

DAO configuration parameters[mainnet]

This document describes the configuration parameters for the Neutron DAO on the mainnet (testnet parameters differ for easier testing/administration).

This page is of interest to those who are interested in the Neutron DAO internals.

Neutron DAO has 3 proposal modules:

- Single choice
- Multiple choice
- Overrule

The first two are used for regular proposals, while the last one is used for overruling the decisions of the subDAOs. Furthermore, the users create single and multiple-choice proposals, while overrule proposals are created automatically by subDAOs. Thus, single and multiple-choice proposals are primarily consistent regarding parameters, and the overrule proposal module is significantly different.

Single and multiple proposals

Deposit

The deposit amount is 1000 NTRN. The user (or smart contract) must deposit 1000 NTRN to submit a proposal. The deposit is refunded according to the deposit refund policy. The deposit refund policy, only `_passed` means the deposit is refunded if the proposal passes and is executed. If a proposal is rejected explicitly or hasn't got a quorum, the deposit stays on the pre-propose module's balance, and the DAO can withdraw it eventually.

Proposal submission restrictions

Open proposal submission is set to `true`. It means that anyone can submit a proposal. The reasoning behind this is that the voting power in Neutron DAO is based on NTRN tokens; becoming a DAO member requires holding NTRN tokens, with even 1 NTRN being sufficient. So, a proposal submission restricted to DAO members wouldn't make sense.

Voting period

The voting period is set to 2 weeks. This timing is typical for many DAOs and, according to the governance practice, is good enough to avoid proposal rejections because of not getting the quorum while still being long enough to review the proposal carefully.

Quorum and threshold

The quorum is set to 5%. Any proposal can only pass if at least 5% of the voting power voted for it. This value might look low because it reaches 10%-40% for many Cosmos chains. However, the big difference between Neutron and most other Cosmos chains is that Neutron DAO is based on smart contracts and DAO DAO. Cosmos chains based on the gov module have native delegation mechanics based on stake delegation to validators. A bigger quorum makes sense since 100% of the stake is delegated to validators who actively participate in governance. Delegations are absent in Neutron DAO (yet), so it's critical to have a quorum of reasonable size so that DAO will stay operable.

Here, single and multiple-choice proposals are different in behavior. A single-choice proposal has a quorum of 5%, and at the same time, it has a threshold of 50%. It means that a single-choice proposal passes if at least 50% of the votes are "for" the proposal. For example, if 6% of the total voting power participated in voting, 4% voted "for," and 2% voted "against," the proposal would pass.

The multiple-choice proposal module has only a quorum and no threshold. If the quorum is reached, the option with the most votes wins. For example, if there are 5 options, the turnout is 6%, and votes are distributed 2%/1%/1%/1%/1%, the first option wins.

Revoting

Revoting is enabled both for single-choice proposals and for multi-choice proposals. There are a couple of reasons for this:

1. It allows DAO members to change their minds and vote for a different option during the voting period.
2. It doesn't allow the DAO to execute a proposal immediately after the quorum is reached.

For proposals, it means that they can only be executed once the voting period is over, even if a quorum/threshold is reached earlier (since it's implied that someone can change their mind).

Proposal execution parameters

The `only_members_execute` parameter is set to `false`. It means that anyone can execute a proposal after it passed. The reasoning for this is that the DAO is a decentralized organization, and it's not reasonable to restrict the execution of proposals to DAO members since one can become a DAO member pretty easily by buying 1 unit and depositing it to the NTRN vault. Moreover, it simplifies possible automation.

The `close_proposal_on_execution_failure` option is set to `true`. It means that if a proposal fails to execute, it becomes closed, and there's no way to make another try to execute it. It's done for security reasons. It's implied that proposals should be carefully tested before getting submitted to the DAO. Execution failure might be evidence of unexpected changes in the environment or a bug in external contracts. If that's the case, DAO probably doesn't want this proposal to stick until the environment changes.

For example, there is a malformed external smart contract call in proposal A, and it's discovered during this proposal execution failure. Proposal B was created to adapt it to the proper external smart contract interface and was successfully executed. However, proposal A isn't closed and can be re-executed. The admin of the external smart contract can adapt the interface to proposal A and re-execute it, while the DAO does not plan it. While this case is hypothetical, it's more secure to keep the `close_proposal_on_execution_failure` option as `true`.

Overrule proposals

Deposit

Deposit is turned off for overrule proposals because they are created automatically by subDAOs. Also, they're meant to be rejected in normal conditions by not reaching the quorum, so a deposit makes no sense.

Proposal submission restrictions

Technically, proposal creation is open. However, overrule proposals are created automatically by subDAOs, so it's not a matter of concern. Overrule proposal creation is carefully restricted on the source code level to prevent the creation of duplicates and spam. Since overrule proposals are created automatically by subDAOs and duplications are prohibited, there's essentially no chance for a user to create an overrule proposal while technically it's possible.

Voting period

The voting period is 3 days. Overrule proposals voting period should be a trade-off between two considerations:

1. It should be long enough to allow DAO members to express their concerns about the underlying subDAO proposal.
2. It should be short enough to make subDAOs operable.

Given that, 3 days is a reasonable trade-off since it's a minimal timeframe that covers weekends and leaves enough time for DAO members to review the proposal.

Quorum and threshold

The threshold is set to 0.5% of the total voting power. Only 0.5% of total voting power is required to overrule any subDAO proposal. This threshold should be low by its nature: in controversial cases, the DAO should be able to prevent the subDAO from doing anything that might be harmful to the DAO. Of course, it also creates a potential surface for sabotage. A whale big enough or a coordinated telegram group can effectively block any subDAO proposal. However, this situation isn't expected in the first place. Secondly, if it happens, it allows DAO to identify the issue of the DAO members' subset being non-aligned (or even malicious) and take the necessary measures to fix it. Since DAO has ultimate power over subDAOs, it's possible. Considering the subDAO mechanics as a tool to make DAO more efficient by alleviating governance pressure, a low threshold is a reasonable trade-off. In the worst case, the DAO rolls back to the default situation, where the DAO itself makes all the decisions.

Unlike single and multi proposals, overrule proposals have an `absolute_percentage` type of threshold. It means that for a proposal to pass, it's only required to get the threshold of yes votes. Even if there are no votes, the proposal still passes if there are more than the threshold. This is done to make overrule proposals pass (and reject the underlying subDAO proposal) even when the subDAO proposal is slightly controversial.

Revoting

Revoting is set to `false`. It's done to be able to execute proposals as soon as the quorum is reached. It's critical to overrule proposals as soon as possible since they have a limited execution time. The overrule proposal execution after the end of the voting period makes no sense since a subDAO proposal has already been executed by that time.

Proposal execution parameters

Proposal execution isn't restricted for the same reasons as single and multiple-choice proposals (there's no sense in restricting it + more automation opportunities). The `close_proposal_on_execution_failure` option is set to `true`. Because overrule proposals are of a very special format and (in general) are created automatically, the chance of overrule proposals to fail is very low. Moreover, the failure during overrule proposal execution indicates some problems with DAO internals. In this case, the better strategy is restricting possible permissionless actions, so re-execution is disabled. [Previous Differences from DAO-DAO Next Overview](#)