

2021-02-23: Indy DID Method Specification Call

Summary

- Revisiting/Finalizing points from last two meetings (state proofs, agreement keys, DIDDoc content)
- Resolving ledger objects as DIDs?

Recording from the call: [20210223 Indy DID Method Specification Call Recording](#)

Antitrust Policy Notice

Linux Foundation meetings involve participation by industry competitors, and it is the intention of the Linux Foundation to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

Examples of types of actions that are prohibited at Linux Foundation meetings and in connection with Linux Foundation activities are described in the Linux Foundation Antitrust Policy available at <http://www.linuxfoundation.org/antitrust-policy>. If you have questions about these matters, please contact your company counsel, or if you are a member of the Linux Foundation, feel free to contact Andrew Updegrave of the firm of Gesmer Updegrave LLP, which provides legal counsel to the Linux Foundation.





Hyperledger is committed to creating a safe community for all. For more information please visit the [Hyperledger Code of Conduct](#)

Welcome and Introductions

Announcements

Attendees

- [Stephen Curran](#)
- [Dan Gisolfi](#)
- [Paul Bastian](#)

Collaboration Channels

- Current hackmd [document](#)
- indy-did-method on RocketChat - <https://chat.hyperledger.org/channel/indy-did-method>
- [indy-did-method](#) repo: ~~ReSpec~~ vs. [SpecUp](#)

Agreed Upon:

- See HackMD Document for most of what we have discussed

Online Discussion (from RocketChat this week)

- Further discussion of State Proofs and what should the transaction be to get back a DIDDoc
- Request for input on the "nymKeyAgreement" flag. Not much in the response.

This Week's Discussion:

- From last week's [discussion](#) – confirmation of approach:
 - Have a single call that gets the NYM and ATTRIB and state proofs for both with same multi-signed merkle root, and leave the assembly to the client.
 - With support for the version_id and version_time resolution parameters.
 - Other options rejected
 - Do nothing
 - Single call with both state proofs and ledger does assembly
 - Write DIDDoc on ledger writes and have state proof for DIDDoc
 - Add WRITE_DIDDOC transaction

- Decision on assembly points of concern:
 - The "nymKeyAgreement" flag
 - Remove the "content" item and just inline the rest of the ATTRIB?
- What error checking should be done on the ATTRIB. See [proposal in the hackmd document](#) (valid JSON, no NYM fields, valid DIDDoc)
- Other Indy Ledger Objects as DIDs (e.g. schema, etc):
 - Drummonds proposal: https://docs.google.com/document/d/1f99z4Bf8F-DA8EbkNjInVQv2ncJGPwF_Dp7I7Mfyhqs/edit
 - Existing transactions

Future Discussions:

- What format for other objects on the ledger as DIDDocs.
- Handling existing objects on the ledger as DIDDocs?
- DNR and DND discussions
 - To find networks we will require at least the first and perhaps the second of these approaches, while the rest are suggested:
 - Config files for one or more known networks
 - A mechanism for a ledger operator to register discovery information for other ledgers (aka "human gossip")
 - A DID/DIDDoc on a ledger will contain cross-registry information
 - A mechanism is needed for finding the DID(s) that contain the registrations – ideas have been put forward - a DID Name Directory (DND) is the likely approach.
 - [Document](#) about the DND and DNR records
 - Decentralized registries based on verifiable credentials
 - Other registry mechanisms, such as the DDNR proposal
 - The DID Method Spec will include a reference to a repo (likely) "indy-did-networks" within Hyperledger that will be a lightly managed, structured repository of folders per Indy network with at least the config file(s) for the networks. Use of the repo is voluntary, but provides a convenient way for networks to publish information about the network. Maintainers will be selected from the community and should exhibit a light hand in accepting PRs, being concerned mainly with structure of the data (not content) and that contributors are not being malicious about updating the information of other network operators. The Hyperledger governance structure may be used for disputes as appropriate. This is not a replacement for the Governance that a specific network should implement.