Currently each block includes a Merkle state root, and as it was discussed here, recalculating the state root all the time is computationally intensive.

Lets imagine that you have a really fast PoS or PBFT shard, that does, say, 1000 transactions per second. Then you need to do lots of state root updates. This really slows down the system.

A question is then, why not to calculate the state root, say, each 60 minutes.

1. Only once in 60 minutes a block would include a state root.
2. A light client that wants to confirm the value of a particular variable X faster than in 60 minutes could randomly pick, say, 20 nodes from the network and ask them for the state of X.
3. If ALL of the nodes would report the same value, the client would accept this value.
4. If there would be at least one dissenting node, the client would submit a request to a much larger set of the nodes (say 100 nodes) and then accept the majority value
5. After one hour, once the state root appears on the chain, the deposits of the nodes that reported an incorrect value would be slashed, and the client would get a bounty for reporting bad guys.