



Document Identifier: DSP2065

Date: 2022-09-15

Version: 2022.2

Redfish Message Registry Guide

Supersedes: 2022.1

Document Class: Informational

Document Status: Published

Document Language: en-US

Copyright Notice

Copyright © 2022 DMTF. All rights reserved.

DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems management and interoperability. Members and non-members may reproduce DMTF specifications and documents, provided that correct attribution is given. As DMTF specifications may be revised from time to time, the particular version and release date should always be noted.

Implementation of certain elements of this standard or proposed standard may be subject to third party patent rights, including provisional patent rights (herein "patent rights"). DMTF makes no representations to users of the standard as to the existence of such rights, and is not responsible to recognize, disclose, or identify any or all such third party patent right, owners or claimants, nor for any incomplete or inaccurate identification or disclosure of such rights, owners or claimants. DMTF shall have no liability to any party, in any manner or circumstance, under any legal theory whatsoever, for failure to recognize, disclose, or identify any such third party patent rights, or for such party's reliance on the standard or incorporation thereof in its product, protocols or testing procedures. DMTF shall have no liability to any party implementing such standard, whether such implementation is foreseeable or not, nor to any patent owner or claimant, and shall have no liability or responsibility for costs or losses incurred if a standard is withdrawn or modified after publication, and shall be indemnified and held harmless by any party implementing the standard from any and all claims of infringement by a patent owner for such implementations.

For information about patents held by third-parties which have notified the DMTF that, in their opinion, such patent may relate to or impact implementations of DMTF standards, visit <http://www.dmtf.org/about/policies/disclosures.php>.

This document's normative language is English. Translation into other languages is permitted.

CONTENTS

1 Overview	4
1.1 Who should read this document?	4
1.2 How can I provide feedback?	4
1.3 Where can I find more information?	4
1.4 Related documents	5
2 Reference guide	7
2.1 Base 1.14.0	7
2.2 Composition 1.1.0	56
2.3 EthernetFabric 1.0.0	63
2.4 Fabric 1.0.0	67
2.5 HeartbeatEvent 1.0.0	88
2.6 JobEvent 1.0.0	89
2.7 License 1.0.1	93
2.8 LogService 1.0.0	96
2.9 NetworkDevice 1.0.1	97
2.10 ResourceEvent 1.2.1	101
2.11 SensorEvent 1.0.0	113
2.12 StorageDevice 1.1.0	124
2.13 TaskEvent 1.0.3	140
2.14 Update 1.0.1	144
3 ANNEX A (informative) Change log	152

1 Overview

The Redfish standard comprises a set of specifications maintained by the Redfish Forum, a working group within the DMTF. The standard defines a protocol that uses RESTful interfaces to provide access to data and operations associated with the management of systems and networks. One of the strengths of the Redfish protocol is that it works with a wide range of servers: from stand-alone servers to rack-mount and bladed environments to large-scale data centers and cloud environments.

The Redfish standard addresses several key issues for infrastructures that require scalability. Large infrastructures often consist of many simple servers of different makes and types. This hyper-scale usage model requires a new approach to systems management. The Redfish protocol addresses these needs by providing a standard protocol based on out-of-band systems management.

With these goals in mind, the Redfish protocol was designed as an open-industry standard to meet scalability requirements in multi-vendor deployments. It easily integrates with commonly used tools, using RESTful interfaces to perform operations and using JSON for data payloads.

1.1 Who should read this document?

Redfish responses can contain messages to help guide clients when next steps or corrective actions need to be taken. This document is useful for client developers who want to understand how to interpret standard Redfish messages. This document is also useful for service developers who want to understand how standard Redfish messages are to be used in their service.

1.2 How can I provide feedback?

Feedback on all Redfish specifications and documents is encouraged. Feedback can be directed to the DMTF and the Redfish Forum by the following means:

- **Redfish User Forum:** <http://www.redfishforum.com> User forum monitored by DMTF Redfish Forum personnel to answer questions about any Redfish-related topics.
- **DMTF Feedback Portal:** <https://www.dmtf.org/standards/feedback> Formal submission portal for enhancements or proposals to the DMTF and Redfish Forum.

