



# **[EIP-7732] Enshrined Proposer-Builder Separation**

Implementation in Prysm and Nimbus  
Separation of ethereum block into consensus and execution parts

**kira, Caleb, Amaan**



⚠ Draft

Standards Track: Core

## EIP-7732: Enshrined Proposer-Builder Separation

Separates the ethereum block in consensus and execution parts, adds a mechanism for the consensus proposer to choose the execution proposer.

**Authors** Francesco D'Amato <[francesco.damato@ethereum.org](mailto:francesco.damato@ethereum.org)>, Barnabé Monnot <[barnabe.monnot@ethereum.org](mailto:barnabe.monnot@ethereum.org)>, Michael Neuder <[michael.neuder@ethereum.org](mailto:michael.neuder@ethereum.org)>, Potuz (@[potuz](https://twitter.com/potuz)), Terence Tsao <[ttsao@offchainlabs.com](mailto:ttsao@offchainlabs.com)>

**Created** 2024-06-28

**Discussion Link** <https://ethereum-magicians.org/t/eip-7732-enshrined-proposer-builder-separation-epbs/19634>

<https://eips.ethereum.org/EIPS/eip-7732>







# Motivations

- Currently,
  - Beacon block proposers request a hash tree root (HTR) from third-party builders.
  - They submit a ***SignedBlindedBeaconBlock*** to a trusted relay.
  - The relay replaces the HTR with the full execution payload off-protocol before broadcasting.
  - This setup requires trust in intermediaries (relays).
- Project was about solving trust issues and **not about transaction ordering or MEV (Maximum Extractable Value)**.







# Motivations

- Trust-free exchange between proposer and builder without intermediary
- Proposers can outsource block construction without trusting builders, and builders can advertise blocks directly without intermediaries.
- Guarantees
  - Proposer payment safety: if an honest builder's payment is processed, payload becomes canonical
  - Builder payload safety: if an honest builder publishes a payload, it becomes a canonical. (honest majority)







# Core Changes

**Execution Layer:** No changes required.

## **Consensus Layer:**

- Modifications in Beacon Chain and Fork Choice logic.
- Introduction of a new “Payload Timeliness Committee (PTC)” which is renamed to “Availability Committee (AC)” in the new design
- A new role called builder or staked builder

