

Meeting Notes for 4/15/2019

Please add, or clarify any discussion topics that were missed or captured incorrectly

- Welcome (Mark)
- Submitted Grid RFCs
 - Product RFC for how to represent Products within Grid, GS1 products. If you are interested please take a look. (Shawn)
 - Seems to be product definition beyond what is provided EDI? (Clive)
 - It is based on how GS1 defines a product and the RFC also describes information about the smart contract and how it would be used in Sawtooth. (Shawn)
 - What is the prior art for Pike Transaction Family? (Clive)
 - We are using Pike as the initial identity mechanism, which defines Agents and Organization and their relationship. We will probably have an RFC for where we would like to take Pike in the future. (Shawn)
 - Primitives RFC for how to store and consume schemas (list, structs, etc) within Grid. Product will make use of these Schemas. If you are interested please take a look. (Shawn)
 - Grid Product: <https://github.com/hyperledger/grid-rfcs/pull/5/files>
 - Grid Primitives: <https://github.com/hyperledger/grid-rfcs/pull/4/files>
- Future RFCs?
 - Grid Components
 - Describes the components included in Grid such as Grid daemon, CLI, REST API and smart contracts. High level description of the Grid Repo layout. If you would like to collaborate on this RFC, please ping me. (Shawn)
 - Grid Product Catalog
 - RFC about how to organize products and share the list of products between organizations using Grid. If you would like to collaborate on this RFC, please ping me. (Shawn)
 - Consensource (Certificate registry prototype) <https://github.com/target/consensource>
 - This project was recently open sourced. Work will continue on it to allow it to work with Grid (Adeeb)
 - Consensource is also meant to have multiple entities participating (Matt)
 - If you would like to collaborate on a different RFC please let us know. We usually start it as a Google doc, where a small group of people can collaborate on it and then submit the RFC to the grid-rfc repo after it has been converted to Markdown. (Shawn)
- Updates on Grid Efforts
 - Andi Gunderson - Working on Grid Schema Smart Contract
 - Eloa & Shannyn Telander - Working on the Grid Rest API
 - Darian Plumb & Ryan Banks - Working on the Pike integration
 - Ben Betts - Working on Grid Documentation infrastructure
 - Peter Pschwarz - Working on Grid Schema RFC and Initial Grid Daemon work
 - Shawn - Working on Hyperledger Transact proposal, and it is planned that Grid will use Transact. Pike integration. Also been working on Diagrams to explain where Grid fits in relation to other Hyperledger projects.
 - Adeeb - Grid RFC for Product and working on the smart contract.
 - David Cecchi - IOT, track and trace sensor, RFC coming soon. GS1 has indicated that they are going to help review RFCs, especially the Product RFC.
 - Dan Middleton - Working on Privacy and Confidential issues.
- Open Forum
 - Grid allows for side-by-side contributions for industry specific use cases such as grid-contrib/track-and-trace. (Clive)
 - We have been working on the scaffolding of how Grid will work. For example, contracts usually have Track & Trace, but there is room for other contracts. The REST API and CLI will also have additions for the new contracts. We hope that it will be obvious on how to expand grid when new contracts are added. (Shawn)
 - When Track & Trace is in the context of Product, for example Coca Cola is a large industry of products, but will Track & Trace be open to other standards of track and trace? (Clive)
 - Product is very GS1 specific right now, Track and trace will be updated to use the same identifier as Grid Products, so in a UI you could look up an instance identifier but then pull up the metadata of the product definition. Track and Trace is very generic right now, we could even have several different implementations of Track and Trace within Grid. (Shawn)
 - For example, Grid could be a place for different implementations of Track & Trace. There are alternative views on how actions are processed, so it can be up to the developers to choose which implementation is right for their use case. (David Cecchi)
 - All pieces of Grid should play nicely together and be reusable. (Shawn)
 - Track-and-trace has domain specific transaction processors and validators. (Clive)
 - Track & Trace currently is a transaction processors but it will be run as a Sabre smart contract. All contracts will be compatible with Sabre. The vision is that if deployed on Sawtooth, all the contracts will be deployed onto the same network. The goal is to make Grid the only thing you need to run but for now you will need to start up Grid, which supports Track & Trace, Schema, Product, and the Sawtooth network for global state. We also want to make it possible to support features that allow you to pick specific pieces of Grid. For example, only schema and product within your custom Grid. (Shawn)
 - This approach seems to be going away from the modularity that can be seen in Sawtooth. (Stephen)
 - I see this more as an evaluation. We are able to run one transaction processor but still deploy whatever smart contracts are required for the network. We are making REST APIs, for example, one processes because having a ton of REST APIs running on the same network becomes hard to handle. (Shawn)
 - How easy is it to port Grid Track & Trace WASM smart contract into another WASM based Chain? (Stephen)
 - WASM does not provide an API for how to write a smart contract. But we have defined what the API looks like for Sabre smart contracts. This is portable between different WASM interpreters, but the chain would need to support Sabre Smart Contract API. (Shawn)
 - Additional industry specific use cases could also be contributed such as grid-contrib/insurance-certification (Clive)
 - Product currently supports GS1 but there is room for other implementation. And we are interested in any reusable components (Shawn).
 - To make Sawtooth identity private keys secure, WASM is best run inside PDOs (private data objects) on SGX based Intel processors. (Clive)
 - There is no plan currently to support this in grid. I am going to work with Dan Middleton on privacy over the next few months (Shawn).
 - WASM supports not only Rust but other cross compilation to javascript of other programming languages such as Java, Dart, C++ to in browser. (Clive)

- We absolutely want to support other languages, we need to find the ones that make sense based on the size of the byte code they produce, so probably not java but for sure C++. We need contributions (Shawn).
- Support of additional programming languages will help the adoption of Hyperledger Sawtooth and other modular Hyperledger projects. (Clive)
 - Absolutely. This is why we have different languages SDKs in Sawtooth (Shawn).
- There are many complex details to make cross-compilation to WASM resilient and robust. Sabre will draw on substrate by ParityTech to support WASM.
 - We have not talked about substrate. We need to look into. We are big fans of the ParityTech guys, as we use their WASM interpreter (Shawn).
- Discussion on developing Grid capabilities for providing visibility for analytics and administration could provide great value to developers and users. (Multiple)