1.3 Where can I find more information?

The following web sites provide more information about the Redfish standard:

- [Redfish Developer Hub](#)

Resources for developers who use Redfish to build applications. Contains an interactive schema explorer, hosted schema, and other links.

- **Redfish Specification Forum**

DMTF Redfish-monitored user forum. Answers questions about Redfish-related topics.

- **DMTF GitHub repositories**

Open source tools and libraries for working with Redfish.

- **Redfish standards**

Schemas, specifications, mockups, white papers, FAQ, educational material, and more.

- **DMTF Redfish Forum**

Working group that maintains the Redfish standard. Site lists member companies, future work and schedules, charter, and information about joining.

1.4 Related documents

The following documents are part of the Redfish development effort. They can be accessed or downloaded from the DMTF Redfish Standards web site: <http://www.dmtf.org/standards/redfish>

- DSP0218 - Platform Level Data Model (PLDM) for Redfish Device Enablement Specification - Binary-encoded JSON (BEJ) and dictionary-based mapping of Redfish schemas and properties into PLDM messages.
- DSP0266 - Redfish Specification - Main Redfish Specification.
- DSP0268 - Redfish Data Model Specification - Normative descriptions and additional text for every schema defined in DSP8010 and example payloads for every resource.
- DSP0270 - Redfish Host Interface Specification - "In-band" or "OS-based" Redfish host interface.
- DSP0272 - Redfish Interoperability Profiles Specification - Structure and JSON document that is used to define and publish an interoperability profile that checks an implementation's conformance to a defined minimum set of functionality.
- DSP2043 - Redfish Mockups Bundle - Set of mockups that can be used as sample output from `GETs` from a Redfish service. Informative in nature, it was used to develop the schema. A person can set up an NGINX or similar server and configure it to output JSON format and then use this directory for demonstration purposes.
- DSP2044 - Redfish White Paper - Non-normative document helping those new to Redfish understand how to interact with the Redfish service and understand common functions and tasks.
- DSP2046 - Redfish Resource and Schema Guide - Informative documentation regarding common Redfish resource properties and a listing of properties that can be found in each of the Redfish resources.
- DSP2053 - Redfish Property Guide - Informative documentation providing an index to individual property

definitions across all Redfish schema.

- DSP2065 - Redfish Message Registry Guide - Informative documentation providing details regarding the messages defined in Redfish standard message registries.
- DSP8010 - Redfish Schema - Redfish schema definitions. These files are normative in nature and are normatively referenced by the *Redfish Specification*. The three schema formats are CSDL (OData Common Schema Definition Language format, which is in XML), JSON Schema, and OpenAPI schema. These schema definitions should be functionally equivalent, thus specifying the schema in two different languages.
- DSP8011 - Redfish Standard Registries - Redfish registry definitions. This bundle of Redfish registries includes message registries used for Redfish-defined messages including events and privilege maps.
- DSP8013 - Redfish Interoperability Profiles Bundle - Bundle of published Redfish interoperability profile documents and supporting schema and sample documents used for creating profiles.

2 Reference guide

The following sections contain details for each message registry and their messages in the Redfish Standard Registries (DSP8011).

2.1 Base 1.14.0

This registry defines the base messages for Redfish

Message	Severity	Description
AccessDenied	Critical	Indicates that while attempting to access, connect to, or transfer to or from another resource, the service denied access.
AccountForSessionNoLongerExists	OK	Indicates that the account for the session has been removed, thus the session has been removed as well.
AccountModified	OK	Indicates that the account was successfully modified.
AccountNotModified	Warning	Indicates that the modification requested for the account was not successful.
AccountRemoved	OK	Indicates that the account was successfully removed.
ActionDeprecated	Warning	Indicates the action is deprecated.
ActionNotSupported	Critical	Indicates that the action supplied with the POST operation is not supported by the resource.
ActionParameterDuplicate	Warning	Indicates that the action was supplied with a duplicated action parameter in the request body.
ActionParameterMissing	Critical	Indicates that the action requested was missing an action parameter that is required to process the action.
ActionParameterNotSupported	Warning	Indicates that the parameter supplied for the action is not supported on the resource.
ActionParameterUnknown	Warning	Indicates that an action was submitted but an action parameter supplied did not match any of the known parameters.
ActionParameterValueConflict	Warning	Indicates that the requested parameter value could not be completed, because of a mismatch with other parameters or properties in the resource.
ActionParameterValueError	Warning	Indicates that a parameter was given an invalid value.
ActionParameterValueFormatError	Warning	Indicates that a parameter was given the correct value type but the value of that parameter was not supported. This includes the value size or length has been exceeded.

