



# Enhancement of LogService for Diagnostic Data

Redfish Forum – Work in progress  
May 2020



## Disclaimer

- The information in this presentation represents a snapshot of work in progress within the DMTF.
- This information is subject to change without notice. The standard specifications remain the normative reference for all information.
- For additional information, see the DMTF website.





## Providing Feedback

- Feedback to the DMTF Redfish Forum is encouraged
  - Submit items using the DMTF feedback portal
  - <https://www.dmtf.org/standards/feedback>
- Questions and comments can be posted on the Redfish User Forum
  - <https://www.redfishforum.com>





## Diagnostic data overview

- Diagnostic data can be collected at any point of time from the system and is stored in a file to troubleshoot problems that have occurred
- The data may consist of a “crash dump”, application core, network configuration, system inventory configuration, journal log, etc.

## Requirements

- Provide a method through which user can collect the diagnostic data from the system and can retrieve it through Redfish
- This diagnostic data can be generated in following ways:
  - User Initiated (collect the diagnostic data at any moment of time)
  - Critical software or hardware failure



## Proposed Data Model – Enhance LogService/LogEntry

Redfish Resource	Action	Details
<b>LogService</b>	<b>CollectDiagnosticData</b>	Collects the diagnostic data

**NOTE:**

1. **CollectDiagnosticData** spawns a task and returns the taskID. Client can query the status of the task using the taskID.

Redfish Resource	Property	Description
<b>LogEntry</b>	<b>AdditionalDataUri</b>	URI of the file associated with the entry
	<b>AdditionalDataSizeBytes</b>	Size of the Additional data file
	<b>DiagnosticDataType</b>	Type of diagnostic data
	<b>OEMDiagnosticDataType</b>	OEM diagnostic data type



## Proposed Data Model (LogService) - Mockup

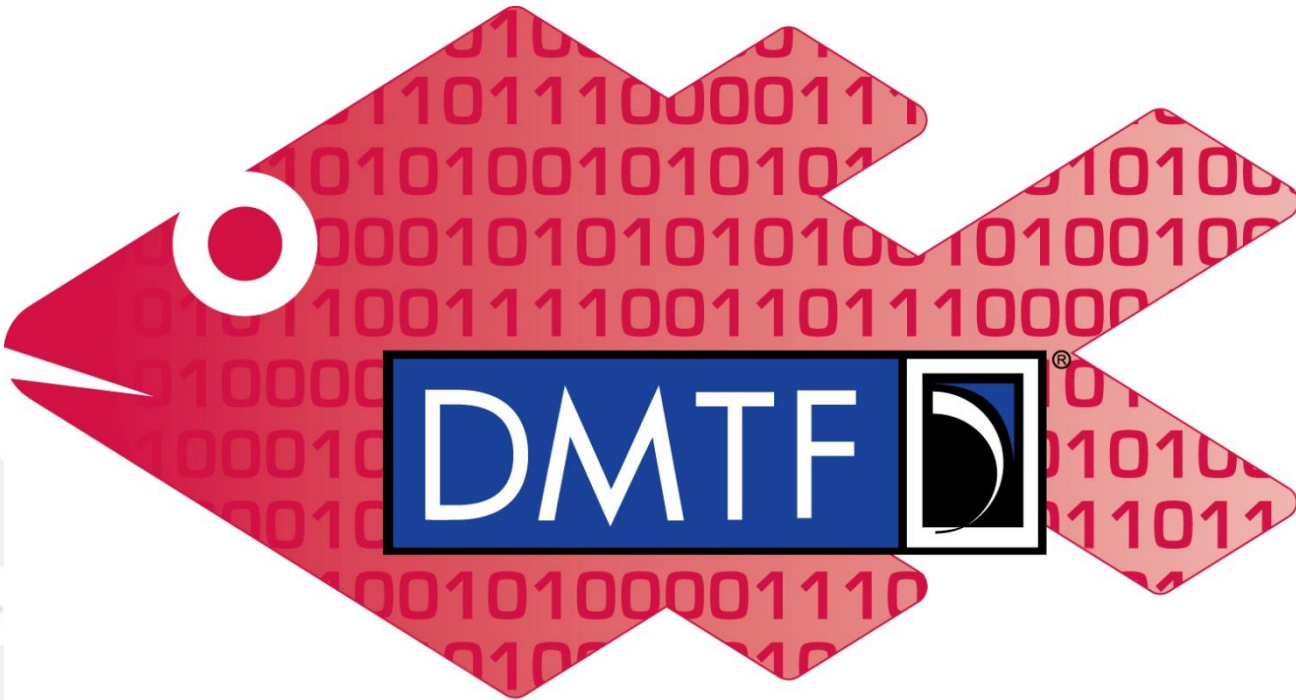
redfish » v1 » Systems » system » LogServices » EventLog

```
{
  "@odata.id": "/redfish/v1/Systems/system/LogServices/EventLog",
  "@odata.type": "#LogService.v1_1_0.LogService",
  "Actions": {
    "#LogService.ClearLog": {
      "target": "/redfish/v1/Systems/system/LogServices/EventLog/Actions/LogService.ClearLog"
    },
    "#LogService.CollectDiagnosticData": {
      "target": "/redfish/v1/Systems/system/LogServices/EventLog/Actions/LogService.CollectDiagnosticData"
    }
  },
  "Description": "System Event Log Service",
  "Entries": {
    "@odata.id": "/redfish/v1/Systems/system/LogServices/EventLog/Entries"
  },
  "Id": "Event Log",
  "Name": "Event Log Service",
  "OverWritePolicy": "WrapsWhenFull"
}
```

## Proposed Data Model (LogEntryCollection) - Mockup

redfish » v1 » Systems » system » LogServices » EventLog » Entries

```
{
  "@odata.id": "/redfish/v1/Systems/system/LogServices/DiagnosticLog/Entries",
  "@odata.type": "#LogEntryCollection.LogEntryCollection",
  "Description": "Collection of System Event Log Entries",
  "Members": [{
    "@odata.id": "/redfish/v1/Systems/system/LogServices/EventLog/Entries/111",
    "@odata.type": "#LogEntry.v1_4_0.LogEntry",
    "Created": "2020-01-01T14:44:00Z",
    "EntryType": "Event",
    "DiagnosticDataType": "PreOS",
    "Id": "111",
    "Message": "User initiated dump",
    "MessageId": "Diagnostics.1.0.UserInitiatedDiagnosticDump",
    "Name": "System Event Log Entry",
    "AdditionalDataUri": "/redfish/v1/Systems/system/LogServices/EventLog/attachement/111",
    "AdditionalDataSizeBytes": 1048576
  }],
  "Members@odata.count": 1,
  "Name": "System Event Log Entries"
}
```



# Redfish