



Welcome !



Fabric Application Developer Community Call

Thurs September 3rd 2020



Agenda



Today's Agenda :

- **Fabric Samples: Enhancements, and sample demo** Paul O'M (IBM)
 - <https://github.com/hyperledger/fabric-samples/#asset-transfer-samples-and-tutorials>
 - <https://github.com/mahoney1/fabric-samples/tree/master/commercial-paper>
- AOB

Note: Any resource links usually shared on:

<https://wiki.hyperledger.org/display/fabric/Resources%3A+Fabric+App+Developer+Community>

Improvements for Commercial Paper (1/3)

- UPDATE README DOC
 - Clearer Intro, Objectives section, why Blockchain and key takeaways:
 - *State* changes (asset lifecycle) vs Transaction *inputs* (asset interaction)
 - Simple collapsible explainers (why CP, and attractions as an investment strategy)
 - Position main docs (readthedocs) for use case and 'detail' : eg. modelling, logical/physical representation, analysis, use of namespaces (akin to registries), use of composite keys (juxtaposed other samples etc)
 - Overview diagram of the 'network' perspective.
 - Fix items passed from Community/Feedback(auth check, state mgmt, duplicate CCs, biz scenario info) –
 - https://docs.google.com/presentation/d/1UxK2HH8SrQyZU58MnuDb9hr1nmekst8b/edit#slide=id.g8471010e2b_0_7
 - Enable users to see the transaction *inputs* already committed using a v2 listener (as each txn is executed)
- ADD QUERIES
 - Add some standard queries:
 - History (asset *history*, with meaningful paper status (eg 'REDEEMED', 'TRADING' etc etc)
 - Ownership (papers by *owner*)
 - Issuer Listing (Partial Key query)
 - Named queries: Criteria-based (*Papers >= \$value, All Papers == 'redeemed'*), etc
 - Ad-Hoc capability (supply Mango yourself for custom query).

Improvements for Commercial Paper (2/3)

- FIXES
 - See next page
- **OVERALL:**
 - Commercial Paper (the sample) needs a better overview (aka “makeover”),
 - Explain how it is simply a financial asset being traded, why it is attractive to investors and why it’s a good use case for ‘blockchain’.
 - Clarity for readers that this is a sample implementation, taken from a structured use case covered elsewhere.
 - Implement fixes and queries (next page)
 - Add diagrams
- ACTIONS:
 - Simple Intro (DONE)
 - Overview diagram of the network, typical actors – collapsible (DONE)
 - What CP is, why its attractive (as investment strategy) – collapsible (DONE)
 - What the scenario is (DONE)
 - Fixes (DONE) – just Node.JS at this time (contract and app)
 - Queries (DONE) - – just Node.JS at this time (contract and app)
 - Add Indexes (TBA)
 - Add Info to ReadtheDocs (TBA) about additions

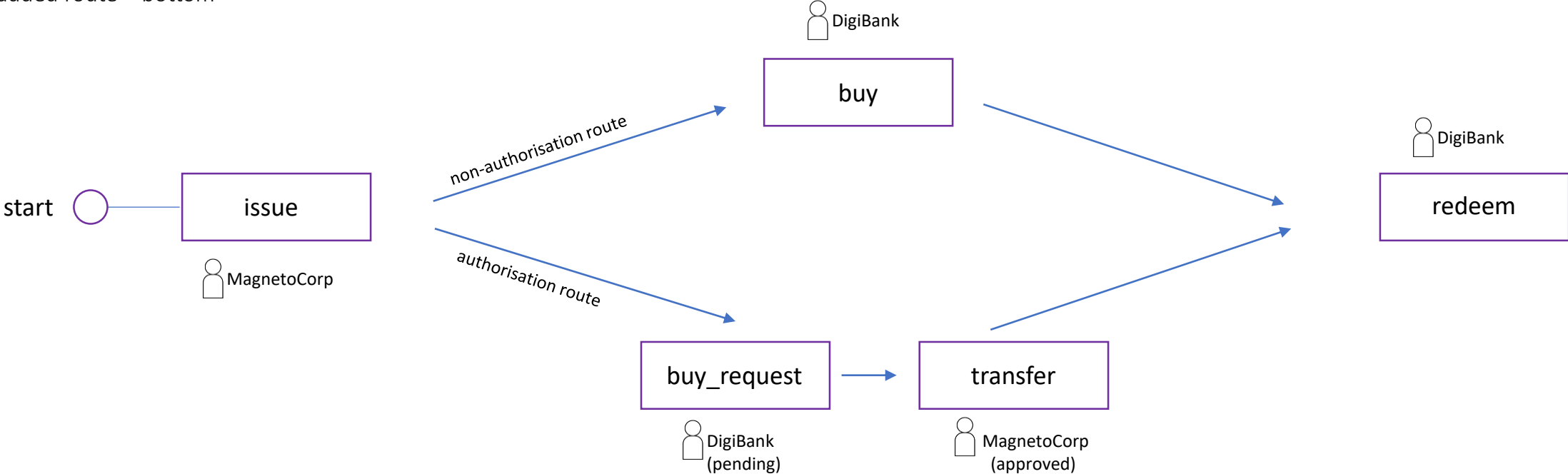
Improvements for Commercial Paper (3/3)