Message	Severity	Description
ActionParameterValueNotInList	Warning	Indicates that a parameter was given the correct value type but the value of that parameter was not supported. The value is not in an enumeration.
ActionParameterValueTypeError	Warning	Indicates that a parameter was given the wrong value type, such as when a number is supplied for a parameter that requires a string.
ChassisPowerStateOffRequired	Warning	Indicates that the request requires a specified chassis to be powered off.
ChassisPowerStateOnRequired	Warning	Indicates that the request requires a specified chassis to be powered on.
ConditionInRelatedResource	Warning	Indicates that one or more fault or error conditions exist in a related resource.
CouldNotEstablishConnection	Critical	Indicates that the attempt to access the resource, file, or image at the URI was unsuccessful because a session could not be established.
CreateFailedMissingReqProperties	Critical	Indicates that a create was attempted on a resource but that properties that are required for the create operation were missing from the request.
CreateLimitReachedForResource	Critical	Indicates that no more resources can be created on the resource as it has reached its create limit.
Created	OK	Indicates that all conditions of a successful create operation have been met.
EmptyJSON	Warning	Indicates that the request body contained an empty JSON object when one or more properties are expected in the body.
EventBufferExceeded	Warning	Indicates undelivered events may have been lost due to a lack of buffer space in the service.
EventSubscriptionLimitExceeded	Critical	Indicates that a event subscription establishment has been requested but the operation failed due to the number of simultaneous connection exceeding the limit of the implementation.
GeneralError	Critical	Indicates that a general error has occurred. Use in <code>@Message.ExtendedInfo</code> is discouraged. When used in <code>@Message.ExtendedInfo</code> , implementations are expected to include a <code>Resolution</code> property with this message and provide a service-defined resolution to indicate how to resolve the error.
HeaderInvalid	Critical	Indicates that a request header is invalid.
HeaderMissing	Critical	Indicates that a required request header is missing.
InsufficientPrivilege	Critical	Indicates that the credentials associated with the established session do not have sufficient privileges for the requested operation.
InsufficientStorage	Critical	Indicates that the operation could not be completed due to a lack of storage or memory available to the service.
InternalError	Critical	Indicates that the request failed for an unknown internal error but that the service is still operational.

Message	Severity	Description
InvalidIndex	Warning	The index is not valid.
InvalidJSON	Critical	Indicates that the request body contains invalid JSON.
InvalidObject	Critical	Indicates that the object in question is invalid according to the implementation. Examples include a firmware update malformed URI.
InvalidURI	Critical	Indicates that the operation encountered a URI that does not correspond to a valid resource.
MalformedJSON	Critical	Indicates that the request body was malformed JSON.
MaximumErrorsExceeded	Critical	Indicates that sufficient errors have occurred that the reporting service cannot return them all.
MissingOrMalformedPart	Critical	Indicates that a multipart request is missing a required part or contains malformed parts.
NetworkNameResolutionNotConfigured	Warning	Indicates that network-based name resolution has not been configured on the service.
NetworkNameResolutionNotSupported	Warning	Indicates the service does not support network-based name resolution.
NoOperation	Warning	Indicates that the requested operation will not perform any changes on the service.
NoValidSession	Critical	Indicates that the operation failed because a valid session is required in order to access any resources.
OperationFailed	Warning	Indicates that one of the internal operations necessary to complete the request failed. Examples of this are when an internal service provider is unable to complete the request, such as in aggregation or RDE.
OperationNotAllowed	Critical	Indicates that the HTTP method in the request is not allowed on this resource.
OperationTimeout	Warning	Indicates that one of the internal operations necessary to complete the request timed out. Examples of this are when an internal service provider is unable to complete the request, such as in aggregation or RDE.
PasswordChangeRequired	Critical	Indicates that the password for the account provided must be changed before accessing the service. The password can be changed with a PATCH to the <code>Password</code> property in the manager account resource instance. Implementations that provide a default password for an account may require a password change prior to first access to the service.
PayloadTooLarge	Critical	Indicates that the supplied payload is too large to be accepted by the service.
PreconditionFailed	Critical	Indicates that the ETag supplied did not match the current ETag of the resource.
PreconditionRequired	Critical	Indicates that the request did not provide the required precondition such as an <code>If-Match</code> or <code>If-None-Match</code> header, or <code>@odata.etag</code> annotations.
PropertyDeprecated	Warning	Indicates the property is deprecated.

