

We propose to add transferShares

function into the stETH

contract which accepts shares amount as input and performs shares movement from one account to another. We also propose to add event TransferShares

which must be emitted along with Transfer

event. We propose that calling transferShares(recipient, sharesAmt)

should lead to exactly the same outcome as calling transfer(recipient, sharesAmt * sharePrice)

. Including that both transferShares

and transfer

should emit TransferShares

and Transfer

events. The only difference being the measurement unit of the input amount.

Full specification and pull request with contracts changes can be found in [lido-improvement-proposals repository](#).

Motivation

stETH

is a [rebasing token](#). Usually, we transfer stETH

using ERC-20 transfer

and transferFrom

functions which accept as input amount of stETH

, not the amount of the underlying shares.

Sometimes we'd better operate with shares directly to avoid possible rounding issues (the first clear example is [LIP-6 proposal](#), see recoverExcessStETH

). Rounding issues usually could appear after a token rebase.

Backward compatibility

We preserve the existing Transfer

event signature cause it's defined by ERC-20. That's why we introduce the new TransferShares

event instead of adding another arg for the existing Transfer

event.

Gas price effects

Emitting the TransferShares

event costs approximately 1900 additional gas, depending on the execution context.

Transfer of stETH

An addition of emitting the TransferShares

event increases the cost of every stETH transfer (call to Lido contract's transfer(...)

) by ~3.6% (1967 gas).

Submitting ETH to Lido contract

Gas price of submitting ETH for minting stETH is also affected because on minting we need to emit `TransferShares(address(0), ...)`

as well as `Transfer(address(0), ...)`

event.

Slight 1891 gas (~2.2%) increase in costs for every call to the Lido contract's `submit(...)`

function.

Handling LidoOracle report

Call to LidoOracle's `reportBeacon`

which leads to consensus will emit $2 + N$

`TransferShares`

events, where N

is the amount of Node Operators stored in the registry (14 at the moment). Thus, cost of calling `reportBeacon`

will increase by ~30000 gas which is ~6%. The cost is paid approximately every day on a beacon chain state report by the Oracle who reports by calling `reportBeacon`

“the last”. It happens to be [0x007DE4a5F7bc37E2F26c0cb2E8A95006EE9B89b5](https://etherscan.io/address/0x007DE4a5F7bc37E2F26c0cb2E8A95006EE9B89b5) these days.

References

- [full LIP-11 specification](#)
- [proposed changes to Lido contracts](#)
- [In-protocol coverage application mechanism proposal \(LIP-6\)](#), see `recoverExcessStETH`

)

- [ERC-20 token standard](#)
- [About rebasing token](#)