Obol Splits

Obol develops and maintains a suite of smart contracts for use with Distributed Validators. These contracts include:

- Withdrawal Recipients: Contracts used for a validator's withdrawal address.
- Split contracts: Contracts to split ether across multiple entities. Developed by Splits.org
- Split controllers: Contracts that can mutate a splitter's configuration.

Two key goals of validator reward management are:

1. To be able to differentiate reward ether from principal ether such that node operators can be paid a percentage thereward
2. they accrue for the principal provider rather than a percentage ofprincipal+reward
3. .
4. To be able to withdraw the rewards in an ongoing manner without exiting the validator.

Without access to the consensus layer state in the EVM to check a validator's status or balance, and due to the incoming ether being from an irregular state transition, neither of these requirements are easily satisfiable.

The following sections outline different contracts that can be composed to form a solution for one or both goals.

Withdrawal Recipients

Validators have two streams of revenue, the consensus layer rewards and the execution layer rewards. Withdrawal Recipients focus on the former, receiving the balance skimming from a validator with >32 ether in an ongoing manner, and receiving the principal of the validator upon exit.

Optimistic Withdrawal Recipient

This is the primary withdrawal recipient Obol uses, as it allows for the separation of reward from principal, as well as permitting the ongoing withdrawal of accruing rewards.

An Optimistic Withdrawal Recipientcontract takes three inputs when deployed:

- Aprincipal
- address: The address that controls where the principal ether will be transferred post-exit.
- Areward
- address: The address where the accruing reward ether is transferred to.
- The amount of ether that makes up the principal.

This contractassumes that any ether that has appeared in it's address since it was last able to do balance accounting is skimming reward from an ongoing validator (or number of validators) unless the change is > 16 ether. This means balance skimming is immediately claimable as reward, while an inflow of e.g. 31 ether is tracked as a return of principal (despite being slashed in this example).

danger Worst-case mass slashings can theoretically exceed 16 ether, if this were to occur, the returned principal would be misclassified as a reward, and distributed to the wrong address. This risk is the drawback that makes this contract variant 'optimistic'. If you intend to use this contract type,it is important you understand and accept this risk , however minute.

The alternative is to use an splits.orgwaterfall contract , which won't allow the claiming of rewards until all principal ether has been returned, meaning validators need to be exited for operators to claim their CL rewards. This contract fits both design goals and can be used with thousands of validators. It is safe to deploy an Optimistic Withdrawal Recipient with a principal higher than you actually end up using, though you should process the accrued rewards before exiting a validator or the reward recipients will be short-changed as that balance may be counted as principal instead of reward the next time the contract is updated. If you activate more validators than you specified in your contract deployment, you will record too much ether as reward and will overpay your reward address with ether that was principal ether, not earned ether. Current iterations of this contract are not designed for editing the amount of principal set.

OWR Factory Deployment

The OptimisticWithdrawalRecipient contract is deployed via afactory contract . The factory is deployed at the following addresses on the following chains.

Chain Address Mainnet 0x119acd7844cbdd5fc09b1c6a4408f490c8f7f522 Goerli 0xe9557FCC055c89515AE9F3A4B1238575Fcd80c26 Holesky Sepolia 0xca78f8fda7ec13ae246e4d4cd38b9ce25a12e64a

Exitable Withdrawal Recipient

A much awaited feature for proof of stake Ethereum is the ability to trigger the exit of a validator with only the withdrawal address. This is tracked inEIP-7002 . Support for this feature will be inheritable in all other withdrawal recipient contracts. This will mitigate the risk to a principal provider of funds being stuck, or a validator being irrecoverably offline.

Split Contracts

A split, or splitter, is a set of contracts that can divide ether or an ERC20 across a number of addresses. Splits are often used in conjunction with withdrawal recipients. Execution Layer rewards for a DV are directed to a split address through the use of afee recipient address. Splits can be either immutable, or mutable by way of an admin address capable of updating them.

Further information about splits can be found on the splits.org team'sdocs site . The addresses of their deployments can be foundhere .

Split Controllers

Splits can be completely edited through the use of thecontroller address, however, total editability of a split is not always wanted. A permissive controller and a restrictive controller are given as examples below.

(Gnosis) SAFE wallet

ASAFE is a common method to administrate a mutable split. The most well-known deployment of this pattern is theprotocol guild . The SAFE can arbitrarily update the split to any set of addresses with any valid set of percentages.

Immutable Split Controller

This is acontract that updates one split configuration with another, exactly once. Only a permissioned address can trigger the change. This contract is suitable for changing a split at an unknown point in future to a configuration pre-defined at deployment.

The Immutable Split Controllerfactory contract can be found at the following addresses:

Chain Address Mainnet Goerli 0x64a2c4A50B1f46c3e2bF753CFe270ceB18b5e18f Holesky Sepolia Edit this page Previous DV Launchpad Next Intro