Message	Severity	Description
PropertyDuplicate	Warning	Indicates that a duplicate property was included in the request body.
PropertyMissing	Warning	Indicates that a required property was not supplied as part of the request.
PropertyNotUpdated	Critical	Indicates that a property was not updated due to an internal service error, but the service is still functional.
PropertyNotWritable	Warning	Indicates that a property was given a value in the request body, but the property is a readonly property.
PropertyUnknown	Warning	Indicates that an unknown property was included in the request body.
PropertyValueConflict	Warning	Indicates that the requested write of a property value could not be completed, because of a conflict with another property value.
PropertyValueDeprecated	Warning	Indicates that a property was given a deprecated value.
PropertyValueError	Warning	Indicates that a property was given an invalid value.
PropertyValueExternalConflict	Warning	Indicates that the requested write of a property value could not be completed, due to the current state or configuration of the resource. This can include configuration conflicts with other resources or parameters that are not exposed by this interface.
PropertyValueFormatError	Warning	Indicates that a property was given the correct value type but the value of that property was not supported.
PropertyValueIncorrect	Warning	Indicates that the requested write of a property value could not be completed, because of an incorrect value of the property. Examples include values that do not match a regular expression requirement or passwords that do not match the implementation constraints.
PropertyValueModified	Warning	Indicates that a property was given the correct value type but the value of that property was modified. Examples are truncated or rounded values.
PropertyValueNotInList	Warning	Indicates that a property was given the correct value type but the value of that property was not supported. The value is not in an enumeration.
PropertyValueOutOfRange	Warning	Indicates that a property was given the correct value type but the value of that property is outside the supported range.
PropertyValueResourceConflict	Warning	Indicates that the requested write of a property value could not be completed, due to the current state or configuration of another resource.
PropertyValueTypeError	Warning	Indicates that a property was given the wrong value type, such as when a number is supplied for a property that requires a string.
QueryCombinationInvalid	Warning	Indicates the request contains multiple query parameters, and that two or more of them cannot be used together.
QueryNotSupported	Warning	Indicates that query is not supported on the implementation.

Message	Severity	Description
QueryNotSupportedOnOperation	Warning	Indicates that query is not supported with the given operation, such as when the <code>\$expand</code> query is attempted with a PATCH operation.
QueryNotSupportedOnResource	Warning	Indicates that query is not supported on the given resource, such as when the <code>\$skip</code> query is attempted on a resource that is not a collection.
QueryParameterOutOfRange	Warning	Indicates that a query parameter was provided that is out of range for the given resource. This can happen with values that are too low or beyond that possible for the supplied resource, such as when a page is requested that is beyond the last page.
QueryParameterValueError	Warning	Indicates that a query parameter was given an invalid value.
QueryParameterValueFormatError	Warning	Indicates that a query parameter was given the correct value type but the value of that parameter was not supported. This includes the value size or length has been exceeded.
QueryParameterValueTypeError	Warning	Indicates that a query parameter was given the wrong value type, such as when a number is supplied for a query parameter that requires a string.
ResetRecommended	Warning	Indicates that a component reset is recommended for error recovery while unaffected applications can continue running without any effects on accuracy and performance.
ResetRequired	Warning	Indicates that a component reset is required for changes, error recovery, or operations to complete.
ResourceAlreadyExists	Critical	Indicates that a resource change or creation was attempted but that the operation cannot proceed because the resource already exists.
ResourceAtUriInUnknownFormat	Critical	Indicates that the URI was valid but the resource or image at that URI was in a format not supported by the service.
ResourceAtUriUnauthorized	Critical	Indicates that the attempt to access the resource, file, or image at the URI was unauthorized.
ResourceCannotBeDeleted	Critical	Indicates that a delete operation was attempted on a resource that cannot be deleted.
ResourceCreationConflict	Warning	Indicates that the requested resource creation could not be completed because the service has a resource that conflicts with the request.
ResourceDeprecated	Warning	Indicates the resource is deprecated.
ResourceExhaustion	Critical	Indicates that a resource could not satisfy the request due to some unavailability of resources. An example is that available capacity has been allocated.
ResourceInStandby	Critical	Indicates that the request could not be performed because the resource is in standby.
ResourceInUse	Warning	Indicates that a change was requested to a resource but the change was rejected due to the resource being in use or transition.

