Based on some explanations given here: Difference between Stateless Ethereum and ReGenesis regarding transaction pool I will propose a hypothetical architecture of Ethereum node that decouples transaction pool handing from most of other things that happen in it. This decoupling means that we could create a special type of Ethereum nodes, called “Transaction Pool nodes”, and their task would be exclusively to receive, verify, and propagate, transactions around the network. One of the main reasons to propose this is that if transaction pool nodes can be separated, engineering work on them can be performed by people specialising in this subject area, and they can add functionality and optimise within the confines of the protocol, and later on, suggest and implement improvements to the protocol.

The main idea is to make the merkle proofs of sender

accounts (which includes balance and nonce, those are required for basic anti-spam measures) mandatory

in the transactions. This will make transactions larger (3k more if we do not switch state to binary merkle tree, and 1k more if we do switch state to binary merkle tree), and it will also make transactions harder to produce. The question is - will this be an acceptable tradeoff?

If this is done, it will be just the first step towards gradually shifting the burden of maintaining the state from the core of the network (relaying nodes and mining nodes) to the perimeter (nodes that create and inject transactions). I believe that it will make the whole system more incentive compatible with further growth.