

Summary

Recap of the 6-month continuous engagement for services between BGD Labs and the Aave DAO, starting on April 1st, 2024, and ending on October 1st, 2024.

What has been done?

1. Aave protocol

→ Aave v1

During this engagement, we moved forward with the third and final deprecation [phase of Aave v1](#), and the community approved now the instance to be considered out-of-support.

Apart from potential upcoming Rescue Missions, no extra development maintenance should create resources' overhead in the future.

→ Aave v2

The community has been slowly and softly deprecating v2 in favor of v3 instances, so from the technical side we have not worked on any further improvement on the system, focusing on monitoring/addressing any type of issues (e.g. the aforementioned aAMPL), and supporting entities creating governance proposals affecting v2.

The only development item we produced was the full deprecation of the stable rate mode on Aave v2, given that it was pretty important to be fully aligned with v3 on this. All the details and governance proposal can be found [HERE](#).

Ampleforth off-boarding resolution

Together with the Chaos Labs, the risk provider of the community, we supported the community to reach a conclusion on the Ampleforth incident described [HERE](#).

This involved doing an in-depth analysis to locate the root of the issue in the aAMPL/vAMPL custom implementation by the Ampleforth team, together with coordinating with Chaos Labs to propose a fair resolution for both aAMPL suppliers and the Aave DAO.

Unfortunately, this task took very important resources, and this should be a good lesson going forward, for example, to not introduce significant custom components on Aave when coming from external teams, unless the trust and reputation of those are unquestionable.

→ Aave v3

Improving Aave v3 has been most probably the most important focus of this engagement phase, both with visible/release-ready results, but also with important strategy changes that needed technical adaptations.

The following is a comprehensive list of work streams over Aave v3:

Aave v3.1 activation

Even if the work on v3.1 was mainly done on Aave <> BGD Phase 2, its criticality and the very important off-chain infrastructure work required to release it with maximum assurances and scalability, cause it to be fully active in production in the middle of Phase 3, on [THIS](#) governance proposal.

We consider the work/time invested was worth it, as it allows us to have a way faster release cycle (v3.2 to be activated approximately two months after, v3.3 upcoming), and with tooling (e.g. [Aave Origin](#)) following the same approach as all other Aave projects.

Aave v3.2 activation

We designed and implemented Aave v3.2, introducing Liquid EModes and removing unused logic in production related to the deprecated stable borrowing mode. A comprehensive description of Aave v3.2 can be found [HERE](#), and its activation is imminent via governance proposal, with ETA this upcoming week of 30th September.

Aave v3.3: WIP

While developing new versions of Aave, we need to do internal exercises of prioritizing features to introduce on each

release, mainly balancing complexity versus go-to-market timing.

Given the groundwork done on v3.1, in parallel with implementing v3.2 we have been working on several other improvements to release in an upcoming v3.3/v3.4. Some of them are related to the requirements of Aave Umbrella, while others are independent of the Aave protocol.

We estimate v3.3 to be approximately 75% ready.

Aave Generalised Risk Stewards (AGRS)

We did multiple internal iterations for the generalization of Risk Stewards from the currently active CapsPlusSteward, resulting in the proposal AGRS, together with a pilot integration program of AGRS with Chaos Labs Edge's Risk Oracle.

All details can be found [HERE](#).

Aave goes multi-pool in the same network

Historically, Aave has experience with maintaining multiple pools on the same network, but apart from minor exceptions on the "v2 era" (v2 main Ethereum, AMM, or even Aave Arc), the approach from v3 release was more on 1 network → 1 pool; and focus more on expanding to new networks.

However, last months the community decided to put effort and resources into all contribution fields to make multiple pools on the same network a reality, at scale. As a result, Aave currently has three v3.1 instances on Ethereum, of which Aave v3 Lido stands out with close to a \$700 million market size.

For this strategy change to happen relatively seamlessly operations-wise, we helped on the technical side by:

- Adapting the Aave tooling to support this multi-pool approach, including but not limited to aave-address-book, aave-proposals, aave-helpers, aave-permissions-list.
- Supporting other contributors like ACI or TokenLogic with advice on for example pool or rewards configurations.
- Answer any technical doubts to external partners involved in those pools, like Lido or EtherFi.

