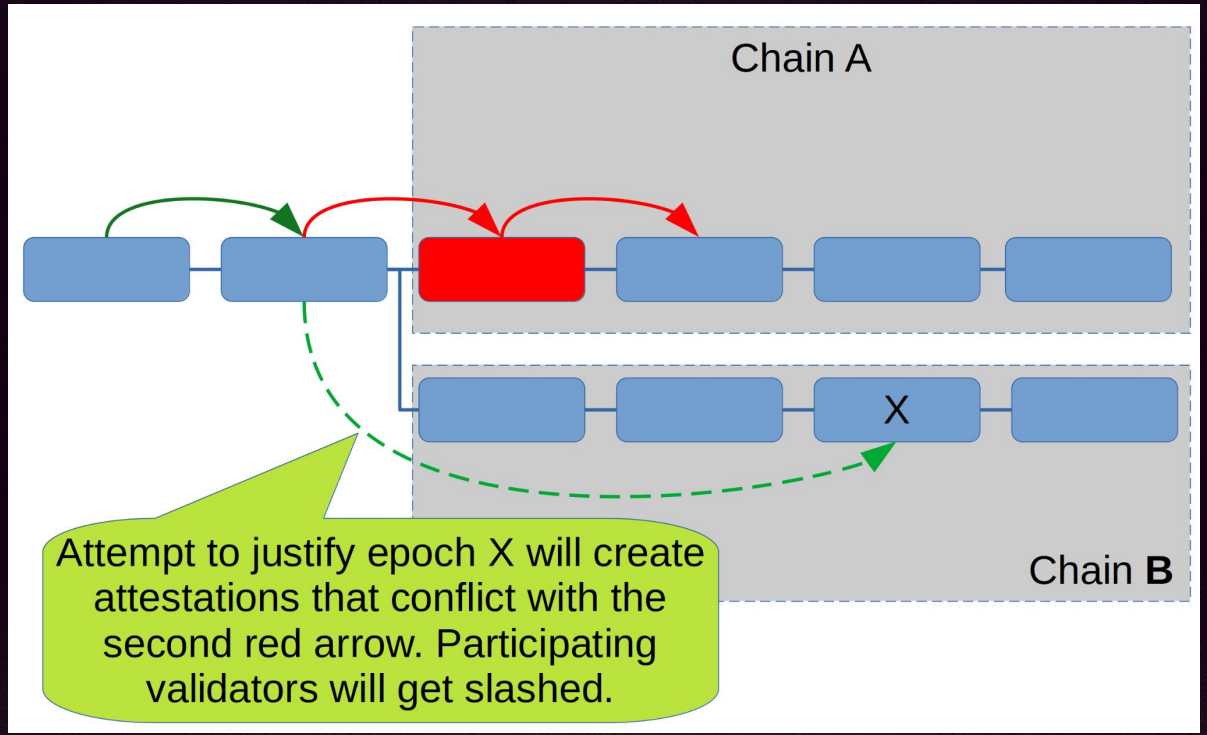
1 9 2.3K

Past incidents: PeerDAS devnet

```
[2024-10-25 01:53:17.16] ERROR blockchain: Could not process slots to get payload
attribute error=could not process slots: could not process deneb epoch: could not
process registry updates: could not get active validator count: could not update
committee cache: context deadline exceeded
[2024-10-25 01:53:17.16]  INFO blockchain: Called fork choice updated with
optimistic block finalizedPayloadBlockHash=0x6250cb5cd6eb
headPayloadBlockHash=0xc8ad3f87a7b9 headSlot=16575
[2024-10-25 01:53:17.17]  INFO blockchain: Chain reorg occurred
commonAncestorRoot=0xd79e57606d6aebe6136d06d9ffe9ab146873cd307f19ee8e7035ef0d3989
6059 depth=131725 distance=131735
newRoot=0x2bb1b092deab8e546551ea8c85dc68cd007fd55ff42435b5a7e3c03c5fb0f639
newSlot=16575 newWeight=0
oldRoot=0xc0ecd36fd4e490877402b490e8b061fb687aefdf6643905a5f3b52e4f83700f7
oldSlot=148290 oldWeight=2160000000000
[2024-10-25 01:53:17.19] ERROR sync: Could not handle p2p pubsub error=could not
update justified checkpoint: context deadline exceeded
topic=/eth2/0d99d9c5/beacon_block/ssz_snappy
```

