26-September-2016

HIP Identifier: Iroha HIP 0.1

Sponsors:
Makoto Takemiya, Soramitsu (takemiya@soramitsu.co.jp)
Toshiya Cho, Hitachi (toshiya.cho.bj@hitachi.com)
Takahiro Inaba, NTT Data (inabatk@nttdata.co.jp)

Abstract: It is proposed to accept the Soramitsu distributed ledger contribution into incubation status as "Iroha." Iroha is a distributed ledger project inspired by the Fabric API that aims to provide a development environment where C++ and mobile application developers to contribute to the Hyperledger Project. The project seeks to complement Fabric, Sawtooth Lake, and other potential projects, through interoperability protocols.

Context: The design and architecture of Iroha is greatly inspired by Fabric, in that membership, blockchain, and chaincode services form the overall architecture. Where possible, APIs were made to be similar to Fabric and, rather than competing with Fabric, the goals of Iroha are to: provide an environment for C++ developers to contribute to Hyperledger, provide infrastructure for mobile and web application support, and provide a framework to experiment with new APIs and consensus algorithms that could potentially be incorporated into Fabric in the future.

Motivation: Currently, the Hyperledger Project lacks an infrastructure project written in C++, thus limiting the potential developers who can contribute. Also, there is not currently a strong focus on user interaction or mobile applications, though both are necessary for the realization of the widespread use of distributed ledger technology. Iroha aims to rectify both of these points, bringing in more developers while providing libraries for mobile user interface development.

Iroha is a distributed ledger project that was designed to be simple and easy to incorporate into infrastructural projects requiring distributed ledger technology. Iroha features a:

- simple construction
- modern, domain-driven C++ design
- emphasis on mobile application development
- new, chain-based Byzantine fault tolerant consensus algorithm, called Sumeragi

Iroha was made to be simple and easy to use. Although Turing complete smart contracts are available via chaincode in Java (running a sandboxed JVM), users do not need to write chaincode in order to define digital assets in Iroha. Common use cases, such as deploying new currencies and sending text messages, are available as part of the core framework.

Iroha is composed of the following: