Last edited: 2018, Dec 4

HIP Identifier: Hyperledger Grid

**Sponsors:**

David Cecchi, Cargill

Shawn Amundson, Bitwise IO

Dan Middleton, Intel

**Abstract:**

Grid is a WebAssembly-based project for building supply chain solutions. It includes a set of libraries, data models, and SDK to accelerate development for supply chain smart contracts and client interfaces. Supply chain is one of the most promising areas for blockchain use cases, and implementation of many of these use cases can benefit from capabilities which can be provided by a shared platform. This project will accelerate the development of blockchain-based solutions to cross-industry supply chain problems.

**Motivation:**

We have many frameworks and technologies at hyperledger but few practical demonstrations of how these technologies can be used. Moreover, the interfaces for the frameworks are very low level so as to be flexible, but that also makes them difficult to use. This project will serve two needs. First it will provide a valuable and more accessible functionality for distributed supply chain usages. Second it will demonstrate in an authentic way how to make use of multiple hyperledger technical ingredients in the same solution stack.

**Context:**

Blockchain solutions are primarily useful as a mechanism to solve problems which require sharing a database between multiple parties (or companies) which have an adversarial trust relationship. Such relationships are commonly found in supply chain ecosystems, because many companies are working in combination but do not necessarily trust each other fully (and in some cases, are direct competitors). Despite the potentially adversarial relationships, companies derive mutual benefit from working together -- including solving business problems with technical solutions (such as blockchain).

Some of the code developed for this project was originally a Hyperledger Sawtooth example focused on traceability, but has since grown into a body of libraries and data specifications which can be used when implementing smart contracts and application components (client apps, web interfaces, reporting databases and transaction event monitors, etc.). The [code repository](https://github.com/hyperledger/sawtooth-supply-chain) includes two example applications, one based on fish traceability and the other based on asset tracking. Additionally, [Pike](https://github.com/hyperledger/sawtooth-sabre/blob/master/contracts/sawtooth-pike/docs/source/pike_transaction_family.rst) will become part of this project; today, Pike allows management of identity concepts related to per-organization smart permissioning code (deployed on-chain and executed in a WASM engine) -- an advanced feature which goes beyond traditional role-based access permissioning.

**Scope:**

Supply chain is a broad topic, and as such, it is helpful to understand the initial focus for this particular project. This project aims to facilitate the use of Hyperledger technologies for supply chain use cases by providing higher-level abstractions and reusable building blocks specifically suited toward these use cases.

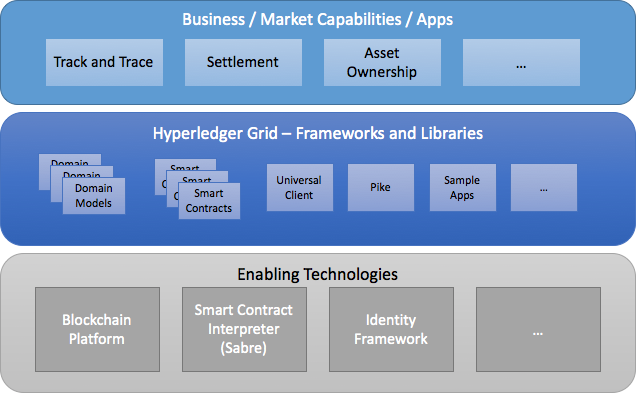
1. Higher level abstractions than conventionally provided by the lower-level blockchain platform stacks. Specifically supply chain-specific data types, data models, and standards as described in the solution section below.
2. Distributed problems related to multi-party interactions, i.e. not as a replacement for internal company systems. A supply chain solution requires integration with internal company systems and processes that do not relate to multi-party interactions - this type of functionality is out of scope for this project. However smart contracts that track, for example, the aggregation of components from multiple organizations are in scope.
3. *Compositions of features required by supply chain usages. As described in the solution section below, this project will prescribe certain types and operations such that there is consistent representation through the system and that operations can be composed. For example, a track and trace operation may also exchange ownership of a product using a smart contract defined in a product-specific namespace while relying on an identity described in an Indy-based namespace.*

*The supply chain project is not an application. It is not a turn-key solution. This project will, however, accelerate the development of blockchain-based solutions to cross-industry supply chain problems.*

**Proposed Status:** Incubation

**Solution:**

Supply Chain is a solution-driven project, not a blockchain implementation. This means that blockchain is an enabler rather than the driver of the project. Supply Chain will be a multi-layered ecosystem of technologies and frameworks that are able to work together. Users will select only the relevant components for their industry or deployment model. The initial basis for this interoperation is in the smart contract layer for the existing track and trace code which has been implemented in Rust/WASM.



Components are intended to be reusable but also prescriptive. By prescriptive we mean that this architecture will require certain formats or patterns. By contrast, platforms like Sawtooth do not bias development towards a particular language, serialization mechanism, etc. For example, the track and trace application should be adoptable by any industry, but the pattern for interaction with underlying blockchain persistence is fixed. The latter assists usability by not creating too many esoteric options for users and developers to divine.