Message	Severity	Description
ResourceMissingAtURI	Critical	Indicates that the operation expected an image or other resource at the provided URI but none was found. Examples of this are in requests that require URIs like firmware update.
ResourceNotFound	Critical	Indicates that the operation expected a resource identifier that corresponds to an existing resource but one was not found.
ResourceTypeIncompatible	Critical	Indicates that the resource type of the operation does not match that for the operation destination. Examples of when this can happen include during a POST to a resource collection using the wrong resource type, an update where the <code>@odata.type</code> properties do not match, or on a major version incompatibility.
RestrictedPrivilege	Warning	Indicates that the operation was not successful because a privilege is restricted.
RestrictedRole	Warning	Indicates that the operation was not successful because the role is restricted.
ServiceDisabled	Warning	Indicates that the operation failed because the service, such as the account service, is disabled and cannot accept requests.
ServiceInUnknownState	Critical	Indicates that the operation failed because the service is in an unknown state and cannot accept additional requests.
ServiceShuttingDown	Critical	Indicates that the operation failed as the service is shutting down, such as when the service reboots.
ServiceTemporarilyUnavailable	Critical	Indicates the service is temporarily unavailable.
SessionLimitExceeded	Critical	Indicates that a session establishment has been requested but the operation failed due to the number of simultaneous sessions exceeding the limit of the implementation.
SessionTerminated	OK	Indicates that the DELETE operation on the session resource resulted in the successful termination of the session.
SourceDoesNotSupportProtocol	Critical	Indicates that while attempting to access, connect to or transfer a resource, file, or image from another location that the other end of the connection did not support the protocol.
StrictAccountTypes	Warning	Indicates the request failed because a set of <code>AccountTypes</code> or <code>OEMAccountTypes</code> was not accepted while <code>StrictAccountTypes</code> is set to <code>true</code> .
StringValueTooLong	Warning	Indicates that a string value passed to the given resource exceeded its length limit. An example is when a shorter limit is imposed by an implementation than that allowed by the specification.
SubscriptionTerminated	OK	An event subscription has been terminated by the service. No further events will be delivered.
Success	OK	Indicates that all conditions of a successful operation have been met.

Message	Severity	Description
UndeterminedFault	Critical	Indicates that a fault or error condition exists but the source of the fault cannot be determined or is unknown to the service.
UnrecognizedRequestBody	Warning	Indicates that the service encountered an unrecognizable request body that could not even be interpreted as malformed JSON.

2.1.1 AccessDenied

Indicates that while attempting to access, connect to, or transfer to or from another resource, the service denied access.

- This message shall be used to indicate that while attempting to access, connect to, or transfer to or from another resource, the service denied access.

Version Added: 1.0.0

Severity: Critical

Resolution: Attempt to ensure that the URI is correct and that the service has the appropriate credentials.

Message and Arguments: "While attempting to establish a connection to '<Arg1>', the service denied access."

1. *string*: The URI provided.
 - This argument shall contain the URI provided for this operation.

2.1.2 AccountForSessionNoLongerExists

Indicates that the account for the session has been removed, thus the session has been removed as well.

