# #

Governance

# #

Concepts

The governance process is divided in a few steps that are outlined below:

- Proposal submission:
- Proposal is submitted to the blockchain with a deposit.
- Vote:
- Once deposit reaches a certain value (MinDeposit
-), proposal is confirmed and vote opens. Bonded Iris holders can then sendTxGovVote
- transactions to vote on the proposal.
- If the proposal involves a software upgrade:
-
    - Signal:
-
    - Validators start signaling that they are ready to switch to the new version.
-
    - Switch:
-
    - Once more than 75% of validators have signaled that they are ready to switch, their software automatically flips to the new version.

#

Proposal submission

#

Right to submit a proposal

Every account can submit proposals by sending a MsgSubmitProposal transaction. Once a proposal is submitted, it is identified by its unique proposalID.

#

Proposal types

A proposal includes an array ofsdk.Msg s which are executed automatically if the proposal passes. The messages are executed by the governanceModuleAccount itself. Modules such asx/upgrade , that want to allow certain messages to be executed by governance only should add a whitelist within the respective msg server, granting the governance module the right to execute the message once a quorum has been reached. The governance module uses theMsgServiceRouter to check that these messages are correctly constructed and have a respective path to execute on but do not perform a full validity check.

#

Deposit

To prevent spam, proposals must be submitted with a deposit in the coins defined by theMinDeposit param.

When a proposal is submitted, it has to be accompanied with a deposit that must be strictly positive, but can be inferior toMinDeposit . The submitter doesn't need to pay for the entire deposit on their own. The newly created proposal is stored in an inactive proposal queue and stays there until its deposit passes theMinDeposit . Other token holders can increase the proposal's deposit by sending aDeposit transaction. If a proposal doesn't pass theMinDeposit before the deposit end time (the time when deposits are no longer accepted), the proposal will be destroyed: the proposal will be removed from state and the deposit will be burned (see x/govEndBlocker). When a proposal deposit passes theMinDeposit threshold (even during the proposal submission) before the deposit end time, the proposal will be moved into the active proposal queue and the voting period will begin.

The deposit is kept in escrow and held by the governanceModuleAccount until the proposal is finalized (passed or rejected).

#

Deposit refund and burn

When a proposal is finalized, the coins from the deposit are either refunded or burned according to the final tally of the proposal:

- If the proposal is approved or rejected but not vetoed, each deposit will be automatically refunded to its respective depositor (transferred from the governanceModuleAccount
-).
- When the proposal is vetoed with greater than 1/3, deposits will be burned from the governanceModuleAccount
- and the proposal information along with its deposit information will be removed from state.
- All refunded or burned deposits are removed from the state. Events are issued when burning or refunding a deposit.

#

Vote

#

Participants

Participants are users that have the right to vote on proposals. On the Cosmos Hub, participants are bonded Iris holders. Unbonded Iris holders and other users do not get the right to participate in governance. However, they can submit and deposit on proposals.

Note that someparticipants can be forbidden to vote on a proposal under a certain validator if:

- participant
- bonded or unbonded Iris to said validator after proposal entered voting period.
- participant
- became validator after proposal entered voting period.

This does not preventparticipant to vote with Iris bonded to other validators. For example, if aparticipant bonded some Iris to validator A before a proposal entered voting period and other Iris to validator B after proposal entered voting period, only the vote under validator B will be forbidden.

#

Voting period

Once a proposal reachesMinDeposit , it immediately entersVoting period . We defineVoting period as the interval between the moment the vote opens and the moment the vote closes.Voting period should always be shorter thanUnbonding period to prevent double voting. The initial value ofVoting period is 2 weeks.

#

Option set

The option set of a proposal refers to the set of choices a participant can choose from when casting its vote.

The initial option set includes the following options:

- Yes
- No
- NoWithVeto
- Abstain

NoWithVeto counts asNo but also adds aVeto vote.Abstain option allows voters to signal that they do not intend to vote in favor or against the proposal but accept the result of the vote.

