Governance

Introduction

Governance is the mechanisms through which participants collectively make decisions about the rules, parameters, and policies that govern a given network. This can include things like protocol upgrades, parameter adjustments, and even more fundamental decisions about the network's direction and purpose. Changes are requested through *governance proposals which represent formal suggestions or requests put forth by a member or participant of a network or community to make changes to that network. These proposals are typically submitted to a decentralized governance system, where participants in the network have the opportunity to review, discuss, and ultimately vote on the proposed changes.

The approval process and the required majority for a proposal to pass can vary depending on the specific governance model of the network. Once a proposal is approved, the changes outlined in the proposal are typically implemented according to the network's governance mechanisms. This could involve deploying new smart contracts, updating software, or making adjustments to on-chain parameters.

How to take part in the governance mechanism

If you wish to take part in governance, you will either need to be running afull validator node or you need tohave delegated stake to an existing validator .

For additional information, visit thevalidators ↗ section of our documentation, or thedelegation ↗ guide.

Stake delegation

If you want to delegate your stake to a validator, the following CLI command should be used:

fetchd

tx

staking

delegate

< VALOPER_ADDRES S

< AMOUN T

--from

< KEY_NAM E

where the begins with the prefixfetchvaloper1... and thefield contains the currency denomination.

For instance:

fetchd

tx

staking

delegate

fetchvaloper1cct4fhhksplu9m9wjljuthjqhjj93z0s97p3g7

1000 atestfet

--from

agent Check out ourdelegation ↗ guide for further information on how to delegate your tokens to a validator using the CLI.

Proposals overview

There arethree types of governance proposal:

1. Text proposals
2. : these are the most basic type of proposal. They can be used to get the opinion from participants of the network on a

given topic.
3. Parameter proposals
4. : these proposals are used to update the value of an existing software parameter of the network.
5. Software upgrade proposals
6. : these are used to propose an upgrade of thefetchd
7. software, particularly in cases where the software changes might not necessary be backwards compatible or in some way present a major update to the network.

The proposal process

Any FET holder can submit a proposal .

In order for the proposal to be open for voting, it needs to come with a deposit that is greater than a parameter calledminDeposit . The deposit need not be provided in its entirety by the submitter. If the initial proposer's deposit is not sufficient, the proposal enters thedeposit period status. Then, any FET holder can increase the deposit by sending adepositTx transaction to the network.

Once the deposit reachesminDeposit , the proposal enters thevoting period , which lasts 2 weeks. Any bonded FET holder can then cast a vote on this proposal, by choosing one of the following options for voting:

- Yes.
- No.
- NoWithVeto.
- Abstain.

At the end of the voting period, the proposal is accepted if there aremore than 50% Yes votes (excluding Abstain votes) andless than 33.33% of NoWithVeto votes (excluding Abstain votes).

Generating proposals

When creating a proposal, you will create a proposalJSON file with all the relevant information.

An example of atext proposal is shown below:

{ "title" :

"Switch to semantic commit messages for fetchd" , "description" :

"This proposal is advocating a switch to sematic commit messages. You can find the full discussion at: https://github.com/fetchai/fetchd/issues/231" , "type" :

"Text" , "deposit" :

"10000000000000000000atestfet" } ℹ It is always recommended that the description of a text proposal has a link to a Github issue with the full proposal text along with the discussions about it. After having created the JSON file, you cangenerate the text proposal on chain by running the following command:

fetchd tx gov submit-proposal --proposal proposal.json --from

Increasing the deposit for a proposal

If you want toincrease the deposit of a proposal , you could do this by running the following command:

fetchd tx gov deposit 100atestfet --from

For instance:

fetchd tx gov deposit 2 100atestfet --from validator

If you want to get theproposalID , you will need to use thetxhash obtained when the proposal was submitted and run the following command:

fetchd query tx

For additional information on how to create a governance proposal using CLI, have a look at ourdedicated guide ↗ .

Listing current proposals

Current proposals can be retrieved by using theCLI or from theblock explorer .

If you wish toget the list of current proposals and their correspondingproposal-ids , then you need to run the following command:

fetchd query gov proposals

Voting on a proposal

If you want tovote for a proposal , run the following command:

fetchd tx gov vote

For instance:

fetchd tx gov vote 5 yes --from validator

ℹ When using CLI commands make sure that your CLI is pointing at the correct network. Check the CLI introduction ↗ in our documentation for additional details on the topic.

Was this page helpful?

How to run a single node test network How to get testnet tokens via the Token Faucet