OMG FDTF DLT WG Call Notes

13 March 2018

# Attendees

* Dan Webb
* Rob Nehmer
* Mike Bennett
* Nick Stavros
* Bobbin Teegarden
* Pete Rivett

# Agenda

* DIDO Review

# Call Notes

## DIDO Review – version 0.8.6

Rob has compared with 0.8 from Jan

Looking for addition on OWL - not seen

Length unchanged

Minor corrections pp 28 and 32.

Changes to references to some acronyms.

### Purpose

What are Nick's ambitions for this group's contribution to DIDO?

For example:

1. Inclusion of OWL / Semantic Web in this.
2. The inclusion of some of the architecture things in DIDO
3. De-buggin of the DIDO RA descriptions against the available architecures

BT has some thoughts:

Problem: how the architecture is represented as layered architecture.

What is the implied ontology of that layered architecture itself?

Need an ontology of the various architecture components.

Need this not to imply there is a layering, versus a many to many potential relationship across those components that are represented as layers.

### Fig 8.

BT: So if we were to try and come up with an ontology reverse engineered out of that diagram?

For example there is an implied interface in the middle; one between wallet and the app and so on.

This may be an interesting exercise for this group to do.

RN: see paragraph beneath that

### Fig 9

MB this seems to conflate the record with the architecture.

RN: See Fig 10 which seems the same as Fig 8 but with different coloring.

BT also is each thing distributed? Separately or the same?

RN is a wallet a node? Is an identity a node? Are the 4 things in blue in Fig 12 a node etc.

Fig 9 suggest that every node in the network contains every aspect of the architecture, rather than the usual idea that the Block specifically is the thing so treated.

BT: Also the server is not shown, or the relation from server to the various pieces.

BT these are areas where the architectures are very different between Tangle, Ethereum etc.

In this variation, the server node does what a network node does in IP for example.

A lot of this is similar to what DNS does.

In Tangle, you enter the network through one node, you validate 3 other things, and then you are in.

So there is some ripple among the servers, and some protocol that supports this. That's the real network, not the application stuff like wallets and identity.

Something needs to be the layer upon which the network shown in Fig 9 is built.

### Architectural considerations

See Leeman Baird YouTube presentation. Explains how the different layers provide their services.

Approaches things from te PoV of algorithms at the basic levels of networking.

https://www.youtube.com/watch?v=IjQkag6VOo0

BT: Try and go to the next Blockchain Expo, where this guy will be speaking.

This will help us determine how you differentiate these different functions.

Include the concept of 'platform', this being the service level where you agree on the order of txns. The 'state' lives on top fo that as a data object, that is fed by the platform.

## Changes:

### Additions - p 24

Dan Webb Leemon Baird’s 40:00 video on Hashgraph architecture and sample dApps:

Dan Webb https://www.youtube.com/watch?v=VuOgdOKOHd8

See 2.3.3 and Fig 13

This is based on last week's conversation.

## Ancillary Data

Ancillary is outside the ledger. Some applications put the data on the ledger itself.

The ancillary data may be one of several forms as shown in Fig 13.

Definition: Ancillary data means data that is not on the ledger.

Ethereum tries to put even source code on the ledger.

The definition of Ancillary data in para 2 is different to the definition of Ancillary data in par 1.

### Two definitions of Ancillary:

1. Information that is ancillary to the application
2. Data that is not on the ledger.

Why will this be a problem down the line?

NS: By greying the line between the different lines of data, the rules will be polluted. Problems since the data would be different.

Code is not traditionally thought of as data. Likewise an ontology.

Bobbin Teegarden Also, what is an Oracle? It's not the Oracle DB...

Which of these 2 meanings shall we call Ancillary Data and what do we call the other one.

Oracular data?

This is an area where we can help.

NS welcomes places where we can help with distinguishing between these kinds of concepts, and in general, the concrete versus the conceptual.

### Comments on Fig 9:

This mixes conceptual and logical.

Conceptually, every node should have every one of these components.

In implementation this is not the case. In a wallet, you don't necessarily have access to the ledger.

Nodes with the ledger might not have the wallets on

And the mining nodes will not have those things.

So there is some configuration whereby each node may have some of these components.

Are we talking about the underlying service to service infrastructure or the app itself that runs on the server network?

Every node on the network has to have distribution software.

That's the application layer.

NS: there are no servers in the distributed architecture.

This is like DNS. There are DNS servers and these distribute the content, by collaboration, according to a protocol.

So there is an underlying server network layer.

NS: in Ethereum and Bitcoin, ever node is an equal partner. They may or may not choose to do all the parts.

That is every node in the Ethereum network

There is something that know what the thing is that hold all these application components.

How does the network know what physical nodes are participating in the network?

How is consensus done?

There is no one way to solve this.

How does the individual node know who to tell?

This is broadcast to the network. In any number of ways according to any number of protocols.

They will have chosen different ones.

Each node need not know everything but it has to know something i.e. what protocols to listen to, what nodes it is connected to.

May be a pub sub protocol - again not always.

The word Server seems to mean 2 things: a machine, and what a server usually does.

There is an IP protocol they use for publishing and discovering each other.

Also look at Hashgraph and Tangle.

Goal: be able to have those guys fit into this architecture.

### What is DIDO for?

It is to provide a descriptive framework that can be applied to any of the possible and existing distributed ledger type architectures.

Layered view: propose that we disconnect these and show them in an ontology.

This would be a classic ontology, against which you show what part of the architecture is implemented and in what order.

There is an implied ontology in Fig 8 is layered with some unexpected dependencies.

We can work on this in the OMG session.

These come from a reading of Bitcoin, Ethereum.

MB recommends that DIDO look at the CORDA / R3 paper that describes the notion of a 'Bitcoin Menu'.

# Reston Agenda

Next week: we have 9 - 10 Wednesday on DIDO

We have our usual GTM for the FDTF calls.