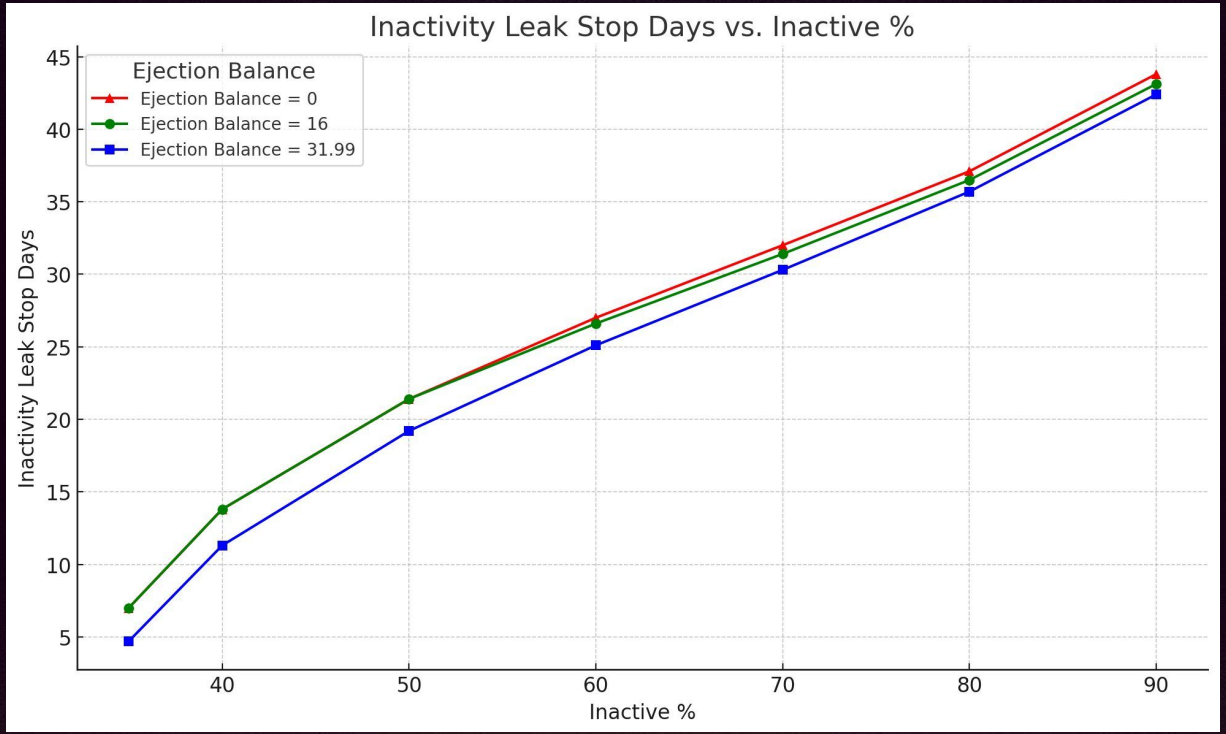
SPIRALOOOOR

Worst case non-finality?



Source: https://dankradfeist.de/ethereum/2022/03/24/run-the-majority-client-at-your-own-peril.html

Worst case non-finality?



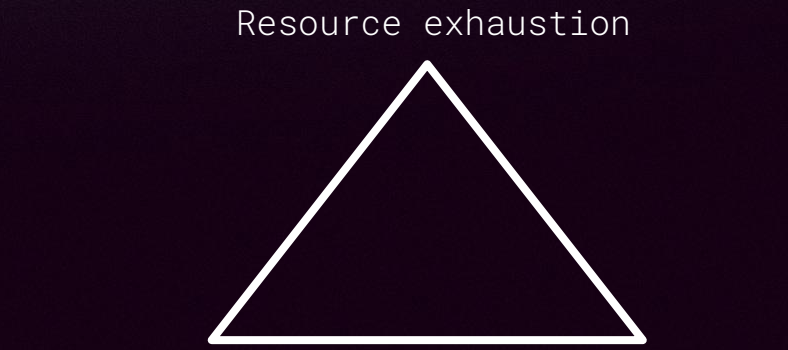
Source: https://hackmd.io/@dapplion/inactivity_leak_maxeb