Note: from the UI, for urgent proposals we should maybe add a ‘Not Urgent’ option that casts aNoWithVeto vote.

#

Weighted Votes

The weighted vote feature allows a staker to split their votes into several voting options. For example, it could use 70% of its voting power to vote Yes and 30% of its voting power to vote No.

Often times the entity owning that address might not be a single individual. For example, a company might have different stakeholders who want to vote differently, and so it makes sense to allow them to split their voting power. Currently, it is not

possible for them to do "passthrough voting" and giving their users voting rights over their tokens. However, with this system, exchanges can poll their users for voting preferences, and then vote on-chain proportionally to the results of the poll.

#

Quorum

Quorum is defined as the minimum percentage of voting power that needs to be casted on a proposal for the result to be valid.

#

Threshold

Threshold is defined as the minimum proportion ofYes votes (excludingAbstain votes) for the proposal to be accepted.

Initially, the threshold is set at 50% with a possibility to veto if more than 1/3rd of votes (excludingAbstain votes) areNoWithVeto votes. This means that proposals are accepted if the proportion ofYes votes (excludingAbstain votes) at the end of the voting period is superior to 50% and if the proportion ofNoWithVeto votes is inferior to 1/3 (excludingAbstain votes).

Proposals can be accepted before the end of the voting period if they meet a special condition. Namely, if the ratio ofYes votes toInitTotalVotingPower exceeds 2:3, the proposal will be immediately accepted, even if theVoting period is not finished.InitTotalVotingPower is the total voting power of all bonded Iris holders at the moment when the vote opens. This condition exists so that the network can react quickly in case of urgency.

#

Inheritance

If a delegator does not vote, it will inherit its validator vote.

- If the delegator votes before its validator, it will not inherit from the validator's vote.
- If the delegator votes after its validator, it will override its validator vote with its own. If the proposal is urgent, it is possible that the vote will close before delegators have a chance to react and override their validator's vote. This is not a problem, as proposals require more than 2/3rd of the total voting power to pass before the end of the voting period. If more than 2/3rd of validators collude, they can censor the votes of delegators anyway.

#

Validator’s punishment for non-voting

At present, validators are not punished for failing to vote.

#

Governance address

Later, we may add permissioned keys that could only sign txs from certain modules. For the MVP, theGovernance address will be the main validator address generated at account creation. This address corresponds to a different PrivKey than the Tendermint PrivKey which is responsible for signing consensus messages. Validators thus do not have to sign governance transactions with the sensitive Tendermint PrivKey.

#

Software Upgrade

The governance process for the software upgrade is described inUpgrade

#

Usage Scenario

#

Parameter change

The parameters of modules can be changed by demand through a proposal of parameter change.

Query module parameters which can be changed through gov. e.g. query the service params

iris queryservice params# Parameter list arbitration_time_limit: 432000s base_denom: stake complaint_retrospect: 1296000s max_request_timeout:"100" min_deposit: - amount:"6000" denom: stake min_deposit_multiple:"200" service_fee_tax:"0.100000000000000000" slash_fraction:"0.001000000000000000" tx_size_limit:"4000"

Send proposal for parameters change

echo '{ "title": "Service Param Change", "description": "Update max request timeout", "changes": [{ "subspace": "service", "key": "MaxRequestTimeout", "value": 150 }], "deposit": "1000iris" }'

proposal.json

iris tx gov submit-legacy-proposal param-change proposal.json--from

< key-name> --fees = 0 .3iris --chain-id= irishub

#

Community pool spending

The community pool funds can be spent through the governance process.

Submit a proposal for community pool spending

echo '{ "title": "Community Pool Spend", "description": "Developer rewards", "recipient": "iaa1s5afhd6gxevu37mkqcvvsj8qeylhn0rz46zdlq", "amount": "10000iris", "deposit": "1000iris" }'

proposal.json

iris tx gov submit-legacy-proposal community-pool-spend proposal.json--from

< key-name> --fees = 0 .3iris --chain-id= irishub

#

Software upgrade

Usage on the software upgrade is introduced inUpgrade