TL;DR

Some basic guidelines and tips to better understand the technical aspects of defining initial asset configurations, when proposing an Aave v3 listing.

As a complement to the risk assessment provided by expert parties, we recommend all external teams looking for/working on Aave v3 listings take a look at these technical tips when writing a proposal on this forum, before the voting steps.

Facilitating technical integrations: context

One of the main challenges of open software systems like the Aave protocol is the documentation and guidance around its integration procedures.

In the case of Aave, being a project with completely decentralized contributors, this gets accentuated, and sometimes some types of contributions can get a bit complicated without use-case-specific technical documentation.

On our initial scope of BGD, we targeted with high priority improving this section of the ecosystem, as it is completely fundamental that all parties bringing value to Aave have good resources to progress, even without anybody else supporting them.

On the one hand, there is work done (and to be done) on the pure implementation level, making available to the community different templates (e.g. https://github.com/bgd-labs/aave-v3-crosschain-listing-template), examples (e.g. https://github.com/bgd-labs/example-aave-v2-listing) and misc-utils.

On the other hand, we think there are aspects to be improved on the documentation side, for example, given use-case-specific written guidance, which combined with code examples, can lead to really easy integration of Aave.

Asset listings (v3)

Over the past few months, we have been giving technical support to numerous teams (usually of other communities) on which components of the protocol need to be “touched” when trying to submit a proposal to list a new asset on an instance of Aave v3.

Doing that, we have noticed an important lack of awareness by those external teams of the list of parameters that should be presented to the community to approve a listing and the technical limitations of the different configurations. This is obviously normal, given that Aave v3 is still months-old, and due to how features-rich it is. But should be improved over time.

This post aims to, from a technical perspective, increase the transparency of required potential listing parameters, to be presented on this forum first, and to mandatorily be included in the Snapshot proposal and on-chain governance phases of asset listings.

It is important to highlight that every asset listing is different, and this guide only tries to give some solid ground for risk experts or other types to suggest deviations from it.

Parameters’ guideline

Loan to value

The loan to value (LTV) defines how much you can borrow against a certain asset when putting it as collateral. For assets that can be used as collateral, the LTV must be non-zero.

Sanity checks:

Currently, the LTV of assets is designed in a way to allow for standard volatility of an asset to be absorbed within a margin, in order to avoid instant-liquidations when opening a position, and in general to set a “safe” range.

Practical tips

:

- Most assets start ≤ 60% LTV, but the LTV might be increased in follow-up proposals in case the asset proofs to be stable for a longer period of time.

Liquidation threshold

While the LTV defines how much you can borrow against your collateral, the liquidation threshold defines at which level of over-collateralization you can be liquidated. There should be an appropriate margin between the two.

Practical tips:

- The liquidation threshold must be above the LTV and must allow for a “safe” margin to not put users at immediate liquidation risk when opening a borrowing.
- While for stablecoins a 5% margin might be sufficient, for more volatile assets a 10-20% margin might be more appropriate.

Liquidation bonus

The liquidation bonus determines a percentage of the liquidated borrowing that gets received extra from the collateral by the liquidator. The liquidation bonus can also range between the diff between the liquidation threshold and 100% as otherwise there is no collateral to pay the bonus. Practically that means an asset with a 95% liquidation threshold can only have up to a 5% liquidation bonus.

Sanity checks:

When an asset has really high liquidation threshold (> 90% usually), it is important to not define a too high liquidation bonus, or even better, reduce the liquidation threshold.

This is to protect against a situation where the liquidation bonus percentage plus liquidation bonus is above 100%, which can put in danger the protocol if any positions get liquidated.

Practical tips:

- It’s important to understand that the more stable and liquid an asset is, the easier is to liquidate it even in harsh market downturns. Therefore, the bonus should be aligned with the aforementioned parameters.
- It’s reasonable to start a discussion at 5% for highly liquid and stable assets and at 8% for more volatile and less liquid assets

Price feed (oracle)

For an asset to be listed, a proper Chainlink price feed needs to be associated to the asset. They can be found on https://docs.chain.link/docs/reference-contracts/.

Sanity checks

:

- For assets to be listed on the Aave v2 Ethereum

pool you need an ETH

price feed.

- For all other networks, you need a USD

price feed.

- By general rule, better to avoid “on-chain” oracles, based purely on DEXes liquidity, as they can be prone to manipulation.

A/S/V tokens (aToken/Variable/Stable)

The underlying tokenization smart contracts represent supplies and borrowings on Aave for a particular asset. For most tokens, the standard implementations will be a sufficient while for tokens with rebase mechanisms or similar a custom implementation might be needed.

Practical tips

:

- If the token your looking into is a default erc20 token with no special functionality:
- In Aave v2 you can deploy the default implementations
- In Aave v3 you can reuse the existing default implementation of another asset already listed in the pool.
- In Aave v2 you can deploy the default implementations
- In Aave v3 you can reuse the existing default implementation of another asset already listed in the pool.

Interest rate strategy

Implementation of the mathematical curve defining the borrow rates based on utilization of the pool. It is designed with 2 slopes (per stable/variable) above and below an optimal utilization rate. It is used to create behaviors of rate increase/decrease depending on the utilization, depending on the asset type.

Practical tips

:

- Assets that have a clear borrow market usually aim for high utilization (e.g. stablecoins)
- Assets that are mostly used as collateral usually aim for moderate utilization
- You can reuse interest rate strategies used by other assets instead of deploying new ones as most stable and collateral assets will share the same parameters

Borrowing enabled

If an asset can be borrowable or not.

