Parent-child interactions

The interaction between two subnets in a parent-child relation, is the basic building block of the recursive IPC hierarchy.

There are following parent-child interactions available in IPC

1. Creating child subnets in the IPC hierarchy.
2. Depositing funds from an account in a subnet to an account in its child.
3. Withdrawing funds from an account in a subnet to an account in its parent.
4. Checkpointing a subnet's replicated state in the replicated state of its parent.
5. Invoking actor functions across subnets.
6. Removing child subnets from the IPC hierarchy.

Checkpointing

Checkpointing is a method for a parent subnet to keep a record of the evolution of its child subnet's state by including snapshots of the child's state (called checkpoints) in the parent's state. If, for some reason, the child subnet misbehaves as a whole, agreement can be reached in the parent subnet about how to proceed.

Checkpointed history of a child subnet cannot be reverted as long as a parent subnet operates as expected. Checkpoints are propagated in a recursive way all the way to the rootnet (L1), which makes child subnets benefit from security of their ancestor subnets.

In case of subnet failure, checkpointing enables participants (e.g., former users of the failed subnet) to agree on picking up an older version of the child subnet's state from before the occurrence of the failure and, say, use that version as the initial state of a new, more robust subnet.

Checkpointing fees

There are a number of fees that are paid during checkpointing:

- When a subnet checkpoints its state to a parent, this is the equivalent of a transaction on the parent. The usual transaction fees of the parent are paid to accomplish this.
- In order for a subnet to be consideredanchored
- to the parent, relayers must have sufficient funds in their respective wallets in the parent to be able to pay for a checkpointed transaction.
- When a cross-net transaction is included in a subnet's checkpoint to a parent, the fees for that transaction are distributed as a reward equally among all the relayers that have submitted an instance of that checkpoint.
- Relayers are allowed to submit a checkpoint and eligible for rewards from the commitment of the first checkpoint in, e.g. epoch h, to the first submission of a checkpoint of epoch$h+1$. From this point on, no new valid submissions for checkpoint h will be accepted.
-

Parent finality

Parent finality is a mechanism for proving that a subnet irreversibly reached a certain state.

It is achieved in the following way

- Validators in the child subnet periodically listen to new blocks from the parent. On the implementation level it is performed by Fendermint node of subnet validator subscribing to events via parent's ETH RPC.
- As part of the consensus algorithm in the child, the leader of the consensus proposes the height and hash of the parent's block that they currently consider final.
- The rest of the validators agree or reject this finality from the parent as part of the process of voting the validity of a block (i.e. validators implicitly agree on the finality of the parent through the block validation).
- As part of the execution of the block, validators implicitly commit the finality seen in the parent. This triggers the execution of all top-down messages from the latest finality to the one committed, as well as any changes on the validator set or collateral that may need to be propagated down.
- When a user performs a top-down message, it is added to a queue in the parent chain. Top-down messages are indexed in the parent by the height where they were committed. This index is used by child subnet validators to determine when a top-down message commitment can be considered as final and their execution can be triggered in the child (when their corresponding finality has been triggered).
-

Previous Subnets Next Circulating supply Last updated26 days ago On this page *Checkpointing * Checkpointing fees *

Parent finality