- FIXES
 - The key structure (latter part) has superfluous stringification and confusing if examining the ledger, querying, or looking at CouchDB documents ; I have got a fix for this (DONE)
 - The Commercial paper list namespace is implemented in the sample code as
-> `'org.papernet.commercialpaperlist'` ; but its logical/physical representation should match
-> `'org.papernet.papers'` (and be congruent with 'readthedocs' commentary) (DONE)
 - There is a field called `'key'` written to the ledger (`'Issuer:papernumber'`) – in addition to the real, namespaced composite key – the real intention of this `'key'` is in memory-processing, not - to be written to the ledger of each commercial paper record (confusing, we already have a key). (DONE)
 - Number fields should be written - as numbers 😊 - better for queries.
 - Removal of unused utility functions
 - Add comments showing 'where' STATE changes are taking place, just for readability. (TBA)

Commercial Paper Lifecycle:

Transaction Alternatives and Actors

existing route – top;
added route – bottom



Community Chat



- **Developer Chat (RocketChat) – channel names for posting:**

<https://chat.hyperledger.org/channel/>

#fabric-chaincode-dev

#fabric-java-chaincode

#fabric-sdk-node

#fabric-sdk-java

#fabric-sdk-go

Community Support



- **Questions & Open Community Support (Stack Overflow, Mailing List, Twitter)**

Contracts: <http://stackoverflow.com/questions/tagged/hyperledger-fabric>
<http://stackoverflow.com/questions/tagged/hyperledger-chaincode>

SDKs: <http://stackoverflow.com/questions/tagged/hyperledger-fabric-sdk-js>
(etc ie add 'language suffix')

Twitter: <https://twitter.com/hyperledger>

Mailing List: details next page

Application Developer Community Support (Mailing List)



- **Hyperledger Mailing List**

To subscribe or unsubscribe, visit
<https://lists.hyperledger.org/g/fabric>

or, via email, send a message with subject or body 'help' to
fabric@lists.hyperledger.org

Lists (Subgroups): <https://lists.hyperledger.org/g/main>

Help: fabric+help@lists.hyperledger.org

You can change your settings once you log in at
<https://lists.hyperledger.org/g/main>

