OIDL-0004 - openIDL Setup and Management Documentation

We Document all levels of the openIDL network setup and management, starting with the openIDL wiki and migrate to ReadTheDocs when ready.

We document the Infrastructure as Code / Setup of the Network using the Wiki (in the short term)

Context and Problem Statement

The openIDL network is a complex network of nodes connected using the Hyperledger Fabric Distributed Ledger Technology.

On top of the HLF, the nodes require application code to interact with ledger.

There are three layers of the architecture:

- Cloud infrastructure - whether in AWS, Azure, GCP or some other, this is the set of virtual machines and networking necessary to run the software.
- Hyper Ledger Fabric - this is the kubernetes assets that support the DLT. This also includes the management console.
- Application - this is the application code that supports the business cases such as stat reporting or data calls.

Decision Drivers

- The different levels of setup are automated as much as possible.
- The documentation must guide a potential participant through all steps
- The documentation must be available to the public
- The documentation must be governed enough to avoid unexpected changes

Considered Options

- Linux Foundation wiki - wiki.openIDL.org
- Microsoft Word
- Google Docs
- Read the Docs
- Video / Screen Capture tutorials
  - Workshop content
- GitHub

Decision Outcome

- For the time being, the documentation is captured in this wiki.
- As it matures, documentation will be migrated to ReadTheDocs

Consequences

- Good, because the wiki is easy to manage, add, remove, move, edit documentation and versioned
- Good, because the wiki is open to all
- Good, because the wiki can be permissioned
- Bad, because wiki's can become disorganized

Pros and Cons of the Options

Linux Foundation Wiki - wiki.openidl.org

- Good, because easy to manage
- Good, because open to all
- Bad, because wiki's can become disorganized

Microsoft Word

- Good, because it is well known
- Good, because it is easy to use
- Good, because it is easy to create professional format
- Good, because diagrams etc are easy to include
- Bad, because it is document oriented and leads to complicated merging
- Bad, because it is sometimes hard to know which version
- Bad, because extra step to pen the document.

Google Docs

- Good, because it is well known
- Good, because it is easy to use
- Good, because diagrams etc are easy to include
- Good, because although it is document oriented it is easy to work together
- Neutral, because it is easy to create professional format
- Bad, because doesn't fit the wiki well for inclusion
- Bad due to access issues

Read the Docs

- Good, because it is used by HLF
- Good, because it is easy to access and navigate
- Good, because it is built on github and is inherently versioned
- Good, because it is built for version control and team development
- Good/Bad: Versioning can be a challenge
- Bad, because diagrams etc are difficult to include
- Bad, because maintenance requires extra steps
- Bad, because raw format is markdown and not easy to read

Videos / Screen Captured Tutorials

- Good, because they provide effective information
- Good, because they are easy to use
- Bad, because they are difficult to keep up to date

GitHub

- Good, commonly used
- Good, version control
- Good, docs stay with codebase
- Bad, hard to navigate, raw form
- Bad, manual navigation