BGD. AAVE <> Celo.

INFRASTRUCTURE / TECHNICAL EVALUATION

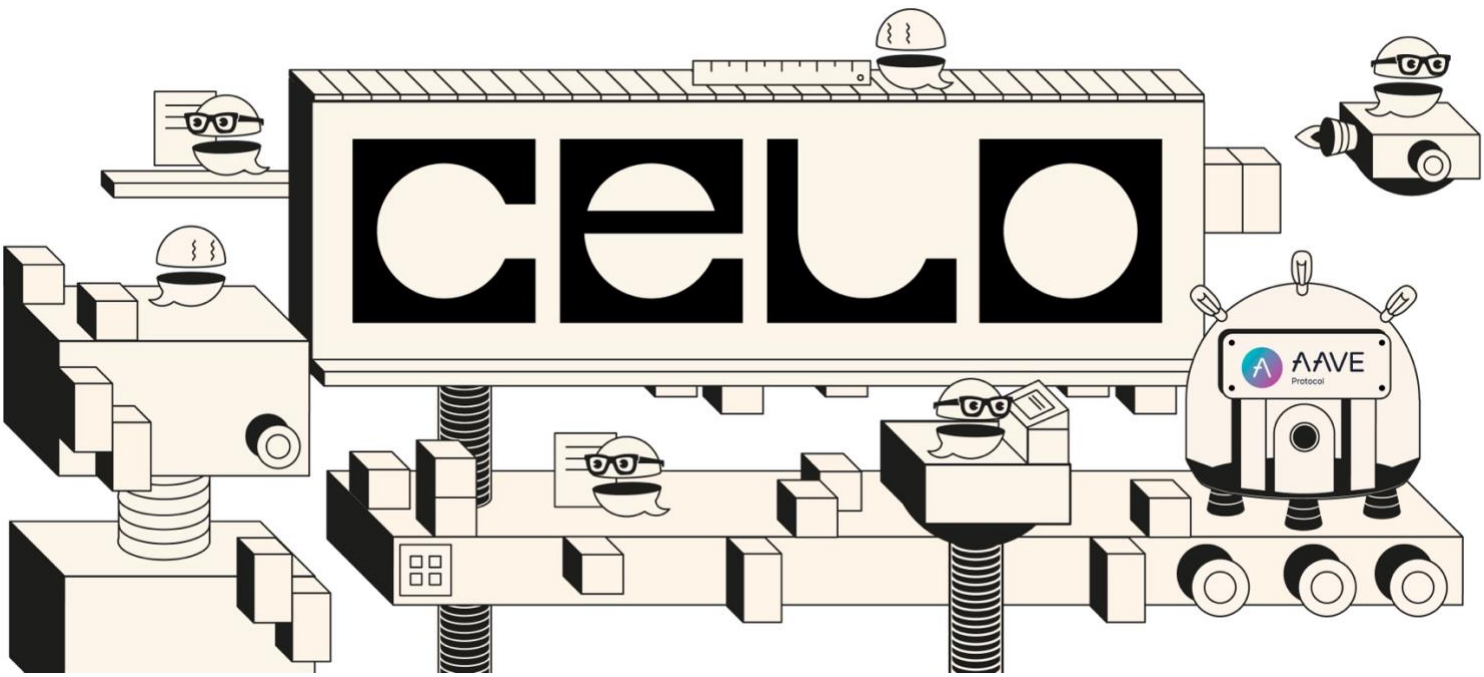


Table of Contents

Introduction

Following our framework of Aave's infrastructure evaluation and the positive outcome of the temperature check, we present the **analysis of Celo regarding its suitability to deploy an instance of the Aave protocol**.

Disclosure

This is an independent assessment of different technical components that we consider important for the Aave software to run optimally, not a categorical analysis stating the network is "good" or "bad", and no kind of "requirement" for Aave to be deployed on the candidate network. That decision is up to the Aave governance, no matter our opinion.

In addition, currently, we have absolutely no financial/investment/services-engagement/any kind of interest in the Celo ecosystem.

When doing the evaluation, we contacted the Celo team as an important source of information, which has always been exemplary and supportive, but everything in this report comes finally and exclusively from our independent criteria.

Report

1. Introduction to Celo

Celo positions itself as a blockchain system, with components based and/or inspired by Ethereum, but completely independent. Technically, it is an EVM-based layer one network, with PoS (Proof-of-Stake) consensus.