Even if there is room to improve tooling and procedure-wise for multi-network and multi-pool strategy going forward, we are confident that the process should be pretty scalable now.

Assets' Pricing: CAPO

On the pricing side of the Aave protocol, we have worked on the following:

- We proposed to apply CAPO for all stable coins on Aave v2, to be aligned with the strategy on v3 [HERE](#).
- We made different improvements and refactoring on the [CAPO repository](#), including adding a [guide for other contributors to add new CAPO instances to it](#).
- More on the listings side (described in a different section on this recap), we did pricing strategy recommendations for assets, mainly depending on their technical and security architecture.
- Given the frequency of listings of LSTs/LRTs and in general, assets tied to an underlying, we started researching new pricing strategies for these assets, with technologies with Chainlink PoR in mind.

2. Aave tooling and infrastructure

→ a.DI (Aave Delivery Infrastructure)

Since December 2023, a.DI has been in production fulfilling all cross-chain communication needs for the Aave governance. This has enabled all types of parameter changes on the 10 active Aave pools, proposals by treasury contributors to rebalance funds across networks, activations of new Aave pools, or even the activation of cross-chain GHO.

The main success metric on this project in our opinion is the lack of any type of operational/security incident: a.DI has worked as expected

Another very important aspect to highlight about a.DI, is the adoption by the Lido DAO of the system, as it can be seen [HERE](#). We see this as a good symptom of the quality of our contribution to Aave, when other very important actors of the ecosystem like Lido adopt independently Aave's technology just for its quality and suitability.

Nevertheless, even if a.DI is stable, we have made different improvements to it during this engagement:

- The biggest contribution has been the release of [a.DI v1.1](#), including 2 major features: a.DI Shuffling and granular access control. Shuffling improved the scalability and cost optimization of the system, while granular access control aligned better with the requirements of Aave for operational vs emergency actions, in a decentralized framework.
- We proposed an update of all a.DI adapters of “native” bridges (e.g. rollups) to add extra metadata (standardizing aspects like events across paths), or improve gas configurations, [HERE](#).
- We updated the LayerZero adapters to their v2, via the maintenance proposa[HERE](#).
- We updated the HyperLane adapters to their v3, via the maintenance proposa[HERE](#).
- We added support on a.DI for both Wormhole and Axelar. Currently pending to be added into the consensus of some a.DI paths, but already in production in a.DI Lido instance.
- As part of the pre-requirements for the activation of a new Aave pool, we enabled a.DI on ZKSync[HERE](#).
- We did an important refactoring/improvement of the a.DI deployment infrastructure ([adi-deploy](#)), to have it more decoupled from the a.DI codebase itself (e.g. using more create2).
- Refactor testing/visibility infrastructure (e.g. configurations diffs, standard payloads similar to governance proposals) for upgrades on the a.DI smart contracts in production via governance.
- On a.DI Monitoring (<https://adi.onaave.com/>):
- Added bridging times, by adapter.
- Burn rates of the system: transaction cost, bridging cost; with granularity per underlying provider.
- Adaptation to support a.DI v1.1, mainly its new Optimal Bandwidth configuration.
- Multiple internal optimisations.
- Added bridging times, by adapter.
- Burn rates of the system: transaction cost, bridging cost; with granularity per underlying provider.
- Adaptation to support a.DI v1.1, mainly its new Optimal Bandwidth configuration.
- Multiple internal optimisations.

Similar to Aave Governance v3, a.DI has reached quite some maturity, and the work going forward will be focused on quality improvements.

→ Aave permissions list

The work on Aave Permissions List has been on the following:

- Maintenance

: support the new networks Aave expands to and new pools, add interpretation of some extra smart contracts of the Aave ecosystem and miscellaneous internal refactoring.

- Contracts upgradeability

: we added a new table of each pool's report, to show which level which contracts of the ecosystem are upgradeable (and controlled by who) versus non-upgradeable. This is a useful way for both users and integrators to simply see the levels of trust deposited into each contract. Example [HERE](#).