Furthermore DLTs typically provide abstract key:value storage and leave asset models to the definition of developers. This is another area where this project will provide higher level assistance to app developers. Supply chains have common composition and traceability patterns that should be enabled by asset classes provided by the library.

Expanding on the Shared code and domain models in the diagram above:

Fundamental Data Type Primitives

* Designed for blockchain (e.g. deterministically serializable and computable)
  + Booleans
  + Enums
  + Structs
  + Fixed precision numerics (in lieu of floating points)
  + See [extensible data types RFC](https://github.com/hyperledger/sawtooth-rfcs/blob/master/text/0013-supply-chain-expand-data-types.md) linked earlier.

Data Models

* Domain Nouns
  + Organization
  + Product
  + Location
* Compliant with industry spec
  + (e.g. [GS1](https://www.gs1.org/sites/default/files/docs/traceability/GS1_Global_Traceability_Standard_i2.pdf))
  + See *Related Work* section for more resources
* Implemented for blockchain
* Property Schemas in conjunction with Universal Client (see below)
  + Schemas are recorded on-chain
  + Schemas are composable
  + Schemas rely on Data Type Primitives referenced above
  + See [Property Schema Definition RFC](https://github.com/hyperledger/sawtooth-rfcs/blob/master/text/0015-supply-chain-property-references.md) for further details.

Identity Models

* Leveraging Hyperledger Indy
* Compliant with industry spec
* Integrated with Pike, Smart Permissions.

Universal Supply Chain Client

* Abstracts UI over asset types
* Accelerates development of user interfaces.
* See [Universal Supply Chain Client RFC](https://github.com/hyperledger/sawtooth-rfcs/blob/master/text/0014-supply-chain-universal-client.md)

Framework Process Contracts

* Implemented for Sabre / Web Assembly
* Thing exchange / transfer
* Thing location update
* Etc.

**Effort and resources:**

The base supply chain application from Sawtooth is fully functional from a configurable user interface down to blockchain state encoding routines. The existing capabilities are centered around track and trace usages. This project will expand that functionality to include other domains of supply chain usage such as integrating the identity framework. Currently we are targeting Hyperledger Indy for this purpose.

The sponsor companies will provide resources to refine use cases, development, testing and field trial. Additionally, as sponsors develop projects based on Supply Chain, there is an ongoing opportunity to contribute back the field tested and refined code.

**How to:** We propose the creation of the following repositories on GitHub to manage Hyperledger Supply Chain resources:

· github.com/hyperledger/grid

**Frequently Asked Questions:**

1. Cross-platform integration: The proposed solution does not explicitly prohibit nor go out of its way to prevent the use of other platforms for enabling technology. From a practical standpoint, there would be expectations on any platform that seeks to support/enable this solution, most notably that it would need to support Sabre.
2. Is this an application?: No. For the people in the back: no. This is a collection of tools, frameworks and libraries that help accelerate the development of applications. As an example, supply chain use cases are typically anchoring on industry standard/adopted data standards. Implementing those data models / standards in open ways via this project will lower the barrier for people to build apps using those standards. Why should anyone who wants to build a solution have to define primitive data types? Or re-implement a GS1 Product data model? There is an opportunity to establish a home for these reusable components to bring meaningful quality and velocity to the apps and products ultimately being built.
3. This seems similar to Composer. How is it different?: They have similar objectives but the mechanism is different. This project doesn’t seek to create a code generation tool. Instead it seeks to provide more useful libraries and constructions that specifically enable supply chain usages.
4. How does the scope of this project compare to that of Indy?: Both solve domain problems that are not solved by the general purpose ledger frameworks. These problems are also widely applicable to different applications.
5. Could you use this for something else besides supply chain? It certainly would be possible to use these enabling capabilities for other domains / subjects outside of supply chain but this project will be grounded in supply chain use cases and standards.

**Related Work:**

GDSN <https://www.gs1.org/services/gdsn>

United Nations Standard Products and Services Code (r) <https://www.unspsc.org/>

Global Trade Item Number <https://www.gtin.info/>

Open Data Initiative<http://opendatatoolkit.worldbank.org/en/starting.html>

Identification of Medicinal Products<https://www.idmp1.com/>

ISO 9001 Quality Management Systems <https://www.iso.org/publication/PUB100304.html>

ISO 14001 Environmental Management Standards

<https://www.iso.org/publication/PUB100411.html>

Hyperledger Architecture Workgroup, Privacy and Confidentiality Sub Working Group report on Supply Chain

<https://docs.google.com/document/d/1-WVtWEh5HKX9Lmbl7PMl5axHrd2uXwIPVZkeGfsHY3E/edit#heading=h.1vd1kqbu6t5j>