Core Contracts (Wrapper/Registry)

How to interact with CELO assets using the wrapper and registry Celo Core Contracts.

Interacting with CELO & cUSD

celo-blockchain has two initial coins: CELO and cUSD (Mento stabletoken). Both implement the ERC20 standard, and to interact with them is as simple as:

const goldtoken =

await kit . contracts . getGoldToken ( ) ;

const balance =

await goldtoken . balanceOf (someAddress) ; To send funds:

const oneGold = kit . web3 . utils . toWei ("1" ,

"ether") ; const tx =

await goldtoken . transfer (someAddress , oneGold) . send ({ from : myAddress , }) ;

const hash =

await tx . getHash ( ) ; const receipt =

await tx . waitReceipt ( ) ; To interact with cUSD, is the same but with a different contract:

const stabletoken =

await kit . contracts . getStableToken ( ) ;

Interacting with Other Celo Contracts

Apart from GoldToken and Mento stabletokens, there are many core contracts.

For the moment, we have contract wrappers for:

- Accounts
- Attestations
- BlockchainParameters
- DobleSigningSlasher
- DowntimeSlasher
- Election
- Escrow
- Exchange (Uniswap kind exchange between Gold and Stable tokens)
- GasPriceMinimum
- GoldToken
- Governance
- LockedGold
- Reserve
- SortedOracles
- Validators
- StableToken

A Note About Contract Addresses

Celo Core Contracts addresses, can be obtained by looking at theRegistry contract. That's actually howkit obtain them.

We expose the registry api, which can be accessed by:

const goldTokenAddress =

await kit . registry . addressFor (CeloContract . GoldToken) ;

Accessing web3 contract wrappers

Some user might want to access web3 native contract wrappers.We encourage to use the Celo contracts instead to avoid mistakes.

To do so, you can:

const web3Exchange =

await kit . _web3Contracts . getExchange ( ) ; We expose native wrappers for all Web3 contracts.

The complete list is:

- Accounts
- Attestations
- BlockchainParameters
- DoubleSigningSlasher
- DowntimeSlasher
- Election
- EpochRewards
- Escrow
- Exchange
- FeeCurrencyWhiteList
- GasPriceMinimum
- GoldToken
- Governance
- LockedGold
- Random
- Registry
- Reserve
- SortedOracles
- StableToken
- Validators

Debugging

If you need to debugkit , we use the well known debug node library.

So set the environment variableDEBUG as:

DEBUG

"kit:*,