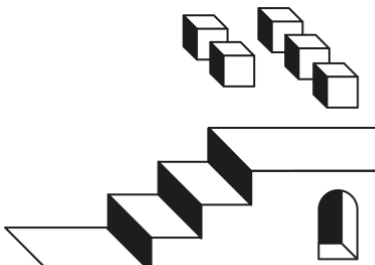
Celo uses the CELO token (allowing others too) both for gas payment and to participate in the consensus mechanism/governance of the network.

2. Our methodology

This report is not trying to be a full analysis of the Celo network but focuses on those aspects that would be important if the Aave community decides to deploy the Aave v3 liquidity protocol there.

In addition to being extensive enough, this report tries to be simple enough for all participants in the Aave governance to understand. However, given its technical nature, it is unavoidable to assume a certain familiarity with the basic concepts touched, like sidechains, oracles, RPC nodes, or blockchain explorers, amongst others.

In order to simplify the interpretation of this report, we will evaluate each component important for Aave separately, and assign simplified “grades”, defined as follows:



Optimal. Fulfilling all minimal requirements, and with extra positive aspects.

Good. Fulfilling the requirements, but improvements can be made.

Acceptable. Fulfilling the requirements, but with “buts”.

Needs improvement. Mandatory requirements not fulfilled at all. Aave will not work properly

3. Evaluation

3.1. Oracle Infrastructure

The Aave protocol uses 2 types of oracles in the case of a network like Celo: prices and PoR (Proof-of-Reserve). We consider that having Chainlink providing oracles, especially for the most important type (prices), is a must for any deployment, with alternatives only considered really ad-hoc, given the additional complexity added on evaluation/integration.

3.1.1. Price feeds

Used by the Aave protocol to price assets listed.

Celo **HAS** Chainlink price feeds for the majority of the assets, but we have noticed it doesn't for the native cUSD stablecoin, which represents an important percentage of the network's TVL.

3.1.2. Proof-of-Reserve feeds

Component indicating to the Aave protocol if the reserves backing bridged assets are healthy. In the case of Celo,

Celo **DOESN'T HAVE** Proof-of-Reserve feeds at the moment.

3.2. Blockchain explorer

For Aave, same as for any other blockchain project, block explorers like Etherscan are a fundamental component, specifically for the following:

1. Verifiable smart contracts code visualization.
2. Read/write interface with smart contracts.
3. Basic data analysis tool for misc aspects like token holders.

Celo has a white-labeled instance of Etherscan on https://celoscan.io/, together with a Blockscout based explorer https://explorer.celo.org/mainnet/.

3.3. Compatibility with Ethereum RPC standard

Basic compatibility with the Ethereum nodes RPC de-facto standard (eth_, *web3_*) is quite an important requirement for Aave or any other protocol, given that it helps to have tools built for Ethereum (or other similar networks) working out-of-the-box just by plugging them to node, of Celo in this case.

Celo **HAS** complete compatibility with go-ethereum. Depending on the node provider there can also be support for non-standard endpoints (e.g. trace_*, debug_traceTransaction).

3.4 Compatibility with Ethereum account format (addresses)

One of the strengths of non-Ethereum networks (e.g. Polygon, Avalanche C-Chain, etc.) is its compatibility with Ethereum private/public keys of accounts. This allows existing account holders on those networks to use the others without creating an ad-hoc wallet for it.

Celo is fully compatible with the Ethereum account format.

3.5. RPC public endpoints and providers

Basic and reliable public RPC infrastructure is a must for Aave, as it is the way to connect to the network, both for data reading and transaction submission.

From what we are aware, Celo has 2 types of RPC endpoints:

1. Forno, an “official” public RPC provided by cLabs. It is rate-limited and not recommended for production.
2. Third-party ones like Infura, Quicknode, Lava, or Ankr.

3.6. Custom behavior (lack of) of the execution layer

Whenever a network has custom/extended behavior with respect to Ethereum, it is important to be aware of it and evaluate if it has any impact on the Aave protocol.

Examples of this potential behavior are the presence of new pre-compiles (compared with Ethereum or similar rollups like Optimism), EVM opcodes, native account abstraction/meta-transactions, chainId definition out of the norm, etc.

