

Work Session: did:indy DID Method Spec-athon

Thursday 22J

Convener: Stephen Curran

Notes-taker(s): Stephen Curran

Tags for the session - technology discussed/ideas considered:

Topics Covered: Moving forward the specification of the new “did:indy” DID Method.

Discussion notes, key understandings, outstanding questions, observations, and, if appropriate to this discussion: action items, next steps:

- Presentation: <http://bit.ly/DIDIndyIIWXXXI>
- DID Method Draft: <https://hackmd.io/@icZC4epNSnqBbYE0hJYseA/S1eUS2BQw>

Notes collecting from the session as we went through the topics:

Documenting the DID Method -- ReSpec or SpecUp? No decision, we’ll experiment with both and select one. For now, we’ll work on the hackmd.io document.

Defining the <network> element of the DID Doc:

- Initial: like to use the first 5 characters of the hash of the genesis file
 - Problem: over time, the genesis file (which is inappropriately named) will change as it must contain IP and ports of nodes of the network. Over time, the original nodes will no doubt disappear.
 - Want to be able to fork a network and have a new network ID for the forked network.
- Other ideas:
 - (Somehow) Add hash of previous genesis file version to new version to create a chain.
 - Add hash of initial genesis file as a transaction to the config ledger to enable verifying the ledger.
 - Look into what the Trustbloc is doing, including the use of an alias name for the ledger.

Finding the ledger from a DID URI:

- No consensus was found, and not a lot of time was spent discussing.
- From the Indy Interop-athon, the idea was to just use configuration vs. some discovery mechanism embedded in the DID. That configuration could be stored by each agent, or a shared service could be developed to enable lookup from DID to ledger location (which is really genesis file location).

Supporting DIDs and DIDDocs using Indy NYMs and ATTRIBS

- Basic consensus reached on how to do this -- the details to be defined.
- Use Indy config and permissions to manage creation of NYM and ATTRIB
- NYM contains the local network identifier for the DID and a verkey to enable control over updating the NYM and adding/updating the ATTRIB.
- An ATTRIB of type DIDDoc contains the entire DIDDoc for the DID.
- If there is no ATTRIB of type DIDDoc, the NYM and verkey must be processed much like did:key handling to produce a DIDDoc by a resolver. Details of “much like” to be specified.
- If there is an ATTRIB of type DIDDoc, the latest version of that is to be returned as the DIDDoc by a resolver.
- The verkey of the NYM is NOT implicitly part of the DIDDoc. If the controller of the DID wants it in the DIDDoc, it must add it to the DIDDoc before writing the ATTRIB of type DIDDoc.

Specifying versions of DIDDocs

- The existing TxnID generated by Indy for the DIDDoc ATTRIB (or the NYM if there is no DIDDoc ATTRIB) will be used as the version-id as documented by the DIDDoc.
- TBD is whether a resolver (and what resolver -- internal or external to indy-node) will supplement the DIDDoc with the TxnID of the DIDDoc on resolution.

Supporting version-id and version-time

- Indexing of TxnID is already being done by Indy, so returning a DID by version-id using TxnID as the version is already supported.
- Indexing of Txns by time and related NYMs will need to be added to Indy to support version-time resolution support.

Supporting ledger objects as DIDs

- The DID Spec may or may not support “type” as an attribute of a DID
- Other Indy ledger objects should be able to be supported by adding an @context to the DID Document for the type, and adding the details about the object into the DID object.
- Indy-node processing may be needed if specially ledger-leveling handling of ledger objects is needed -- e.g. the ledger “knowing” that an object is a “schema” and hence handled in some special way. It’s not clear that is needed.

Next Steps

- It was agreed that the next discussion of the Indy DID Method would occur at the next Hyperledger Indy Contributors meeting and at that time, a regular meeting to evolve the spec would be planned and contributors added.