Stride's Technical Architecture

A technical overview of how Stride enabled interchain liquid staking.Suggest Edits

High-Level System Design

Users stake their tokens on Stride from any Cosmos chain. Rewards accumulate in real time. No minimum. They will receive staked tokens immediately when they liquid stake. These staked tokens can be freely traded, and can be redeemed with Stride at any time to receive your original tokens plus staking rewards.

On the backend, Stride permissionlessly stakes these tokens on the host chain and compounds user rewards. Stride lets users use your staked tokens to compound their yields. Continue to earn staking yield, and earn additional yield by lending, LPing, and more. They can set their own risk tolerance in Cosmos DeFi.

Users can always redeem from Stride. When they select "redeem" on the Stride website, Stride will initiate unbonding on the host zone. Once the unbonding period elapses, the users will receive native tokens in their wallets.

In-Depth System Design

*Written by the wonderfulInformal Systems audit team.

In Stride, staking occurs once per "stride epoch" (every 6 hours). The process spans over 3 "stride epochs":

- stride epoch n
- : New deposit record (DR) which tracks all deposits in a given epoch for a given host zone is created with 0 tokens. LiquidStake is called then and stTokens are minted, but the actual staking of the user’s tokens is addressed in epoch n+2.
- stride epoch n+1
- : All tokens on DR from epoch n are IBC transferred from Stride’s module account to Delegation ICA. Whenever a change to delegation happens all of the rewards are withdrawn (to Withdraw ICA). • epoch n+2: Tokens on the DR are staked (by weight) across all (30 at the moment) Stride validators.

Reinvestment executes automatically and the rewards are auto-compounded on every epoch (6h). 90% of the rewards are being sent to Delegation ICA (reinvestment) and 10% to the Fee ICA (the only place where Stride charges the fees)

- stride epoch n
- : Queries Interchain Query (ICQ) to check balances of Withdraw ICA and creates a new record for those tokens.
- stride epoch n+1
- : Transfers tokens to Delegation ICA from the Withdraw ICA.
- stride epoch n+2
- : Stakes the tokens.

Per an SDK constraint, we can issue no more than 7 undelegation messages in a given unbonding period. The unbonding period dictates the cadence (in number of days) with which we submit undelegation messages, such that the 7 messages are spaced out throughout the period. We calculate this by dividing the period by 7 and then adding 1 as a buffer. For example: If our unbonding period is 21 days (as is the case for Cosmos Hub), we issue an undelegation every 4th day for a delegator and validator pair.

- daily
- : EpochUnbondingRecord is created. It stores many HostZoneUnbondings (HZU), with one HZU per host zone (e.g. for CosmosHub we have 1 HZU per epoch). When the user sends 1stAtom to Stride (Stride now custodies this 1 stAtom) and specifies an address on the Cosmos Hub that the tokens should be sent, HZU is updated and UserRedemptionRecord (URR) is created (claim on user’s tokens that they can trigger later once the tokens have unbonded).
- stride epoch n + (unbondingPeriodOnHostZone / 7 + 1)
- (m = unbondingPeriodOnHostZone / 7 + 1): For CosmosHub it happens every 4 days (unbPeriod = 21 days). MsgUndelegate is triggered and all of the pulled unbonding tokens are undelegated (MsgUndelegate ICAs are triggered across the validators Stride has delegated to). HZUs are updated with unbonding time.
- At the time the unbonding is completed
- : Tokens are transferred to Redemption ICA account. The URR is updated so that the tokens are claimable and anyone can transfer tokens to the already specified address which is stored on URR. So this is an ICA call that transfers tokens back to the end user’s account.