**Engine API:** No changes needed.





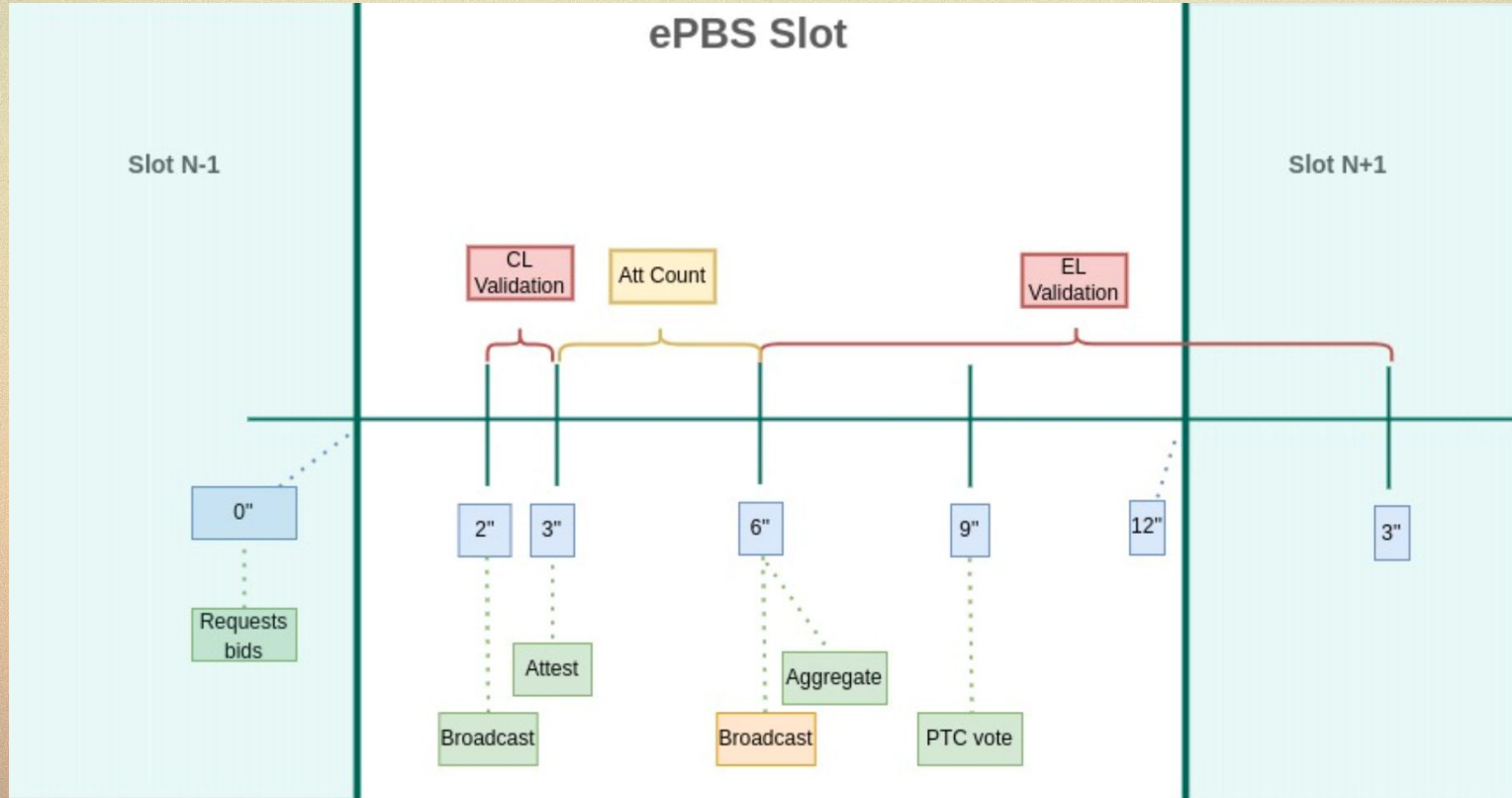


# Core Changes

- Separate consensus and execution validation of block
  - 9s to validate execution instead of 1-2s
  - 12-15s to validate DA instead of 1-2s, more time to produce kzg proofs
- Consensus block no longer carries execution
  - resulting in faster propagation & verification
- Consensus liveness no longer depends on execution results
- Any builder can set the floor of the auction through p2p market space



# Core Changes







# Core Changes

- At any given slot, the blockchain's head can be:
  - Skipped: Block is missing
  - Empty: Block is on-chain, execution payload is missing
  - Full: Both block and execution payload are included on-chain
- Initial Fork choice rule design
  - (block, slot) voting with new payload and withheld boosts





# Prysm implementation highlights

(ss by terence)

- Fork into to ePBS epoch
- Operate under new slot intervals
- Validator self-build and propose consensus block / execution payload
- Get PTC duties and perform PTC duties
- Verify consensus block and execution payload at different cutoff
- Chain finalize under epbs epoch

```
time="2024-11-07 08:17:30" level=debug msg="Synced new block"
block=0xe209004... chainServiceProcessedTime=60.665625ms
dataAvailabilityWaitedTime=0s deposits=0 epoch=6 finalizedEpoch=4
finalizedRoot=0xa332f926... justifiedEpoch=5
justifiedRoot=0x8ae59858... parentRoot=0xe5604191...
prefix=blockchain sinceSlotStartTime=383.192ms slot=36
slotInEpoch=0 version=epbs
```



Potuz @potuz\_eth · 18h

Just in time for devcon. This is a prysm node finalizing on ePBS. A short 📺

```
time="2024-11-07 08:17:30" level=debug msg="Synced new block"
block=0xe209004... chainServiceProcessedTime=60.665625ms
dataAvailabilityWaitedTime=0s deposits=0 epoch=6 finalizedEpoch=4
finalizedRoot=0xa332f926... justifiedEpoch=5
justifiedRoot=0x8ae59858... parentRoot=0xe5604191...
prefix=blockchain sinceSlotStartTime=383.192ms slot=36
slotInEpoch=0 version=epbs
```

0:13



2



10



82



4.4K







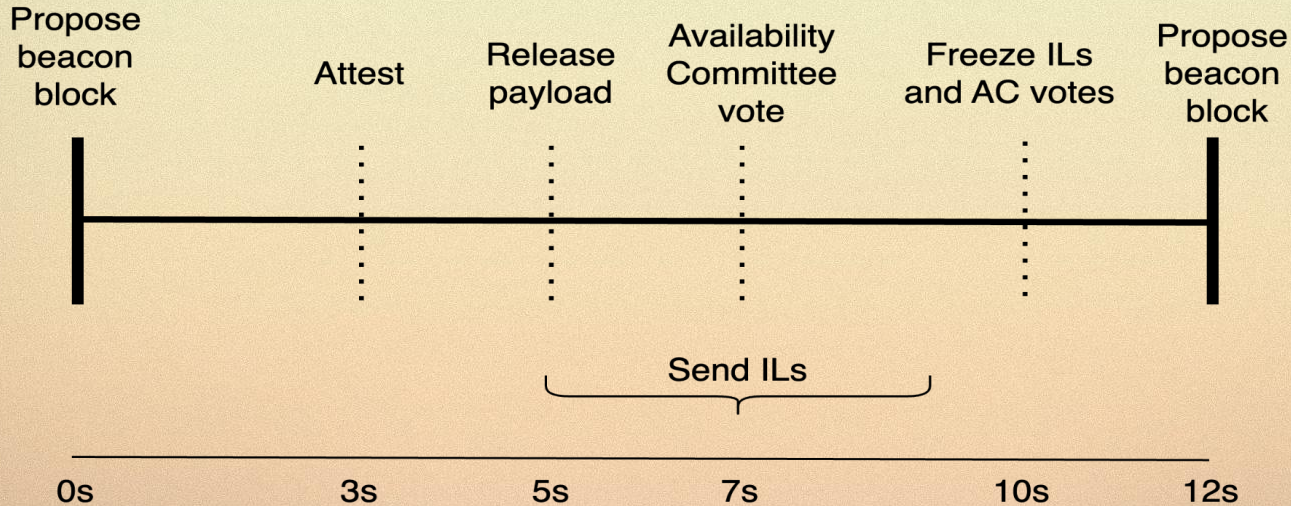
# Future works

- Implement the new fork-choice design
- Hope that more clients start implementation for ePBS (teku has already started it)
- Update fork-choice consensus specs according to new design
- Spec tests
- Devnet testing





# New fork-choice Design



(Francesco D' Amarto, "All-in-one epbs/FOCIL/DAS fork-choice")



# A Special Thank you to:

Mentors:

- potuz
- terence
- tersec

Researchers:

- Barnabé
- francesco

EPF organizers:

- Mario and Josh







**Thank you!**

**EPF Fellows**

(Caleb) [calebomoniyitomiwa@gmail.com](mailto:calebomoniyitomiwa@gmail.com)

(kira) [shyampkira@gmail.com](mailto:shyampkira@gmail.com)