

Percentage Transfer Manager

- **Contract name:** PercentageTransferManager.sol
- **Type:** Transfer Manager Module
- **Compatible Protocol Version:** TBD
- **Associated LucidChart:** TBD
- **Github Repo:** <https://github.com/PolymathNetwork/polymath-core/blob/master/contracts/modules/TransferManager/PercentageTransferManager.sol>

How it works

Summary: This contract is a transfer manager for limiting percentage of token supply that a single address can hold.

DISCLAIMER

Under certain conditions, the limit could be bypassed if a large token holder does the following:

- redeems a huge portion of their tokens. It will cause the total supply to drop
- which can result in some other token holders having a percentage of tokens
- higher than the intended limit.

Key functionalities (as defined in the Smart Contract)

Initialization

Configure

Summary: This function is used to initialize the variables on the entire contract

```
/**  
 * @notice Used to initialize the variables of the contract
```

```

* @param _maxHolderPercentage Maximum amount of ST20 tokens(in
%) can hold by the investor
*/
    function configure(uint256 _maxHolderPercentage, bool _all
owPrimaryIssuance)
    }

```

Verify Transfer

Summary: This function is used to verify the transfer transaction and prevent a given account to end up with more tokens than it is allowed to. It also works by checking if an address is on the whitelist, then allowing that address to hold more than maxHolderPercentage of the tokens.

```

/** @notice verify the transfer transaction and prevent a give
n account to end up with more tokens than allowed
* @param _from Address of the sender
* @param _to Address of the receiver
* @param _amount The amount of tokens to transfer
*/
    function verifyTransfer(address _from, address _to, uint25
6 _amount, bytes /* _data */, bool /* _isTransfer */)

    }

```

ChangeHolderPercentage

Summary: This function sets the maximum percentage that an individual investor / token holder can hold.

```

/**
* @notice maximum percentage that an individual token holder c
an hold

```

```

* @param _maxHolderPercentage is the new maximum percentage (multiplied by 10**16)
*/
function changeHolderPercentage(uint256 _maxHolderPercentage)
}

```

ModifyWhitelist

Summary: This function allows the issuers to add or remove investor addresses from their STO whitelist.

```

/**
* @notice adds or removes addresses from the whitelist.
* @param _investor is the address to whitelist
* @param _valid whether or not the address is to be added or removed from the whitelist
*/
function modifyWhitelist(address _investor, bool _valid)
}

```

ModifyWhitelistMulti

Summary: This function allows issuers to add or remove investor addresses from their STO whitelist.

```

/**
* @notice adds or removes addresses from the whitelist.
* @param _investors Array of the addresses to whitelist
* @param _valids Array of boolean value to decide whether or not the address it to be added or removed from the whitelist

```

```
*/  
    function modifyWhitelistMulti(address[] _investors, bool[]  
_valids)  
    }
```

SetAllowPrimaryIssuance

Summary: This function allows the issuer to set whether or not they want to consider a primary issuance of transfers.

```
/**  
 * @notice yes or no to allow primary issuance transfers  
 * @param _allowPrimaryIssuance whether to allow all primary is  
suance transfers  
 */  
    function setAllowPrimaryIssuance(bool _allowPrimaryIssuanc  
e)  
    }
```

Get Permissions

Summary: This function returns a permission flag for the associated percentage transfer manager.

```
/**  
 * @notice returns a permissions flag  
 */  
    function getPermissions()  
    }
```

Special considerations / notes

None

Troubleshooting / FAQs

None

Know Issues / bugs

- The PercentageTM has a flag the allows for bypassing the restriction on “primary issuance”. Since on 2.0.0 we can’t differentiate whether minting is coming from and STO or Issuer, this flag will also allow to bypass the restriction when the issuer is minting tokens.

Changelog

(Pre-v1.5.0)