- This message shall be used to indicate that the account for the session has been removed, thus the session has been removed as well.

Version Added: 1.0.0

Severity: OK

Resolution: Attempt to connect with a valid account.

Message and Arguments: "The account for the current session has been removed, thus the current session has been removed as well."

2.1.3 AccountModified

Indicates that the account was successfully modified.

- This message shall be used to indicate that the account was successfully modified.

Version Added: 1.0.0

Severity: OK

Resolution: No resolution is required.

Message and Arguments: "The account was successfully modified."

2.1.4 AccountNotModified

Indicates that the modification requested for the account was not successful.

- This message shall be used to indicate that the modification requested for the account was not successful.

Version Added: 1.0.0

Severity: Warning

Resolution: The modification may have failed due to permission issues or issues with the request body.

Message and Arguments: "The account modification request failed."

2.1.5 AccountRemoved

Indicates that the account was successfully removed.

- This message shall be used to indicate that the account was successfully removed.

Version Added: 1.0.0

Severity: OK

Resolution: No resolution is required.

Message and Arguments: "The account was successfully removed."

2.1.6 ActionDeprecated

Indicates the action is deprecated.

- This message shall be used to indicate that the action is deprecated.

Version Added: 1.10.0

Severity: Warning

Resolution: Refer to the schema guide for more information.

Message and Arguments: "The action <Arg1> is deprecated."

1. *string*: The name of the action.
 - This argument shall contain the name of the action that is deprecated.

2.1.7 ActionNotSupported

Indicates that the action supplied with the POST operation is not supported by the resource.

- This message shall be used to indicate that the action supplied with the POST operation is not supported by the resource.

Version Added: 1.0.0

Severity: Critical

Resolution: The action supplied cannot be resubmitted to the implementation. Perhaps the action was invalid, the wrong resource was the target or the implementation documentation may be of assistance.

Message and Arguments: "The action <Arg1> is not supported by the resource."

1. *string*: The name of the action.
 - This argument shall contain the name of the action.

2.1.8 ActionParameterDuplicate

Indicates that the action was supplied with a duplicated action parameter in the request body.

- This message shall be used to indicate that the action was supplied with a duplicated action parameter in the request body.

Version Added: 1.0.0

Severity: Warning

Resolution: Resubmit the action with only one instance of the action parameter in the request body if the operation failed.

Message and Arguments: "The action <Arg1> was submitted with more than one value for the parameter <Arg2>."

1. *string*: The name of the action.
 - This argument shall contain the name of the action.
2. *string*: The name of the action parameter.
 - This argument shall contain the name of the action parameter.

2.1.9 ActionParameterMissing

Indicates that the action requested was missing an action parameter that is required to process the action.

- This message shall be used to indicate that the action requested was missing an action parameter that is required to process the action.

Version Added: 1.0.0

Severity: Critical

Resolution: Supply the action with the required parameter in the request body when the request is resubmitted.

Message and Arguments: "The action <Arg1> requires the parameter <Arg2> to be present in the request body."

1. *string*: The name of the action.
 - This argument shall contain the name of the action.
2. *string*: The name of the action parameter.
 - This argument shall contain the name of the action parameter.

2.1.10 ActionParameterNotSupported

Indicates that the parameter supplied for the action is not supported on the resource.

- This message shall be used to indicate that the parameter supplied for the action is not supported on the resource.

Version Added: 1.0.0

Severity: Warning

Resolution: Remove the parameter supplied and resubmit the request if the operation failed.

Message and Arguments: "The parameter <Arg1> for the action <Arg2> is not supported on the target resource."

1. *string*: The name of the action parameter.
 - This argument shall contain the name of the action parameter.
2. *string*: The name of the action.
 - This argument shall contain the name of the action.

2.1.11 ActionParameterUnknown

Indicates that an action was submitted but an action parameter supplied did not match any of the known parameters.

- This message shall be used to indicate that an action was submitted but an action parameter supplied did not match any of the known parameters.

Version Added: 1.0.0

Severity: Warning

Resolution: Correct the invalid action parameter and resubmit the request if the operation failed.

Message and Arguments: "The action <Arg1> was submitted with the invalid parameter <Arg2>."