Deposit & Liquid Staking

Withdrawal and deposit belong to regular bank transfers (outside of Stride). After transferring native tokens to Stride, liquid staking can be processed and it includes:

1. Sending IBC/Tokens to stakeibc account
2. Minting stTokens to stakeibc account
3. Sending stTokens from stakeibc account to user account on Stride

Staking & Reinvesment

Staking and reinvestment steps:

1. Sweeps the deposit record (DR) marked
2. TRANSFER_QUEUE
3. from previous epochs. Under the hub, it con structs IBC
4. MsgTransfer
5. with 30min timeout.
6. TransferCallback
7. is also created which is been called
8. OnAcknowledgementPacket
9. or
10. OnTimeoutPacket
11. . In the case of nill ack or ack_error DR’s status is set back to
12. TRANSFER_QUEUE
13. otherwise it becomes a candidate for delegation with
14. DELEGATION_QUEUE
15. flag.
16. Delegates DRs with status
17. DELEGATION_QUEUE
18. . It creates a set of
19. MsgDelegate
20. msgs (delegation to every validator from that host zone whose relative amount is positive). Each validator gets
21. targetAmount=valWeight*depRecordAmount / totalValWeight
22. . Also,
23. DelegateCallback
24. is defined. In the case of the happy ibc path, the zone’s staked balance is increased by a delegated amount to each validator whose
25. delegationAmt
26. is updated and finally DR is removed.
27. Rewards are automatically sent to the Withdrawal ICA.
28. WithdrawalBalanceCallback
29. executes ICA
30. SendTx
31. with
32. MsgSend
33. to Delegation ICA (90% reward) and Fee ICA (10% reward). It also adds
34. ReinvestCallback
35. which triggers as previous ones, and in the happy path, it creates a new DR with
36. DELEGATION_QUEUE
37. status (using
38. WITHDRAWAL_ICA
39. source this time, not STRIDE source) while the sad path is ignored.

Unbonding

Unbonding goes as follows:

1. User sends
2. RedeemStake
3. msg which creates
4. UserRedemptionRecord
5. (URR), calculates the amount of native tokens to get back:
6. nativeAmount=stAmount*redemptionRate
7. , updates HostZoneUnbonding (HZU) record and sends numStAtoms to module’s account where they will eventually be burned after unbonding.
8. 7 times per unbonding period, Stride initiates unbondings for all host zones with ICA
9. SendTx
10. containing
11. MsgUndelegate
12. (from Delegation ICA to all validators) and sets HZU status to
13. UNBONDING_IN_PROGRESS
14. (note: HZU is created at
15. RegisterHostZone

16. with initial status set to
17. UNBONDING_QUEUE
18.). If ICA SendTx fails,
19. UndelegateCallback
20. will roll back the HZU status to
21. UNBONDING_QUEUE
22. , otherwise it burns escrowed stTokens, updates: HZU status to
23. EXIT_TRANSFER_QUEUE
24. , HostZone staked balance and validator amounts.
25. The funds are in escrow for the unbonding period and then get automatically transferred to back to the Delegation ICA.
26. Once per day, if the unbonding period has elapsed, all HZU's tokens with
27. EXIT_TRANSFER_QUEUE
28. are swept as a batch to Redemption ICA. If everything goes well, HZU status is set to
29. CLAIMABLE
30. , otherwise it's reset to
31. EXIT_TRANSFER_QUEUE
32. during
33. RedemptionCallback
34. execution.
35. Users can claim their unbonded tokes via
36. ClaimUndelegatedToken
37. which also sets claim pending to TRUE (to avoid double claims). ICA
38. SendTx
39. is constructed with
40. MsgSend
41. inside (from Redemption ICA to user address on host) with 10min timeout. If no ack errors, URR is removed and HZU's native token amount is decremented by the claimed value. Else, if timeot or ack failure, claim pending is set to false.

Interchain Accounts

It can be seen that many paths lead to ICA's SendTx , including both staking and unstaking, while all paths (except direct Msgs) have the same root - the BeforeEpochStart function.

There are only two scenarios for ICA's RegisterInterchainAccount to be called. The first one is when registering the host zone, 4 ICAs are created. The second one, in the case of timeout, if a channel closes, the controller chain must be able to regain access to registered interchain accounts by simply opening a new channel which is done through RestoreInterchainAccount . IBCTransfer is called at one place only from record's module when sweeping existing deposits from Stride to the Host Zones. Updated2 months ago

Stride: Liquid staking for Cosmos blockchains Getting Involved Did this page help you?Yes No *Table of Contents * *High-Level System Design * *In-Depth System Design * * *Deposit & Liquid Staking * * *Staking & Reinvesment * * *Unbonding * * * Interchain Accounts