October 21, 2020

Attendees:

Bharath Kumar Bhaskar Kizhur

Bob Monkman

Jonas Arndt

Susan Bowen

Jon Mechling

Muthukkumaran Ramalingam

Piotr Zedlewski

Divya Nettem

Alexander Vul

John Leung

Martin Halstead

Joseprabu Inbaraj

Daniel Nilsson

William Caban

Rafal Sztejna

Marcin Nicpon

Tomas Fredberg

Agenda:

- We need to continue the conversation about API Data Model shared between Aggregator, Composer and other parts of ODIM. This was started on Oct 7 TSC Meeting.
- Where are we on the Composition Service?
- Martin Halstead sent out a white paper proposal on September 25th. Any feedback?
- We need to pick a name for repository for BMC simulator that will be contributed by Intel.
- What will be ODIM official copyright header? Current contribution from HPE contains company name, shall we follow that pattern? What if company X modifies source code file created by company Y?

Minutes:

Composition Service

Some discussions around the stand-alone services and how they would have a hard time to use things like the Tasks and the message bus. To allow for that we would need to open up APIs to the existing ODIM task mechanism. The Kafka based message bus can obviously be reached by Kafka APIs but what happens if the underlying message bus implementation is switched out? In ODIM there is some abstraction in place for that but it is Golang based.

It appears that we are basically agreeing about the approach here. We need to understand where the composition service is though. The composition service only manages the resources in it. John was talking about the current DMTF task force and how they are discussing a hierarchical composition services. Local and centralized ones. We should focus on the local one in ODIM was the conclusion.

If the Composition service is on top of ODIM his some use cases would not work. If somebody does a firmware upgrade ODIM wouldn't have a clue as to what resources are used or not. So the Composition service needs to be part of the other services in the same stack. We need them to be tightly coupled. The question here is if this forces the composition service to also be implemented in Golang and in the same stack or not.

Recording

Chat was not used during this meeting