

Reworking HL Iroha Consensus API

Title	We will enable Iroha to easily use different consensus modules by creating a proper API.
Status	COMPLETED
Difficulty	MEDIUM

Description

In some of the projects this year we've decided to focus on modularity and decoupling which makes a lot of sense when we talk about open source and interoperability. In our chats we receive questions about decoupling different components – some of the users would like to plug different modules in and out.

To provide them with that option and make development easier and more secure, we would like to explore the option of following the example of the pluggable consensus in HL Sawtooth. In the best case scenario result, we will be able to reuse consensus modules from Sawtooth in Iroha and vice-versa in the future.

So, overall, this project will increase HL projects interoperability and make Iroha more flexible.

Additional Information

In HL Sawtooth, the consensus module is separated by a consensus endpoint: https://sawtooth.hyperledger.org/docs/core/nightly/1-2/sysadmin_guide/about_dynamic_consensus.html

HL Iroha currently includes a unique consensus algorithm called YAC: <https://arxiv.org/pdf/1809.00554.pdf>. It has 2 modes of operation: CFT and BFT, but the latter only under an assumption of non-malicious ordering service. These modes are controlled in the source code, which is not flexible.

Learning Objectives

- Architectural vision of a flexible blockchain platform
- Experience of integration of open source projects
- Research and comparison of different code bases
- Communicating with communities from different projects, maybe – collecting feedback from the community
- Writing documentation on pluggable components

Expected Outcome

The result of this project is a reworked source code of Iroha that provides a simple API for plugging different consensus modules found in other projects, along with some consensus modules that can work with Iroha.

Relation to Hyperledger

HL Iroha, HL Sawtooth

Education Level

See skills

Skills

C++, basic understanding of consensus in distributed systems.

Future plans

Seamless reusability of consensus providers across HL projects.

Preferred Hours and Length of Internship

Full time preferred, but part time also possible.

Mentors Names and Contact Info

Andrei Lebedev (andrei@soramitsu.co.jp, Telegram: lebdron),

Mikhail Boldyrev (boldrev@soramitsu.co.jp, Telegram: mboldyrev).