Sharding

The main goal of sharding is to overcome scalability issues. Without sharding, every node in the network has to process all transactions and execute all smart contracts. Sharding parallelizes execution, by dividing the network into smaller components or shards. Nodes are then assigned to different shards with separate account balances.

Each shard essentially corresponds to a separate blockchain, that can be run almost independently of the other shards. This means that transactions on one shard are only processed by the nodes on that shard, allowing more transactions to be processed overall.

Sharding architecture

Concordium's blockchain has a two-level sharded design with a robust control chain and light-weight shards. The control chain manages shards, provides a finalisation service to the shards and provides a vehicle for cross-shard transactions. Each shard runs an individual blockchain and uses the control chain to coordinate the individual shards.

A shard defines an ordering of all transactions within the shard. As the control chain provides ordering/synchronisation across all shards, the entirety of shards can be considered a single totally ordered blockchain. As shards allow for efficient reading of only the parts of this global ledger needed for a given application, this architecture makes it possible to create shards for specific purposes, including private shards as described below.

Obtaining security and efficiency

For optimal efficiency, there should be many shards. Initially, Concordium runs an optimistic consensus algoríthm on the shards, but shards may use different algorithms. Concorium will support more consensus algorithms to meet the specific requirements on a shard. In order to optimise the use of the resources on the blockchain, a shard is run by a small committee. However, the risk associated with sampling only a small number of nodes per shard is that a large fraction of them are corrupted. Normally, for a consensus protocol, as run by the shard committee, to be secure, only a small fraction of participants can be corrupted.

The important insight allowing us to circumvent this issue is that security consists of two parts: safety and liveness. Safety means the system does not make mistakes. Liveness means the system does not come to a halt. These two properties can be balanced so that the level of corruption one can tolerate is different for both.

This means one can set the parameters of the (shard consensus) protocols such that safety holds, i.e., that the system makes no mistakes, even with high corruption levels, whereas the system may come to halt if only a few nodes are corrupted.

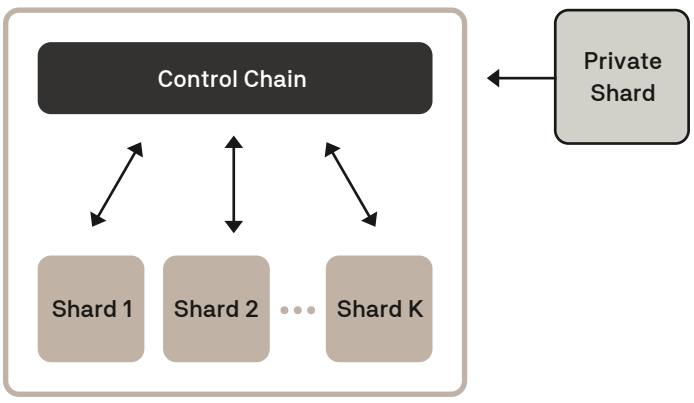
We will utilize this insight as follows. The control chain consists of many nodes with broad decentralization. The shards run with few nodes per shard, tolerating relatively high corruption for safety. This gives us scalability and safety but with limited liveness in the shard. To improve the latter, the control chain monitors the shards and if any come to a halt because of too many corrupted nodes, it resamples the set of nodes running that shard. This re-establishes liveness and with safety and liveness in place, we have security.

Intershard signaling

Intershard signaling allows transactions between shards and communication of smart contracts on different shards. Our protocol for this operates as follows: When a block finalizes on a shard, it contains a list of outgoing messages for other shards. The nodes of the sending shard sign the list of outgoing messages. The nodes in the receiving shard can obtain the list of nodes running on the sending shard together with their public keys from the control chain, which allows them to verify the signed messages from the sending shard. Once a message is verified, it is executed on the receiving shard.

Private shards

The sharding mechanism also allows private shards. In a private shard the control chain cannot see the transactions on the shard, it only provides Finality as a Service (FaaS) and coordination to relaunch deadlocked shards. The private shard can ultimately run its own consensus algorithm and use its own identity providers and anonymity revokers. Private shards provide a cheap way for an individual, country or corporation to launch their own blockchain while having the benefit of the strong finalisation service provided by the control chain.



Concordium Blockchain