

I have been thinking a good bit about log accumulators and stateless transactions as I've discussed in [Roast this Concept: One big off chain shard](#) and [State minimized implementation on current evm](#). Here is a schematic for how the collations will become eventually consistent.

Collators run transactions and produce log accumulations of state transitions. They pick transactions based on gas price offered

Instead of having just one collator, we allow competition. Any staked collator that falls into a random range can start one. Only one will win, and the others that start one can still create uncles. The collations are verified by other random verifiers that take the collators op list and verify the same root through a hide-and-reveal scheme.

At the end of each accumulation, we have one winner and a bunch of uncles. The collators can re-sign their collation after checking state changes and have those signed items(that have already been verified) easily included in the next accumulation. The next collator will include them because they have already been verified and the uncle collator will be on the stake hood.

This allows transactions including a large amount of data collection to eventually get included even if they aren't included right away.

Uncle producers get some kind of reward but must share some of it with the eventual includer.

[

Eventual%20Collation

1931×1160 110 KB

](<https://ethresear.ch/uploads/default/original/2X/e/ed268adaa2eb73a7f784e11e3eb0c236915af7b6.png>)