Aave permissions list has become a pretty fundamental infrastructure piece for quality assurance of both the new Aave pool and generally any Aave governance proposals, in order to have visibility about any type of permissions change on both simulation (e.g. Tenderly) and pre-production (e.g. during voting) stages.

→ Aave Address Book

Aave Address Book (and its UI Search on Aave) has become one of the most used projects in the Aave ecosystem, being the “source of truth” for the official Aave smart contract addresses.

During this engagement, we did the following:

- Adapted the repository whenever necessary for compatibility with new Aave versions, like v3.1 or v3.2.

- Added automatic validations for contracts to be verified when added to Aave Address Book.
- Added automatic validations of cross-references: when an address is added, we verify that of for example getters in other contracts, the value is the same.
- Added support to Gnosis Safe address book, for users to be able to import a compatible format all Aave ecosystem addresses.
- Misc internal improvements whenever required.

→ **Aave Governance v3**

Aave Governance v3 has been working without any significant problems since its activation, and we have focused mainly on improving internally vote.onaave.com, the interface used by a good part of governance participants.

Amongst other, we have improved:

- Improving the logic handling data fetching from IPFS, handling better fallback strategies to improve stability when the main gateway has any problem.
- Add support to new networks like ZKSync.
- Internal refactoring of icons (assets, wallets, networks).
- General maintenance to keep dependencies updated.

On the smart contracts side, we have not made any improvement in order to prioritize other fields of contribution, given that Aave Governance v3 is already completely efficient and scalable.

→ **Safety Module**

On the Safety Module side, our focus has been on Umbrella, covered by a separate scope on the Aave <> BGD Phase 3, and that we will update about pretty soon.

The current pre-Umbrell system is running normally in production, and apart from supporting other contributors on aspects like changing rewards, no extra improvements have been made.

→ **Aave Robot**

On the Robot side, the work was mainly on the maintenance, optimization and coverage directions, with the following contributions:

- Update the whole Robot system to be compatible with Chainlink Automation v2.
- Improve gas management mechanics, introducing stricter gas price limits to only execute proposals whenever there are no gas spikes.

→ **Maintenance proposals**

During this period, we created 9 governance proposals via the maintenance proposals flow, generally complementary to other ongoing projects (e.g. a.DI, network deployments, Guardian-related).

All of them can be found on the Technical maintenance proposals thread, after [this one](#).

→ **Aave Seatbelt**

On Aave Seatbelt, in addition to all required maintenance and bug fixing, we have made the following improvements:

- Given that Certora now is an independent user of the tool, that allowed us to receive external feedback, and consequently apply it to the tool, like the interpretation of extra fields Aave-related.
- As it is quite frequent that the new network where Aave lives doesn't have Tenderly support (previously a hard dependency of Seatbelt), we have added a fallback mechanism of Foundry-based state snapshots and diffs, to show reports on all changes directly related to the Aave protocol, even without interpretation for external systems.

→ **Aave Proposals and Aave Helpers**

Aave Proposals and Aave Helpers are pretty complementary tools, and continuously evolve whenever external contributors

require improvements, or we need them while developing other projects (e.g. adding extra fork tests on certain governance proposals).

The focus during this engagement has been purely on maintenance with items like:

- Improving the proposals generator scripts.
- Improving the preview features when creating proposals.
- Improving the defaults tests applied on all governance proposals in fork, in the pre-creation stage.
- All sorts of minor changes and bug fixes.

3. Technical support to other contributors

→ Proposals reviews

During this engagement, we have supported all other contributors reviewing more than 100 governance proposals before they are created on-chain, a rhythm of 1 proposal every ~2 days.

Similar to previous engagements, the complexity of the proposal heavily varies depending on the content, but we are pretty glad about how both our expertise and the tooling we have built allow us to keep very high speed for the DAO in this field while having incredibly high-quality assurance.

In summary, every single proposal and new sub-system activated on Aave has been reviewed in some or multiple stages by BGD.

Additionally, and while keeping the required independence, we have supported Certora in their involvement in reviewing the same proposals on the on-chain stage, together with addressing any kind of doubt or problem arising from the proposals.

