One of the processes that need to be done in a sharded chain of the type we are researching is cross-linking the shard chain back into the beacon chain (see here: Cross-links between main chain and shards). A randomly sampled committee in the beacon chain needs to approve a block in the sharded chain, for every shard, once in a while (think: every hour or so).

There are two ways I can think of to accomplish this:

On-chain aggregation

The beacon state keeps track of “the committee” for every shard; these are nodes assigned to watch that shard for the very short term (think an hour at a time). During every epoch, every validator can make a vote once (same as FFG). This vote always contains values related to FFG voting, but for any committee that that particular validator was assigned to, in that vote they will also vote on a cross-link for that committee.

Pros:

- Every validator gets a vote in every epoch, incentivization simple

Cons:

- Requires two separate votes for cross-links vs CASes
- Not necessarily clear which block to make a CAS for

Off-chain aggregation

Every block in the beacon chain can include all of the signatures for one particular CAS, from the previous period. The block creator can choose which one, but is incentivized to choose (i) CASes from shards that have not been cross-linked recently, and (ii) 0-skip (ie. full size) CASes.

Pros:

- No need for on-beacon-chain logic keeping track of how many votes from some CAS have already been included
- No need for on-beacon-chain logic trying to coordinate which CAS to sign

Cons:

- Large pools can prefer CASes that contain more of their own signatures, thereby unfairly privileging themselves.
- Not every validator will be able to vote in every epoch, weakening the FFG cycle
- Particularly during >33% offline situations, it’s important for a validator to be able to constantly signal to the system that they are participating.

There is also a version where the beacon block creator has no choice which shard and which height to include a CAS from; he is told which shard and height, and can either include it or not. However, this gives a single actor a lot of power to prevent one shard from being cross-linked for a significant length of time.

Mitigations

We can try to mitigate the consequences of the above approaches. Here are a few ideas:

- Have a fairly simple mechanism for specifying what block validators should be trying to make a cross-link for, eg. “the block made during period N-1”. However, this then establishes the creator of that block as a monopoly actor that can prevent a cross-link by not showing up.
- A beacon block proposer including signatures from a CAS would have the option to include additional signatures from the same committee, so there would then be two parties who can include signatures (thanks @JustinDrake!)
- Not just that beacon block proposer, but also the next k proposers, can include additional signatures from the same committee.
- The first beacon block proposer can only make a cross-link at a particular height; if they do not, the next proposer can make a cross-link at the next height. In this way, we avoid the problem of both monopoly CAS aggregators (ie. shard proposers) and monopoly beacon proposers.