What to do

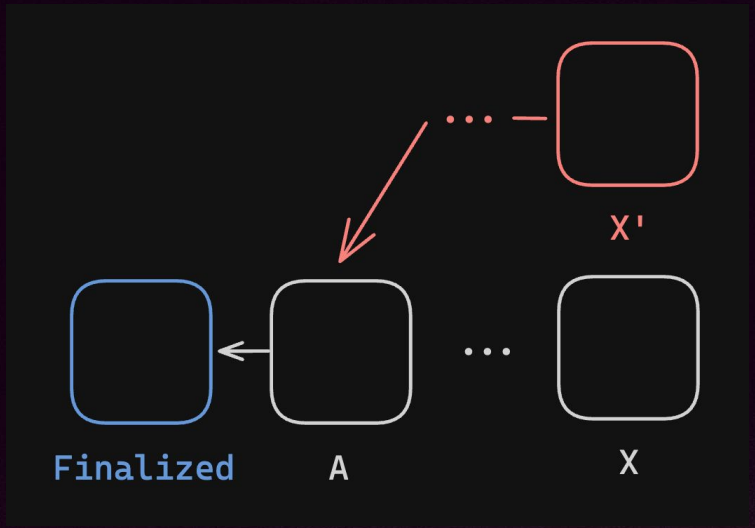
Mitigations

- Don’t release buggy clients
- Don’t run out of resources
  - Disk space
  - Memory
  - CPU time to process canonical things
- Reject expensive useless network objects
  - Attestations
  - Blocks?

Another trilema



Don't run out of disk space



Only store states of epochs with blocks

Only store states of valid blocks

Store only every N epochs

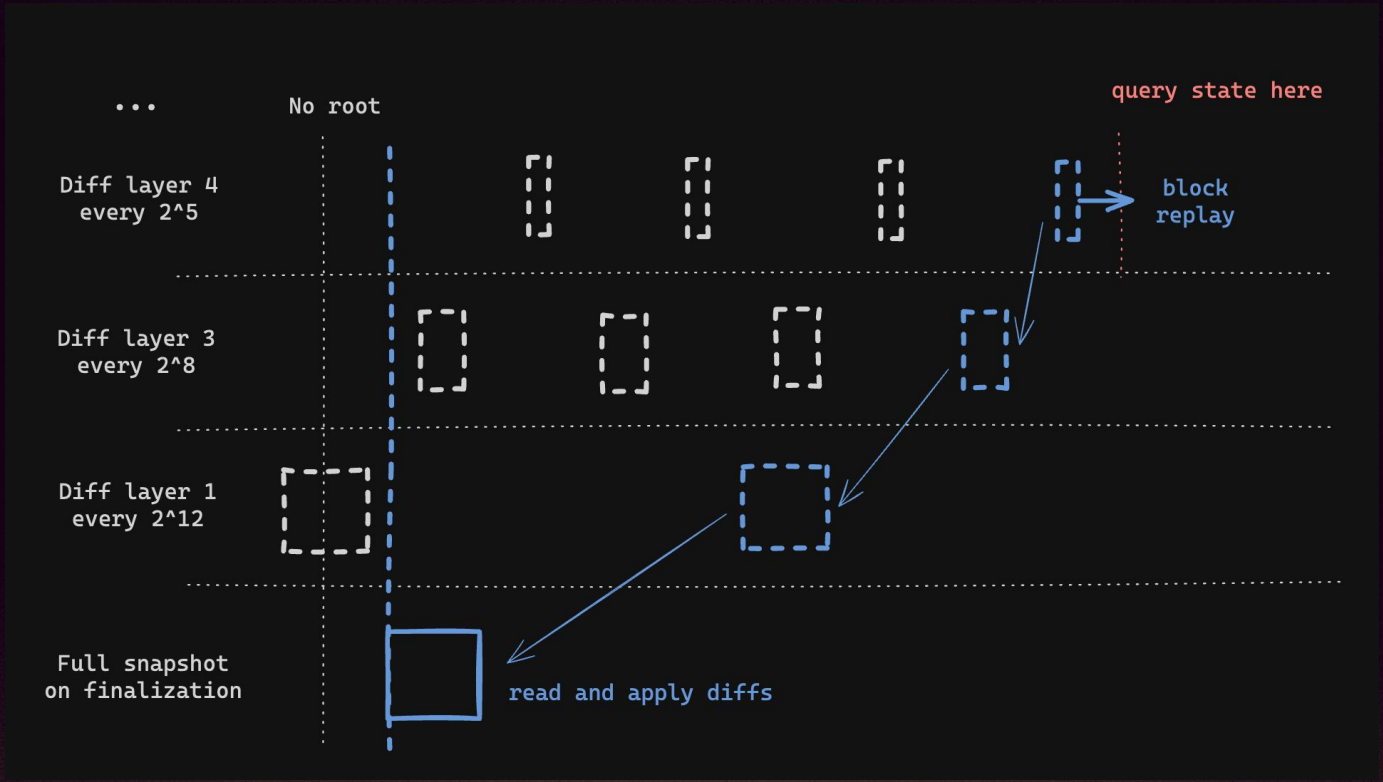
Store states as diffs :)

