This document is to be considered the source of truth for the latest version of the Stages Framework, which was first introduced here. A changelog can be found at the bottom of the document.

Specification

Stage 0 requirements

Does the project call itself a rollup?

To be considered a rollup, the project must self-identify as such. This requirement is straightforward and helps distinguish rollups from other scaling solution, such as Optimiums, Validiums or other types of bridges.

Are L2 state roots posted on L1?

Posting state roots on L1 is a key characteristic of rollups that allows for withdrawals. If a rollup does not post state roots on L1, it falls short of a fundamental component of a bridged rollup.

Does the project provide

Data Availability (DA) on L1

Ensuring data availability on L1 is essential for the security and reliability of a rollup. This means that all data necessary to reconstruct the L2 state must be available on L1, enhancing the system's transparency and auditability.

Is software capable of reconstructing the rollup's state source available?

A rollup node software capable of reconstructing the L2 state from L1 data should be available, contributing significantly to transparency and trust. This allows anyone to review, audit, and run the software, enabling users and external observers to independently validate the proposed state roots against published data.

Stage 1 requirements

Does the project use a proper proof system?

The proof system is used to adjudicate whether the proposed state root is correct or not. In the case of a fraud proof system, it allows invalid roots to be rejected. For zk rollups, the proof system is required to accept a proposed state root. If state diffs are used for data availability, the proof system must also ensure that all state changes are included in the diff.

Are there at least 5 external actors that can submit a fraud proof?

A fraud proof system requires at least one honest actor to verify the correctness of proposed state roots and potentially dispute them. For Stage 2 the proof system must be open to all participants, but for Stage 1 we allow an allowlist. The fraud proof system must allow a minimum of 5 external actors to perform this task.

Can the users

exit without the operator's coordination

?

The system should be designed so that user withdrawals cannot be blocked by the rollup operators. The rollup must implement mechanisms that allow users to exit independently, ensuring they can always access and control their assets.

Do users have at least 7 days to exit in case of unwanted upgrades (Security Council and governance excluded)?

This requirement is designed to protect users in the event of significant changes to the system, such as upgrades or modifications that they do not agree with. A minimum exit period of 7 days provides users with sufficient time to withdraw their assets and exit the system if they choose. At this stage, a Security Council and a governance system are permitted to act more swiftly. Note that a 7-day upgrade delay alone might not be sufficient: if any delay to withdraw is present (for example, a delay to force transactions in case of censoring operators, or a challenge period), it is subtracted from the exit window.

Is the Security Council properly set up?

The Security Council acts as a safeguard in the system, ready to step in in the event of bugs or issues with the proof system. It must function through a multisig setup consisting of at least 8 participants and require a 75% consensus threshold. Furthermore, the participants in the set need to be decentralized and diverse enough, possibly coming from different companies and jurisdictions. This setup ensures a diversity of viewpoints and minimizes the risk of any single party exerting undue influence. For the sake of transparency and accountability, the identities (or the pseudonyms) of the council

participants should also be publicly disclosed.

Stage 2 requirements

Is the fraud proof system permissionless?

In this stage, the fraud proof system should be fully decentralized and open to everyone. This means that anyone, not just a set of allowlisted actors, should be able to submit fraud proofs. This is a key requirement to ensure that the system is not controlled by a limited set of entities and instead is subject to the collective scrutiny of the entire community.

Do users have at least 30 days to exit in case of unwanted upgrades?

Users should be provided with at least 30 days to exit the system in case of unwanted upgrades, including upgrades initiated by a DAO. This ample time frame allows users to react to significant changes in the system that they may not agree with and withdraw their assets if needed. One exception that we make is given the existence of a onchain bug detection system (e.g. two valid contradicting zk proofs), instant upgrades are allowed for detected bugs.

Is the Security Council restricted to act only due to errors detected on chain?

In the final stage of rollup development, the power of the Security Council should be highly limited. It should only be able to intervene in the case of adjudicable onchain bugs, which are serious flaws in the system that could cause significant harm if not addressed. By restricting the council's actions to these types of errors, the system becomes more decentralized and the trust placed in the Security Council is reduced. This moves the rollup further towards the ideal of trust minimization, where the code itself is the ultimate authority. An example of this feature is present in the Polygon zkEVM contracts, where the rollup goes in "Emergency Mode" if two different valid proofs can be submitted using the same batches.

Changelog

- Dec 7, 2023: Updated the requirements for the Security Councils, detailed rationale here.
- Aug 25, 2023: Renamed Optimistic chains to Optimiums (PR).