Given the alignment with the Go-Ethereum node implementation (geth) we checked that:

- The chainId behavior is appropriate, with the id 42220 for Celo not clashing with any other.
- Apart from the ones present on Ethereum, Celo has additional pre-compiles that can be found here. We have verified they should not have any impact on Aave.
- The GASLIMIT and DIFFICULTY Ethereum opcodes are not available on Celo, but this should not have any impact on Aave.
- The CELO asset has a dual ERC20/native nature, which implies that balance changes (via transfers) can happen without ERC20 functions being triggered. We have verified that this should not have any impact on Aave.
- Celo allows payment of gas fees in non-CELO tokens, but it is also fully compatible with the Ethereum transaction format.

In summary, Celo has multiple custom components compared with Ethereum, but the compatibility is acceptable to host Aave.

3.7. Support of wallet providers

Wallet products like Metamask, Ledger, Coinbase Wallet, and others, are fundamental pieces of the infrastructure for users to access the Aave protocol. So it is a strong requirement for a network to be supported by a subset of them.

Given its EVM compatibility in the context of this document, Celo is transparently supported by the majority of all the chain-agnostic wallets, like Metamask, Ledger, Coinbase Wallet, or Frame.

Additionally, Celo has a native mobile wallet, focused on user experience, Valora.

3.8. On-chain multi-signature infrastructure

The permissions on the Aave ecosystem are directly held by on-chain governance smart contracts across all the networks.

However, different protection/emergency mechanisms, like a.DI cross-chain proposals, or freezing/pausing an Aave asset/pool, depend on the Aave Guardian, who is capable of acting faster than the governance process.

Consequently, having on-chain multi-signature contracts is a requisite to have Aave on a different network, with a high preference for industry-standard tools like Gnosis Safe.

Celo **HAS** an official instance of the Safe contracts on-chain, supported natively inside the official Safe interface.

3.9. Transactions simulation infrastructure (fork)

Lately, a really important development experience component is the ability to execute test transactions (simulations) on forked production networks.

A good part of the tooling around Aave depends on simulations by using different libraries/frameworks like Hardhat, Foundry, or Tenderly. This way, it is possible to rapidly prototype new developments, get extra assurances on governance proposals and protocol upgrades, change risk parameters, etc.

As it is our main smart contracts development framework, we have tested that it is possible to do fork simulations on Celo with Foundry. Given its EVM compatibility, it should be perfectly doable with Hardhat too.

Tenderly doesn't support Celo.

3.10. Chain data/indexing solutions

For different projects and entities integrating Aave, and even if not a blocker for deployment, it is important that solutions like TheGraph or Dune are operating on the candidate network, to avoid building from scratch data pipelines.

Celo is supported on Dune and TheGraph.

3.11. Bridging infrastructure: assets, messages

Given the central role of Ethereum in the DeFi and Aave ecosystems, proper bridging infrastructure to/from is a must for any candidate network.

There are 2 types of bridging affecting Aave: assets and generic messaging. In the case of Celo, the state of these 2 types is as follows:

- **Assets.** sub-use case of the generic messaging infrastructure focused on assets like ERC20 or ERC721. In the case of Celo, there is no so-called "canonical" bridge, but different third-party cross-chain communication providers, each one with their one bridged assets. A complete list can be found here https://docs.celo.org/protocol/bridge#token-bridges

  This is not ideal for a protocol like Aave, as it will create important liquidity fragmentation, and important trust in third parties. We don't recommend to use those assets that are 1) non-canonical or 2) non-native to Celo.

- **Generic messaging.** Same as with assets, Celo doesn't have a "canonical" cross-chain communication bridge. However, given that multiple third-party providers support Celo, the Aave a.DI infrastructure is compatible.

3.12. Commitment in security incidents

Having proper mechanisms and procedures to prevent and react to security incidents is something quite fundamental for any platform and application, and networks like Celo are no exception.

The Celo team will keep a private channel of communication with the assigned technical team of the Aave community (e.g. BGD), for any incident of update concerning the network and consequently, Aave on Celo.

Procedures (e.g. network and core smart contract updates) are quite well documented for Celo, which solidifies the approach to security.

3.13. Network security/technical model

At the core of any candidate network analysis are its morphology (which type of network it is) and security model (which parties are involved in the control over the network; decentralization degree).

Regarding its morphology/type, Celo is technically a layer one blockchain, very similar to Ethereum itself.

