

StakeSnapshot - 1.0.0

Summary

This contract is a companion contract to the main stake contract. It ensures snapshots are created correctly at the right time.

Parameters

Parameter	Description	Type	Default
imPaideiaDaoKey	Token ID of the dao config nft	Coll[Byte]	
stakeStateTokenId	Token ID of the stake state nft	Coll[Byte]	

Registers

Register	Description	Type
----------	-------------	------

Assets

Token name	amount
------------	--------

Transactions

Transaction name	Transaction description
StakeSnapshot	In this transaction 1 or more SplitProfit boxes are handled by splitting them between the treasury and the stake state

StakeSnapshot

In this transaction 1 or more SplitProfit boxes are handled by splitting them between the treasury and the stake state

Context Variables

CV Index	Type	Description
0	Coll[Byte]	Proof for retrieving current DAO Config values

Config

Config key	Type	Description
im.paideia.contracts.staking.snapshot	PaideiaContractSignature	Signature of the stake snapshot contract
im.paideia.staking.emission.amount	Long	Amount of governance tokens to be distributed each staking cycle from the treasury to the stakers
im.paideia.staking.emission.delay	Long	Rewards are delayed by this number of staking cycles
im.paideia.staking.cyclelength	Long	Length (in ms) of a staking cycle
im.paideia.staking.weight.pureparticipation	Byte	Percentage pure participation counts towards earning rewards

im.paideia.staking.weight.participation	Byte	Percentage participation counts towards earning rewards
im.paideia.dao.tokenid	Coll[Byte]	Governance Token ID of the DAO

Inputs

Utxo type	Description
StakeState	The utxo containing the staking state
StakeSnapshot	The utxo ensuring proper execution of calculating the compound

Data Inputs

Utxo type	Description
DAO Config	Config utxo of this DAO

Outputs

Utxo type	Description
StakeState	The utxo containing the updated staking state
StakeSnapshot	A copy of the input utxo
Miner	Miner fee

Conditions

1. DAO Config input is correct
2. Stake state input is correct
3. The new snapshot matches current state and is added to end of snapshot queue
4. Snapshot queue pop done correctly
5. Snapshot queue matches emission delay
6. Profit is reset for the next cycle
7. Time for next snapshot is set according to cycle length
8. Snapshot is not taken too early
9. Register values not relevant to this action in the stake state are unchanged