Sanity check

With a borrowing cap in place, most assets are suitable for borrowing.

Nevertheless, it makes sense to gauge community feedback, especially for governance tokens there could be potential for governance attacks if the asset was borrowable. For assets, without a clear borrowing profile, it might make sense to take the conservative step of disabling borrowing.

Practical tips:

- for governance tokens, it’s better to disable borrowing when there’s no clear use-case for borrowing.

Stable borrowing enabled

Allows the asset to be borrowed at a stable rate mode.

Practical tips:

- To enable stable borrowing a risk expert party needs to do a proper assessment, so better to start off with stable borrowing disabled.

Reserve Factor

Percentage of interest accrued to the protocol, discounted from the total that suppliers would receive.

Practical

tips

:

- Highly borrowed assets usually have a 10% reserve factor, while assets mostly used as collateral have a 20%.

Liquidation Protocol Fee

Percentage of liquidation bonus that is transferred to the protocol.

Practical

tips:

- Currently, liquidation protocol fees range between 10 and 20%. The rationale is that the higher the risk for the protocol, the higher the reward should be.

Debt Ceiling

A dollar-denominated cap of aggregated borrowings across the pool can be opened against a certain asset. A non-zero debt ceiling makes an asset isolated, meaning that it must be the only collateral for a certain address. All addresses have a shared debt ceiling against the isolated asset, which means there can only be a limited aggregated amount of borrowings across the pool using the isolated asset as collateral.

Sanity checks

Currently, assets that have a mint functionality, non-standard erc20 functionality, or similar unpredictable risk have a conservative debt ceiling to limit protocol exposure to sudden price spikes or similar.

Practical

tips:

- For isolated assets, it makes sense to start the debt ceiling at around 5% of the dollar-denominated market cap of the asset, but also no more than 5% of the Aave pool size.

Borrowable in isolation

When true, this asset can be borrowed against isolated assets.

As the debt ceiling is determined as a static dollar amount, only assets with a proven track record of stability are considered borrowable in isolation otherwise, the ceiling will be semi-functional as the ceiling might be surpassed due to asset volatility.

Sanity checks:

Currently, only stable assets that have a historically proven track record of staying very close to peg are considered borrowable in isolation.

Practical tips:

- At the moment, we think extensive risk analysis by an expert party should be done on this. It’s probably better to not enable assets as borrowable in isolation before that, but up to each asset listing proposal.

Supply Cap

A token-units-denominated amount of the maximum that can be supplied of the given asset in the pool. Once the cap is reached, no more tokens can be deposited into the protocol.

A conservative supply cap is recommended as it will act as an additional guard against various attacks (e.g. infinite mint). The cap can easily be increased in a follow-up proposal.

Sanity checks

:

A supply cap should always be present and never be above the total supply of that asset.

Practical tips

:

- It’s unlikely that the sole use-case of an asset is to be deposited into Aave, therefore a conservative supply cap would usually start with ~20%/30% of the total circulating supply of a certain asset and might be increased in follow-up proposals

Borrow Cap

A token-units-denominated amount of the maximum that can be borrowed in the pool. Once the cap is reached, no more tokens can be borrowed from the protocol.

Sanity checks:

Tokens that are mainly used as collateral and have no obvious borrowing use cases usually should start off with a conservative borrow cap slightly above optimal utilization. With this, the asset can still be borrowed while limiting the exposure of the protocol.

Practical tips

:

- If the supply cap is 100M and optimal utilization is at 45% the borrowing cap might be at 55M. Initially to limit protocol exposure, while still allowing for borrowing.

eMode (efficiency mode)

An eMode category defines a group of assets that can have tighter risk parameters, because of how correlated they are, allowing for better capital efficiency within the category. All assets within a certain eMode share the same eMode configuration parameters(loan to value, liquidation threshold, liquidation bonus & reserve factor).

An asset can be added to an existing eMode category or a new eMode category can be created. An eMode allows for better capital efficiency within the category of assets. All assets within a certain eMode share the same eMode configuration parameters(loan to value, liquidation threshold, liquidation bonus & reserve factor).

Sanity checks:

- A new eMode category only makes sense when there are at least two assets that would belong in that category. The LTV and liquidation threshold parameters should still allow for economically feasible liquidations in all circumstances.
- There needs to be a proven track record of historic price correlation between the assets inside an eMode, or other asset fundamentals.

Practical tips:

- An eMode can have a common oracle shared between all assets of the category. Only in very unique use cases setting a common oracle can make sense, usually, it’s better to have independent oracles per asset.

eModes might be groups of price-correlated assets like:

- wETH / stETH / wstETH / rETH / …
- wBTC / BTC.b / …
- USDC / DAI / sUSD / …
- agEUR / EURS / jEUR / …

Unbacked mint cap

The unbacked mint cap determines how much of an asset can be minted to be utilized by the Portal functionality.

Practical tips

:

- Currently, newly listed assets all start out with a 0 cap, especially as there are no whitelisted portals yet utilizing the feature

Siloed borrowing

An asset can be flagged as siloed which makes it borrowable only when being the single borrowed asset by an address. This feature is close to the debt ceiling / isolated assets but for the borrowing side.

Practical tip

:

- This feature needs to be assessed by a risk expert party and policies have to be defined around it.