→ Tech support in rewards programs

Currently, the Aave DAO has 2 types of infrastructure to incentivize activity on its systems: the Aave v3 native incentives smart contracts, and the Merit program by ACI. Our technical support has been focused on the Aave v3 incentives smart contract side, by reviewing activations and maintaining the repository associated with the system [GitHub - bgd-labs/example-liquidity-mining-aave-v3: Example repo on how to enable and configure liquidity mining on aave v3](#), which allows for entities like ACI to activate/modify rewards programs in a pretty straightforward and safe manner.

→ GHO

On the GHO side, our focus has been to support other contributors like Aave Labs when the developments affect directly the Aave infrastructure (e.g. integrations with the Aave protocol), both in design and implementation phases.

The major contribution during this engagement was doing an extensive review/analysis of cross-chain GHO design and implementation before its release, confirming to the Aave community its suitability, together with giving precise feedback on aspects to change or improve before and after the activation.

The analysis can be found [HERE](#).

Additionally, we had given minor advice and support on other components like the GHO stewards used by the ALC (Aave Liquidity Committee).

→ Advisory on AAVEnomics

On July 25th, ACI proposed a new iteration of \$AAVE tokenomics [HERE](#). Even if until now not being a technical topic, we advise them on the initial feasibility of different aspects of the proposal, and especially on the synergies with Umbrella given the direct impact of the project on the current Aave Safety Module and \$stkAAVE/stkABPT.

After that, we contributed to the discussion with a more precise idea of integration between Umbrella and AAVEnomics: slashing hooks, described in detail [HERE](#).

4. Security & Technical analysis

→ Immunefi

Similar as in previous engagements, we have been receiving and addressing multiple bug bounty submissions on the Aave <> Immunefi program. Fortunately, no major report was received during this time, and only a [June Request for Bounty](#)

[Payout](#) and an upcoming one this month was needed.

Additionally, we have done the appropriate maintenance on the program whenever scope needed to be updated, or rules.

Even if doesn't belong to this recap, we will be proposing different optimizations to the bug bounty program going forward, given that this should be dynamic and always evolving.

→ Onboarding of new security reviewers

Even if more intangible than other items, especially for Aave v3.2 and Umbrella we have put focus on “scouting” and onboarding new security auditors, to increase the pool of them for Aave projects, while testing quality.

The results of this were quite satisfactory regarding Aave v3.2 (more details on the post of that specific project), which should benefit Aave in the medium and long term.

→ Cantina Aave v3.1 competition

As part of the Aave v3.1 security procedures, we evaluated, set up, supervised, and reviewed submissions of the [Aave v3.1 Cantina competition](#).

Being the first competition for the Aave protocol, we derived important learnings on how and when contests/competitions can play a role during security processes; apart from the obvious benefits the v3.1 one provided.

→ Participation in security incidents

- We led the analysis, protective actions, and resolution of the Aave v3 ZKSync activation issue [HERE](#).
- We initiated the analysis/reaction of the [Periphery Contracts Incident](#), and once the problem was more isolated, supported Aave Labs on everything required concerning the integration of the Aave periphery contracts on their UI.
- We evaluated several “false alarms” about problems potentially affecting Aave or its integrations (e.g. assets listed). Fortunately, none of them apart from the previously described were real threats.

→ Asset listings. Technical/security analysis

During this period, it has been quite frequent to have assets candidates for listing where its technical architecture and general security are even more important than usual for the Aave protocol. That is the case of for example LSTs, whose only responsible pricing strategy makes Aave very dependent on the asset's security, or others like tBTC, where the architecture of the custody and mint/burn flows play an enormous role in pricing them efficiently.

During Phase 3, we have contributed with the following analysis:

- [tBTC](#).
- [sfrxETH](#).
- [sUSDe](#).
- [ETHx](#).
- [osETH](#).
- [weETH](#).

→ Analysis of networks to activate Aave

On the network evaluation side, we have only published a report about ZKSync [HERE](#). The rationale for the reduced number of new networks is the following:

- Even if Aave is very scalable at the moment in terms of expansion to networks, generally we believe it is important to balance expansions with improving the quality and features of the product. In this scope, we have prioritized slightly more protocol improvements.
- We underestimated the effort required for ZKSync, which is a good lesson for ourselves and the community going forward: tooling compatibility and full stack equivalence between Ethereum and other networks can imply tremendous overhead, even when the team behind the network is allocating resources, like in the ZKSync case.
- The change of strategy to multi-pool in the same network in the order of 1-2 analyses to be slightly delayed, planned

for the following weeks. Even if this is not ideal, we think it is both a characteristic and strength of our scopes: if we see the community is demanding something with business urgency, we are able to swap focus between tasks, while keeping quality.

