Assuming there are contracts hosted on the BeaconChain, there should be some function to yank contracts from various shards and add them to the BeaconChain.

We define a bounded set A

that contains contracts. A

is bounded in order to restrict it to those contracts deemed as popular by various validators and to create some competition between contracts.

Assuming contract C

is not in A

and receives more transactions than any contract in A

, a validator would create a proposal to add this contract to set A

. Once the set has been filled the contract with the least transactions to it would be replaced by C

.

The reason behind this is to move the most popular contracts into a highly available global state. This would reduce latency between cross-shard transactions with popular contracts.

Loredana wrote up a similar proposal using a master shard:https://medium.com/@loredana.cirstea/a-master-shard-to-account-for-ethereum-2-0-global-scope-c9b475415fa3, however this did not contain yanking logic.