The definition of enshrined rollups suggests to ZK-SNARK everything, including the execution layer. Here we propose an alternative design that launches a specific number (like 64) of rollups on the execution layer by using precompiled contracts. We call it native rollups, which will bear part of the advantages of enshrined rollups.

Precompiled Contracts & Rollup Slots

There are 64 pre-deployed contracts as “rollup slots,” which will be called directly by batch & proof transactions from rollups. These slots will call a precompiled contract for proof verification and update local state roots if successful. The precompiled contract can accelerate the verification of zero-knowledge proofs with optimizations in binary codes.

[

流程图

1167×649 54.5 KB

](https://ethresear.ch/uploads/default/original/2X/f/f69b6515f11e15d1cade29f052664bd0d45b66e0.jpeg)

Settlement Priority & Batch Reward

Batch & proof transactions successfully updating the state roots in rollup slots will be rewarded (to the block producer) with coins so that they will have a higher priority in the mempool and settle immediately. If not successful, they will be charged with gas, which is relatively low due to using the precompiled contract.