→ Review deployments by Catapulta

During this engagement, we reviewed two pool deployments by Catapulta (engaged for 6 months [HERE](#)): the Aave v3 Lido Ethereum and the Aave v3 Etherfi Ethereum pools.

Even if there was no problem with the collaboration or outcome, we detected a lack of optimality during the process, so our recommendation going forward is for us BGD to again proceed with these activations replacing Catapulta

, as it is complicated to decouple the system that we still need to deploy (for different networks) and our tooling, from the deployment and activation proposals themselves.

5. Umbrella

Being part of a separate scope and not strictly time-based contribution, we will do an update about Umbrella in a separate post.

6. Miscellaneous

- We followed and analyzed the MATIC to POL migration (some phases still on-going), [HERE](#).
- We provided all necessary technical feedback to ACI about the Aave Guardian Renewal (some steps still pending on governance), together with supporting the Aave Protocol Guardian in events like the ZKSync pause on the initial activation. [HERE](#).
- We made several improvements on [aave-cli](#), a tool/package used both by us internally and other contributors (e.g. Avara Labs) to for example create test fork environments with proposals executed that still running in production.
- Gave technical feedback to all types of contributors when requested, including ACI, Aave Labs, Chaos Labs, TokenLogic, Certora, Karpatkey, LlamaRisk, or Catapulta.
- Discussed with countless partners about all-Aave topics, from listings to technical integrations, passing through security, and any other aspect. This is complicated to present quantitatively, but our hope is that partners of Aave and the community know that for anything technical-related, BGD Labs is always available to help.

Conclusions and, next?

- On what we think is a good note, this engagement has been more of the same compared with others: making the current active systems work as expected, keep evolving them and still create new features and sub-systems.
- We have focused especially on Aave v3 core this time, and confirmed something very important going forward: Aave v3 is a powerful system, that combines a battle-tested codebase, with good flexibility to evolve it.
- The Aave DAO has become substantially more efficient, with multiple regular and independent contributors helping each other when needed, while keeping independence. This naturally has affected development flow, as from BGD we can reduce the support required by other parties.
- Documentation of projects and visibility is still a field we think we could improve. Even if we put important focus on both, we tend to prioritise execution and a system like Aave should be even better in that direction.
- Generally, we think the separation of a continuous scope versus project-based is suitable for our role, given how unpredictable are the needs on the first versus the clearer planning on the second (even if with only an approximate duration). This is akin to development projects in centralized organizations, but working in the context of a truly decentralized DAO, it is simply different.

We thank the whole community and all its service providers for making our work easier: [@ACI](#), [@ChaosLabs](#), [@Certora](#), [@TokenLogic](#), [@Karpatkey](#), [@AaveLabs](#), Catapulta ([@kartojal](#)), and [@LlamaRisk](#).

Also thanks to all governance participants who week by week made a tremendous effort (sometimes not so appreciated) to understand governance proposals, vote on them, and generally be involved in this forum: [@EzR3aL](#), [@ACI](#), [@TokenLogic](#), [@Karpatkey](#), Areta ([@sid_aret](#)), [@WintermuteGovernance](#), StableLab ([@Kene_StableLab](#)), Keyrock ([@0xkeyrock.eth](#)), [@SaucyBlock](#), [@DAOplomats.eth](#), [@Event_Horizon](#); and everybody else writing regularly or just putting some time following Aave.

Finally, thanks to all Aave and DeFi users. Everything this DAO and its contributors create is finally for you.

Just_Use_Aave

.

We are satisfied with the outcome of Aave <> Bored Ghosts Phase 3 continuous engagement, and we hope the community is likewise, even if we are open to any type of feedback, positive and/or negative.

We intend to keep working for the Aave DAO, so, we will be creating a proposal for Phase 4 (continuous) in the upcoming days

.