

# Comparing Layer-1 and Layer-2 Blockchain Protocols in Reputation-Based Networks

07-400 Milestone 4 Report

March 25, 2022

## 1 Major Changes

There have been no major changes since my last milestone report.

## 2 What I Have Accomplished Since My Last Meeting

I implemented query functions for the NodeSDK, or the shared interface for the various ledger backends being developed in my research group. These query functions include retrieving bolt instances held by a user, bolt specifications owned by user, and information about bolt specifications.

When implementing these query functions, we realized that the 32-byte public keys that defined bolt specifications and user wallets seemed unnecessarily long. Furthermore, in a Layer-1 protocol such as Solana, the amount of data stored impacts the cost of the smart contract. We hypothesized that mapping 4-byte unique identifiers to 32-byte public keys would be a much more efficient use of space, lowering costs without impacting query times too much. I have implemented the mapping between 4-byte unique identifiers and 32-byte public keys, and I have taken measurements with respect to cost and time for querying 10, 50, and 100 bolts, and for conducting transactions such as minting and transferring bolts.

## 3 Meeting Milestones

I have met my milestone for this week since I have have query functions to the ledger prototype that meet the API specifications and have profiled its performance.

## 4 Surprises

One surprise that I ran into when I was implementing 4-byte unique IDs was that Solana's blockchain does not support global variables. This is most likely due to the distributed nature of the system. Currently, I am generating the unique IDs on the client and passing it instruction data to the ledger when creating accounts.

## 5 Looking Ahead

In the next two weeks, I hope to make progress towards the common client interface, by revising the parts of my current prototype that do not fit with our current API. This might require making additional modifications to both the client and backend code. For instance, in order to move constants used in the client interface into my smart contract, I will need to implement cross-program invocation into for account creation into my ledger function that creates a bolt specification.

## 6 Revisions to Future Milestones

I am revising my next milestones as follows:

- **Milestone 5 (3/29):** Revise the ledger prototype to meet the API specifications for existing functions.

## 7 Resources Needed

No additional resources are needed at this time.