

FDC3 - February 6th, 2018 Minutes

Attendees:

Nikhil Doshi, Barclays
Alexei Samonov, BGC
Tim Kolečke, Citadel
Zeno Lee, Cloud9
Alan Cenkus, FactSet
Dan Riley, FactSet
Johan Sandersson, FactSet
Gavin Lauchlan, JP Morgan
Shahab Mahtab, JP Morgan
Ben Lis, LEI
Ian Alderson, Morgan Stanley
Peter Smulovics, Morgan Stanley
Nick Kolba, OpenFin
Espen Overbye, OpenFin
Chuck Danielsson, OpenFin
Osiris Forcade, RBC
Benoit Muller, Thomson Reuters
Jorge Santos, Thomson Reuters
Mathieu Lorentz, TRowe
David Carlson, Wellington
Eamonn Fanning, Wellington
Ed Broach, Wellington
Joseph Carroll, Wellington

Proposal Update: Context Specification

Since our last meeting, breakout groups were scheduled and met on both Context Data & Intents. An initial version 0.0.1 of the Context Data specification has been proposed to the group. Please see the proposal here:

<https://github.com/FDC3/ContextData/blob/master/Specification-Draft.MD>

Security types become reserved names - by convention today
i.e. you use the type Security, it needs to be an FDC3 security
If you want to make something up, call it something more unique
If it's a CUSIP it should be a CUSIP all caps

In the proposal - who defines short name/long name?

The Symphony Standard is a Symphony standard, not FDC3. The Symphony object has structure to aid display

FDC3 assumes your app will

- 1) pull in an identifier
- 2) resolve the display information by that identifier (ticker, ISIN, CUSIP)

What about a portfolio object or composite object?

Not supported today, we should have a breakout subgroup around this. It's not clear yet what a portfolio object would look like. But by default we can have a multi-context array.

You could turn your list into an array. Recognize it would be a really big list but not sure what you could pass around by portfolio.

Not sure if the roll of FDC3 would be to say `eikon portfolio type`, `factset portfolio`, vs. Security - which has 1 meaning

REMINDER: Goal is to get to an approval state @ which point we vote to approve
Fully expect we'll want to add/change things over time @ which point we'll go through a process to do that. Will be a working document which will be used to implement solutions

I thought we decided no state on intents. One of the verbs is `add` which would require state, no?

Agree, to be covered in the next breakout session

What is the lifecycle of these nouns & verbs - how will the app determine the noun/verb protocol exists?

The way we're thinking about this - these intents would live inside an app registry, apps would be tagged with the & there would be a search mechanism to find apps by intent which would be used explicitly or implicitly.

Given that it would be relatively dynamic, in general the lifecycle shouldn't be too much of a problem to manage. However, if we are removing things there could be some more thought that we need to have there.

We are anticipating there being some iteration to get this all correct. For example, web intents. Standards can sometimes fail - for both good reasons & circumstantial reasons - deprecate things that aren't working well without breaking applications.

DEMO

An example of the API demo was presented to the group, for the reference implementation please see our Github.

fdc3.open(name, intent, context) -> now takes a structured context that follows the spec we just saw

Also added an API that will perform searches against the app registry

API **get** is public, **post** requires authorization

Demonstrate show, reads context changes. Will relaunch if app is closed, etc.

Action Items

- We will be voting on the Context Data specification in the next meeting, please review before the next session