Unfinalized states diffs

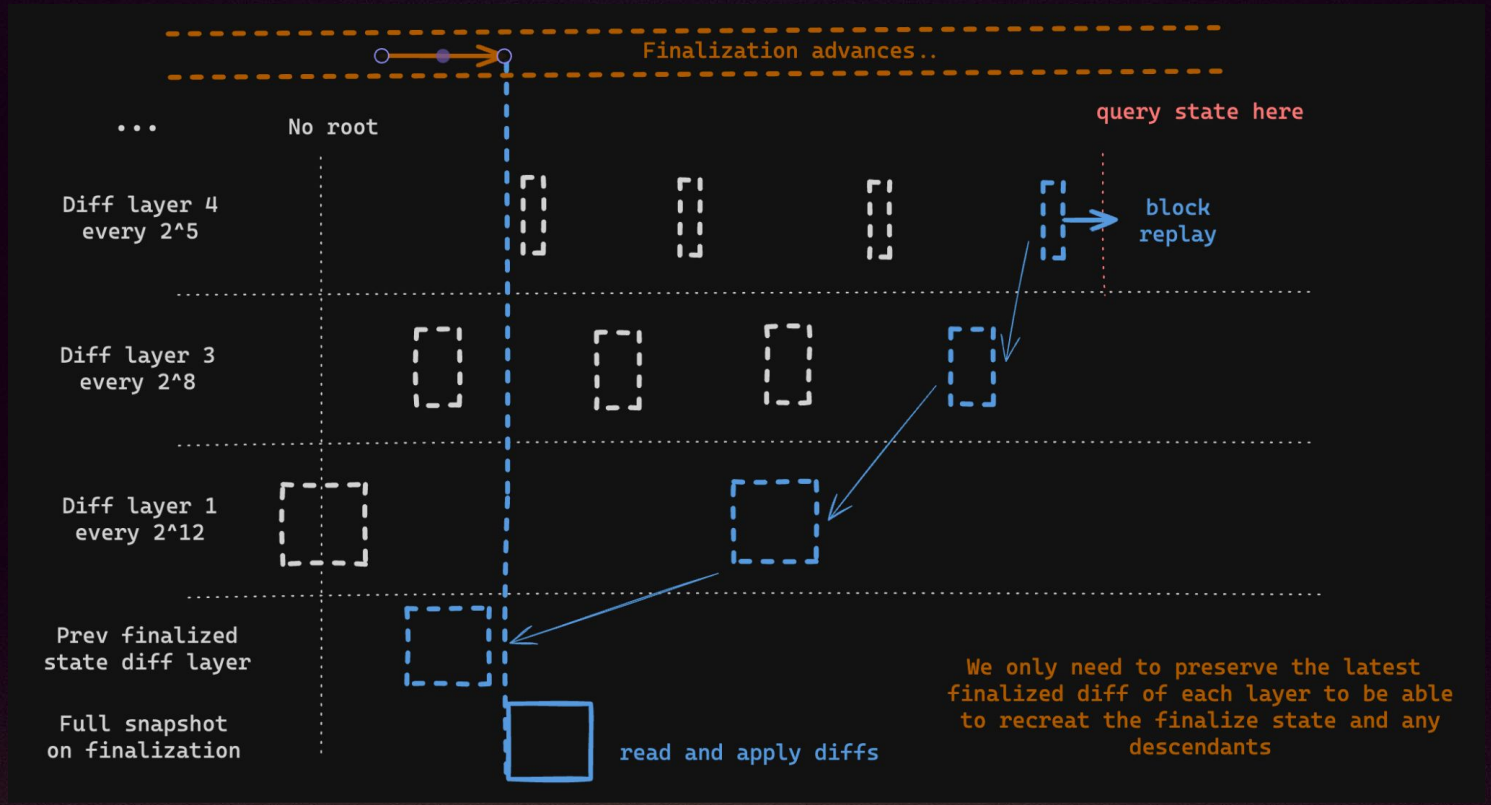
Slot diff	Diff size (bytes)	Compute time	Apply time
64	574,203	56 ms	39 ms
32,000	4,983,576	143 ms	55 ms
320,000	8,499,854	151 ms	57 ms
3,200,000	50,795,068	700 ms	127 ms

