

# Governance Guide

## Governance Guide

With the addition of new features in Aave V3, the scope for potential governance proposals is substantially widened:

- Supply caps, borrow caps, and isolation mode can reduce the risk for onboarding new assets
- With the `ROLE_ADMIN`
- designation, Aave Governance can approve a variety of permissions
- - Roles could potentially be used for automated asset listing or parameter adjustment
- \*
- Every aToken and debtToken now supports multiple rewards per asset
- 

All proposals follow the same general process from idea to execution, which is detailed in this guide. There are also pages with templates for completing these steps with standard proposal types:

- [Asset Listing](#)
- [Rewards](#)
- [Permissions](#)
- [Parameter Tuning](#)
- 

## General Governance Process

[Governance Forum](#) → [Snapshot](#) → [AIP](#) → [Create Proposal](#) → [Voting and Execution](#)

### Governance Forum

The first step in the proposal process is introducing the proposal to the community through the [governance forum](#). The Aave Request for Comments ("ARC") is a post on the forum to allow the community to discuss the details of a proposal. Here is a guide for [Creating an ARC](#); you can also find templates for specific proposal types in the links above.

### Snapshot

The [Aave Snapshot Space](#) is a place for voters to gauge community sentiment for on-chain votes, or decide off-chain proposals. The Snapshot poll should contain the same information as the ARC, and should always link back to the original ARC post.

In the future, the results of Snapshot votes could be relayed for on-chain execution through SnapshotX, more details on the [governance forum post](#).

### AIP

Any proposal which passes the Snapshot phase and requires an on-chain payload moves to the "AIP" phase. An AIP - Aave Improvement Proposal - is the proposal payload which will be submitted on chain and voted on by the community. The AIP consists of two main components: the AIP text and implementation.

The AIP text is a structured text description of the proposal which is uploaded to IPFS, and the IPFS hash is submitted with the on-chain proposal. To generate an AIP text for your proposal, create a fork of the [AIP repo](#), and follow the steps in the README.

The AIP implementation is the array of transactions which will be executed if the proposal succeeds. All AIPs should be tested (and potentially audited) to ensure safety and correctness. See the proposal type pages for steps on generating the execution parameters.

The AIP text and implementation should be reviewed (e.g., devs through Aave Grants DAO, risk/security contributors such as [Gauntlet](#) or [Certora](#)) before submitting on-chain.

### Create Proposal

Once the AIP has been reviewed, the next step is to create the proposal on-chain. To do this, the [create](#) function is called on the AaveGovernanceV2 contract with the proposal payload, which can be generated using the [Aave Proposals AIP](#) repo.

`create` can only be called if the user has more proposition power than the [proposal threshold](#). The required proposition power depends on the executor being used.

### Voting and Execution

The proposal life cycle is detailed in the diagram or in text form below:

?

Once a proposal is created, it's status is set to PENDING and will move to ACTIVE once the [voting delay](#) period has elapsed.

Once the delay period is passed and [voting duration](#) has elapsed, the proposal is status is set to SUCCEEDED if:

- The voting power (in % of total voting power) of for-votes reaches the quorum set by the [minimum quorum](#)
- parameter, AND
- The difference between for-votes and against-votes (in % of total voting power) exceeds the vote differential threshold set by the [vote differential](#)
- parameter
- 

Or set to FAILED otherwise.

A SUCCEEDED proposal can be queued and will be executed after the execution [delay](#) and before [grace period](#) expiration.

The validation and execution of the proposal is performed by the time lock executor. The proposal is set to QUEUED until it is EXECUTED , or if a queued proposal is not executed before expiration, it is set to EXPIRED.

[Previous Testing Guide](#) [Next Asset Listing](#) Last updated 4 months ago On this page \* [Governance Guide](#) \* [General Governance Process](#)

Was this helpful?