Further Links



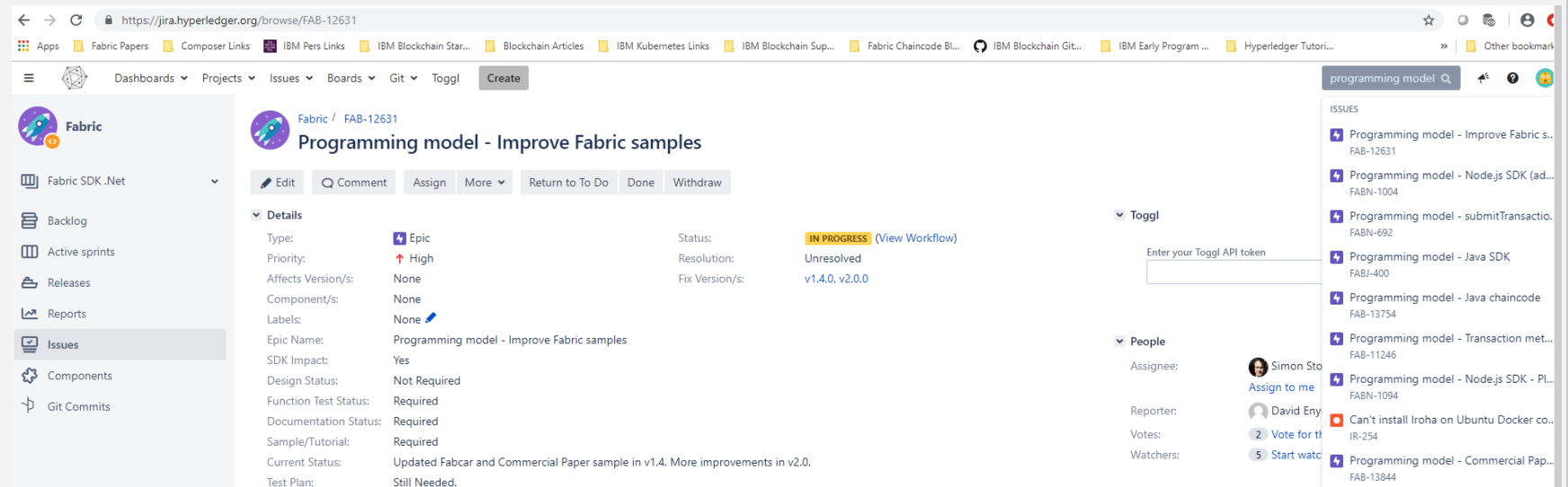
- **Fabric Developer - Community Home Page**
<https://wiki.hyperledger.org/display/fabric/Fabric+Application+Developer+Community+Calls>
- **Fabric Developer Community – resources**
<https://wiki.hyperledger.org/display/fabric/Resources%3A+Fabric+App+Developer+Community>

- Developing Applications using Hyperledger Fabric** (using the new programming model)
https://hyperledger-fabric.readthedocs.io/en/latest/developapps/developing_applications.html
- Fabric Application Developer samples (Contracts, SDK, Tutorials):**
 - **Fabric samples Repo (eg Commercial Paper, Fabcar):** : <https://github.com/hyperledger/fabric-samples/tree/release-1.4/>
 - **'Animal Tracking' TypeScript sample Contract/App Client (1.4)** <https://github.com/mahoney1/animaltracking> **with**
 - **Tutorial:** <https://github.com/mahoney1/docs/blob/master/animaltracking-tutorial.md>
- Go developer API (prototype) example** - (see README): <https://github.com/awjh-ibm/fabric-go-developer-api>
- Resources, tips and best practices** to share with the community
<https://github.com/ampretia/fabric-application-examples>

Fabric Team encourages you to review/comment on current Fabric JIRAs (ie stories, epics, requests relating to work-in-progress)

Eg. JIRA search - 'programming model' – (results below)

JIRAs – feel free to comment, give your input



The screenshot shows a JIRA issue page for 'Programming model - Improve Fabric samples' (FAB-12631). The issue is categorized as an Epic with a High priority. The status is 'IN PROGRESS' and it is currently 'Unresolved'. The fix versions are v1.4.0 and v2.0.0. The issue description includes a current status update: 'Updated Fabcar and Commercial Paper sample in v1.4. More improvements in v2.0.' and a test plan status of 'Still Needed.'.

The right sidebar shows a search for 'programming model' with a list of related issues:

- Programming model - Improve Fabric s... FAB-12631
- Programming model - Node.js SDK (ad... FABN-1004
- Programming model - submitTransactio... FABN-692
- Programming model - Java SDK FABJ-400
- Programming model - Java chaincode FAB-13754
- Programming model - Transaction met... FAB-11246
- Programming model - Node.js SDK - Pl... FABN-1094
- Can't install Iroha on Ubuntu Docker co... IR-254
- Programming model - Commercial Pap... FAB-13844

Reference



New Programming Model (1.4.x) :

- JIRA references - Fabric Programming Model Info:
 - <https://jira.hyperledger.org/projects/FABN/issues/FABN-692>
 - <https://jira.hyperledger.org/browse/FAB-11246>

THANKS!



CIAO FOR NOW!