Diffs against a recent Mainnet state (~200 MB)

Unfinalized states diffs



Unfinalized states diffs

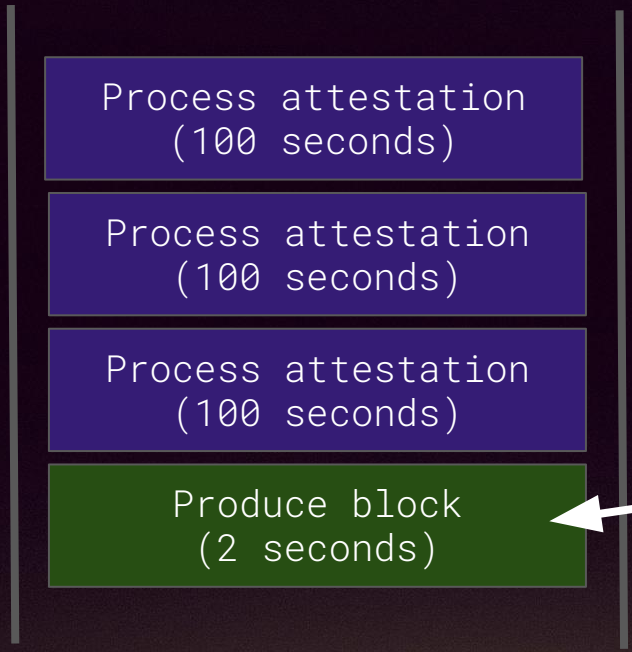


Don't run out of memory

- Tree states :)

Prioritize useful blocks / attestations

Remember the past incident Mainnet 2023



Work queue is clogged
with “garbage”

No blocks / attestations
get produced in time
= drop in participation

Prioritize useful blocks / attestations

Priority 0:
descendants of recent heads, production

Produce block
(2 seconds)

Priority 1:
else

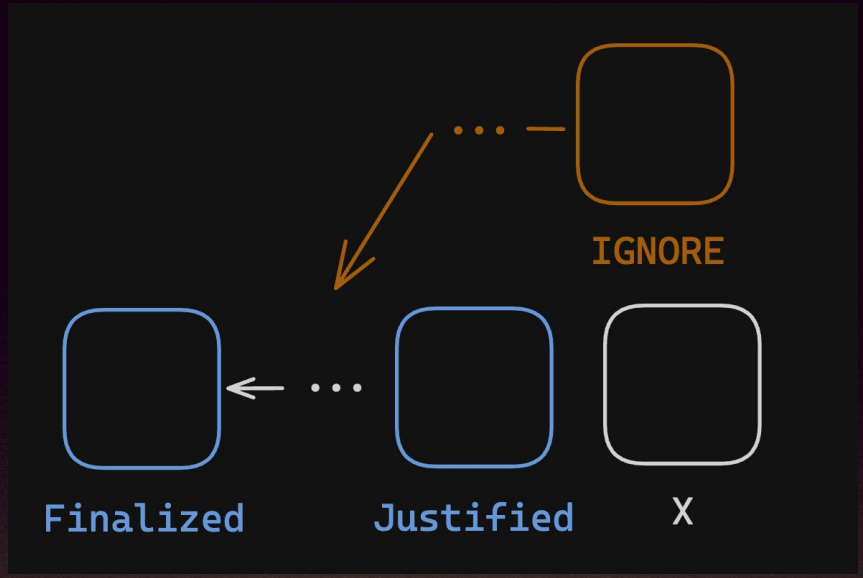
Process attestation
(100 seconds)

Process attestation
(100 seconds)

Process attestation
(100 seconds)

Reject certain blocks / attestations

- Dangerous optimizations that can affect liveness



Next steps

Next steps

Run a cyclic non-finality tests of 1 month

Update and deploy attack tools

Progressively harden against the found failure modes = don’t wait for mainnet to break

h/t ethpanda ops

Non-finality could kill ethereum	
core devs will never let ethereum not finalize	
all core devs are on holiday in Thailand	

99.5%

Average participation in 2024

Thank you!

dapplion

Core dev, Sigma Prime

@dapplion