

# DMTF Profiles Proposal

## Proposal for Profiles

### Introduction

DMTF Redfish is a great standard that due to its very active community is evolving rapidly. More and more hardware vendors are implementing Redfish capabilities on their BMC interfaces. While it is great that you get rapid adoption there is also an ongoing problem with the implementation, and that is interoperability. The approach the DMTF Redfish specification is taking is to be as flexible as possible. It does that by making the vast majority of properties optional. With that approach we can see that the Redfish implementation from different vendors are different. This in turn poses a challenge when trying to write clients that programmatically interact with the different vendors' BMCs. Of course, this is part of what ODIM is trying to address.

### Redfish Profiles

To address this side effect, that is the result of a very open and flexible spec, the DMTF standards body came up with the Redfish Profile concept. A Redfish profile is basically tightening up the loosely defined spec and defines what properties need to be present as well as what permissions should be tied to those properties (i.e read-only, read-writable). Of course this is very useful. The issue is really that the vast majority of the hardware vendors still don't pay much attention to this. There are no common profiles adopted by HPE, Dell, and others. Again, this is where ODIM comes into the picture.

### Redfish Profiles in ODIM

The introduction of Redfish profiles to ODIM solves mainly two problems

1. It gives the users of northbound Redfish interfaces an idea of what properties to expect to be present when interacting with ODIM. This as well as if these properties are read-only or read-writable
2. It give the developers of plugins an idea of what properties they need to implement for the plugin to work in an ODIM environment

**Apart from this the idea to introduce profiles in ODIM is also to get them socialized in the industry and by that apply pressure on hardware vendors to adopt them. Only when the vast majority of hardware vendors have adopted some common profiles will we really achieve real interoperability.**

Of course, ODIM only has a couple of "Open Source" plugins available. There are also commercial plugins from some vendors. In order for the introduction of profiles in ODIM to be successful we need them to be a key component of ODIM that is constantly refined and vetted. They cannot be a one-off that is created, published, and forgotten about. We've seen that in other community driven efforts and it is really hard to not view that as a failure.

### Proposal in an essence

So how to we make the current profiles (that are hidden in a branch) successful? Here is what I am proposing

- Open collaboration in reviewing the existing profiles to improve them. Note that they are in a draft state in a branch. We need to set a time line here so this doesn't go on forever
- Work with OCP to get our added properties and changed permissions added to the OCP profiles
- Enhance the current open source plugins that are hosted by the project so that they are compliant with the profiles.
- Make the CI/CD pipeline include tests that verify compliance (e.g. DMTF Redfish-Interop\_Validator). Note that this tool currently might have some issues that need to be worked out with DMTF. Not sure what the status is there.
- Make sure the BMC Simulator support the Profiles and make it part of the CI/CD pipeline as well.
- Reject any new plugins contributions that do not comply with the profiles.
- Continuously refine the profiles as new requirements come in and the industry is changing.

### Conclusion

If we manage to pull all this through we have a platform that can be used to influence the industry and hardware vendors. As the vendors start looking at our profiles and make changes to their interfaces to comply with them the big reason to have vendor specific plugins would be removed. All vendors that are compliant with our profiles should function with ODIM's open source plugins.

Without these profiles ODIM is basically just an abstraction layer that solves current problems and provides value add on top of that. We want to be more than that. We want to influence the industry to fix the root of the interoperability issue that exists.

### Presentation from TSC



ODIM Profiles.pdf