StakeIBC

Suggest Edits

The StakeIBC Module

The StakeIBC Module contains Stride's main app logic:

- it exposes core liquid staking entry points to the user (liquid staking and redeeming)
- it executes automated beginBlocker and endBlocker logic to stake funds on relevant host zones using Interchain Accounts
- it handles registering new host zones and adjusting host zone validator sets and weights
- it defines Stride's core data structures (e.g. hostZone)
- it defines all the callbacks used when issuing Interchain Account logic

Nearly all of Stride's functionality is built using interchain accounts (ICAs), which are a new functionality in Cosmos, and a critical component of IBC. ICAs allow accounts on Zone A to be controlled by Zone B. ICAs communicate with one another using Interchain Queries (ICQs), which involve Zone A querying Zone B for relevant information.

Two Zones communicate via a connection and channel. All communications between the Controller Zone (the chain that is querying) and the Host Zone (the chain that is being queried) is done through a dedicated IBC channel between the two chains, which is opened the first time the two chains interact.

For context, ICS standards define that each channel is associated with a particular connection, and a connection may have any number of associated channels.

Params

DepositInterval (default uint64 = 1) DelegateInterval (default uint64 = 1) ReinvestInterval (default uint64 = 1) RewardsInterval (default uint64 = 1) RedemptionRateInterval (default uint64 = 1) StrideCommission (default uint64 = 10) ICATimeoutNanos(default uint64 = 600000000000) BufferSize (default uint64 = 5) IbcTimeoutBlocks (default uint64 = 300) FeeTransferTimeoutNanos (default uint64 = 1800000000000) DefaultMinRedemptionRateThreshold (default uint64 = 90) DefaultMaxRedemptionRateThreshold (default uint64 = 150) MaxStakeICACallsPerEpoch (default uint64 = 100) IBCTransferTimeoutNanos (default uint64 = 1800000000000) MinRedemptionRates (default uint64 = 90) MaxRedemptionRates (default uint64 = 150) ValidatorSlashQueryThreshold (default uint64 = 1)

Keeper functions

- LiquidStake()
- RedeemStake()
- ClaimUndelegatedTokens()
- RebalanceValidators()
- AddValidators()
- ChangeValidatorWeight()
- DeleteValidator()
- RegisterHostZone()
- ClearBalance()
- RestoreInterchainAccount()
- UpdateValidatorSharesExchRate()

State

Callbacks

- SplitDelegation
- DelegateCallback
- ClaimCallback
- ReinvestCallback
- UndelegateCallback
- RedemptionCallback
- Rebalancing
- RebalanceCallback

HostZone

- HostZone

- ICAAccount
- MinValidatorRequirements

Host Zone Validators

- Validator
- ValidatorExchangeRate

Misc

- GenesisState
- EpochTracker
- Delegation

Governance

- AddValidatorsProposal

Queries

- QueryInterchainAccountFromAddress
- QueryParams
- QueryGetValidators
- QueryGetHostZone
- QueryAllHostZone
- QueryModuleAddress
- QueryGetEpochTracker
- QueryAllEpochTracker
- QueryGetNextPacketSequence

Events

stakeibc module emits the following events:

Type: Attribute Key → Attribute Value

registerHostZone: module → stakeibc registerHostZone: connectionId → connectionId registerHostZone: chainId → chainId submitHostZoneUnbonding: hostZone → chainId submitHostZoneUnbonding: newAmountUnbonding → totalAmtToUnbond stakeExistingDepositsOnHostZone: hostZone → chainId stakeExistingDepositsOnHostZone: newAmountStaked → amount onAckPacket (IBC): module → moduleName onAckPacket (IBC): ack → ackInfo Updated7 months ago

Module Overviews Records Did this page help you?Yes No *Table of Contents * *The StakeIBC Module * * *Params * * *Keeper functions * * *State * * *Queries * * *Events * * *Type: Attribute Key → Attribute Value