Regarding its security model, the approach of the Celo network is similar to other Proof-of-Stake blockchains:

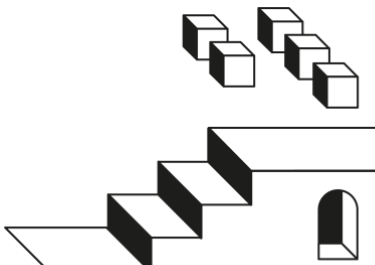
- There are validators (capped to 110 members at the moment), who lock CELO in order to verify the validity of blocks, and them to the blockchain. The cap is currently filled, with extra candidates ready to start validating.
- This validators can belong to so-called validator groups, which allow for holders of CELO tokens to delegate their voting power to them.
- Under certain circumstances, validators can be slashed, affecting their CELO stake.
- The network also has a pretty important layer of on-chain governance smart contracts, controlling core smart contracts like native contracts, and other basic Celo parameters. Similar as with validation of blocks, it is required to lock CELO in order to vote on the on-chain governance.

3.13.1. Centralization points

In one of the stages of the Celo on-chain governance procedure (Approval), a 3-of-9 multisig needs to approve submitted proposals to the governance contract, for them to proceed to voting.

The multi-sig can’t pass a proposal, so in practise the only permissions it has is veto power.

No core smart contracts have centralised points for our understanding, all are governance-controlled.



3.13.2. Upgradability strategy and sync with Ethereum

The Celo network has a pretty well-defined procedure for both network upgrades and core smart contracts, like the Governance.

- On smart contracts, the procedure involves different guidelines on how to update the code (checking backwards compatibility, storage layout conformance, etc), together with steps on how to submit governance proposals for the update. All the information about it can be found HERE.
- Network updates (blockchain client) follow a similar well-defined procedure, dependant on validator nodes updating their software after it gets published. Major upgrades are done approximately yearly, while minor usually quarterly.

  All the information about it can be found HERE.

Generally, procedures surrounding upgrades are public and quite well documented by our review.

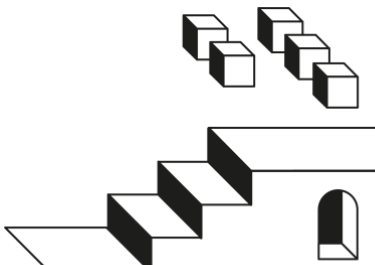
3.13.3. Security programs (e.g. bug bounty)

Celo has an on-going bug bounty campaign that can be found HERE.

3.13.4. Security audits

A list of the audits of Celo, can be found HERE.

An overview of all the infrastructural smart contracts of Celo can be found on the Celo official docs.



Summary


ORACLE INFRASTRUCTURE
ON-CHAIN MULTI-SIGNATURE INFRASTRUCTURE
BLOCKCHAIN EXPLORER
TRANSACTIONS SIMULATION INFRASTRUCTURE (FORK)
COMPATIBILITY WITH ETHEREUM RPC STANDARD
CHAIN DATA/INDEXING SOLUTIONS
COMPATIBILITY WITH ETHEREUM ACCOUNT FORMAT (ADDRESSES)
BRIDGING INFRASTRUCTURE: ASSETS, MESSAGES
RPC PUBLIC ENDPOINTS AND PROVIDERS
COMMITMENT TO SECURITY-INCIDENTS
CUSTOM BEHAVIOR (LACK OF) OF THE EXECUTION LAYER
NETWORK SECURITY/ TECHNICAL MODEL
SUPPORT OF WALLET PROVIDERS

From our analysis, we conclude that Celo, is an acceptable network candidate in regard to technical requirements, with no current hard blocker for the Aave v3 protocol to work properly. Our main consideration for this conclusion is the inherited infrastructure from Ethereum, with quite good maturity.

An expansion of Aave there will imply allocating some development resources for both the initial setup, together with some overhead of maintenance and monitoring over time, similar to other networks like Polygon.

Important points to evaluate by the community and risk providers are the following:

- **High-dependency on third-party bridge providers for assets**. We don't think Aave should list assets bridged by third-party bridge providers, as that would create high-complexity in the future if more providers appear. We suggest to only list native assets to the chain, or those where the team behind the asset is in charge of mint/burn or bridging.
- **Planned migration in the future**. Celo will transition to a roll-up in the future, but we got assurance that the hard-fork will be seamless for the networks, so we think it makes sense to proceed already with a deployment regarding this consideration.