

July 29, 2020

Participants

[Tomas Fredberg](#) (Ericsson)

Agenda

- Logo Conclusion
- Introduction of proposals:
 - Update Service
 - Plugin for Unmanaged Racks
- K8S instead of Docker in CICD
- Task States not saved today - Minor discussions and for HPE to understand if we need a new proposal to change that (don't assume so)

[July 29 2020 - Meeting Recording](#)

[July 29 2020 - Chat File](#)

Notes

Logo

With regards to the logo the poll is [here](#) and it is clear that Concept 4 was the most favored Logo. After a vote it was decided to adopt that logo



Proposals

There two major proposals pending for TSC review. One is for the Update Service and another for a Plugin for Unmanaged Racks:

- [Update Service](#)
- [Plugin for Unmanaged Racks](#)

For the updates service we concluded that as this is the implementation of an existing service in Redfish we could make a quick vote to accept that as a contribution target. The vote was in favor of this.

The second proposal to have a plugin for unmanaged racks led to a bigger discussion and we will need some more time to review the proposal. It was decided that we would continue to do that over the next week and perhaps make a decision by next TSC meeting. The main reason for the existence of this type of plugin is to be able to address and manipulate resources under `/redfish/v1/Chassis/<uuid>` where `ChassisType=rack` and specifically the `Contains` property but also things like `Location`.

Other pending work

Aggregation Service

We also discussed updating the existing Aggregation Service so that aligns with the spec. The reason it does not today is that it was developed in parallel with the contribution of the service to DMTF. During the DMTF discussions it slightly changed and this is why there is a delta. This TSC determined was more of a bug fix than a full blown proposal. Changes to the current Aggregation Service will therefore be treated as a bug fix.

CICD

We decided to move the current CICD from being based on Docker to instead utilize K8S. This was done through a vote

Persistence of Tasks

We had a discussion on how the current Task implementation is not persisted and that this could be problematic during things like reboots. As a client it would be beneficial for ODIM to at least know that a task that use to exist was in flight after a reboot and to display some type of state (maybe 'Interrupted') for the task after a reboot. This again can be treated as a bug fix