Integration tests for smart contracts

There is aniproved version of integration tests framework which is used for testing new smart contracts. It is based on top ofCosmopark andContracts2ts .

Cosmopark

Cosmopark – is a tool that allows to run multiple networks on the same machine. Under the hood it uses docker containers and require docker images for a network you want to run. It can spin upHermes Relayer andNeutron Query Relayer for a deployment if required.

Contracts2ts

Contracts2ts – is a tool that allows to generate typescript clients for set of contracts. It uses json generated schemas from contracts withwrite_api method.

How to use

1. Clone therepository
2. Place your own contracts source code in thecontracts
3. folder. Thepump
4. contract is just an example which can be removed
5. Runmake schema
6. to generate json schemas for your contracts
7. make build
8.
    - Build your contracts
9. cd integration-tests
10. yarn
11. yarn build-images
12.
    - Build docker images for the networks used
13. yarn build-ts-client
14.
    - Build TS client for your contracts
15. Implement you own tests in thesrc/testcases
16. folder. Thepump
17. test files are examples which can be removed
18. yarn test

What's inside the tests (integration_tests

folder)

src/testSuite.ts contains configuration of the networks used with defined network params and docker image names.

src/testcases folder contains the tests for the contracts. Each test is a separate file with a set of tests for a contract. Please check the existing tests to understand how to write your own. In thebeforeAll method you can see the configuration of the networks and the deployment of the contracts.

src/vite.config.ts contains of the configuration for the tests.

Environment variables

MAX_THREADS - maximum threads to run tests in parallelPrevious Integration tests for chain Next Overview