1. *string*: The name of the action.
 - This argument shall contain the name of the action.
2. *string*: The name of the action parameter.
 - This argument shall contain the name of the action parameter.

2.1.12 ActionParameterValueConflict

Indicates that the requested parameter value could not be completed, because of a mismatch with other parameters or properties in the resource.

- This message shall be used to indicate that the requested parameter value could not be completed, because of a mismatch with other parameters or properties in the resource.

Version Added: 1.12.0

Severity: Warning

Resolution: No resolution is required.

Message and Arguments: "The parameter '<Arg1>' with the requested value of '<Arg2>' does not meet the constraints of the implementation."

1. *string*: The name of the parameter for which a write was requested.
 - This argument shall contain the name of the parameter provided in the action request.
2. *string*: The value of the parameter with which there is a conflict.
 - This argument shall contain the value of the parameter with which there is a conflict.

2.1.13 ActionParameterValueError

Indicates that a parameter was given an invalid value.

- This message shall be used to indicate that a parameter was given an invalid value, and the value cannot or should not be included in the message.

Version Added: 1.11.0

Severity: Warning

Resolution: Correct the value for the parameter in the request body and resubmit the request if the operation failed.

Message and Arguments: "The value for the parameter <Arg1> in the action <Arg2> is invalid."

1. *string*: The name of the action parameter.
 - This argument shall contain the name of the action parameter.
2. *string*: The name of the action.
 - This argument shall contain the name of the action.

2.1.14 ActionParameterValueFormatError

Indicates that a parameter was given the correct value type but the value of that parameter was not supported. This includes the value size or length has been exceeded.

- This message shall be used to indicate that a parameter was given the correct value type but the value of that parameter was not supported.

Version Added: 1.0.0

Severity: Warning

Resolution: Correct the value for the parameter in the request body and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the parameter <Arg2> in the action <Arg3> is of a different format than the parameter can accept."

1. *string*: The value provided for the action parameter.
 - This argument shall contain the value provided for the action parameter. Numeric values shall be

converted to a string, and null values shall be represented by the string `null`.

2. *string*: The name of the action parameter.
 - This argument shall contain the name of the action parameter.
3. *string*: The name of the action.
 - This argument shall contain the name of the action.

2.1.15 ActionParameterValueNotInList

Indicates that a parameter was given the correct value type but the value of that parameter was not supported. The value is not in an enumeration.

- This message shall be used to indicate that a parameter was given the correct value type but the value of that parameter was not supported.

Version Added: 1.10.0

Severity: Warning

Resolution: Choose a value from the enumeration list that the implementation can support and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the parameter <Arg2> in the action <Arg3> is not in the list of acceptable values."

1. *string*: The value provided for the action parameter.
 - This argument shall contain the value provided for the action parameter.
2. *string*: The name of the action parameter.
 - This argument shall contain the name of the action parameter.
3. *string*: The name of the action.
 - This argument shall contain the name of the action.

2.1.16 ActionParameterValueTypeError

Indicates that a parameter was given the wrong value type, such as when a number is supplied for a parameter that requires a string.

- This message shall be used to indicate that a parameter was given the wrong value type, such as when a number is supplied for a parameter that requires a string.

Version Added: 1.0.0

Severity: Warning

Resolution: Correct the value for the parameter in the request body and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the parameter <Arg2> in the action <Arg3> is of a different type than the parameter can accept."

1. *string*: The value provided for the action parameter.
 - This argument shall contain the value provided for the action parameter. Numeric values shall be converted to a string, and null values shall be represented by the string `null`.
2. *string*: The name of the action parameter.
 - This argument shall contain the name of the action parameter.
3. *string*: The name of the action.
 - This argument shall contain the name of the action.

2.1.17 ChassisPowerStateOffRequired

Indicates that the request requires a specified chassis to be powered off.

- This message shall be used to indicate that the request requires the specified chassis to be powered off as indicated by its `PowerState` property.

Version Added: 1.6.0

Severity: Warning

Resolution: Power off the specified chassis and resubmit the request.

Message and Arguments: "The Chassis with Id '<Arg1>' requires to be powered off to perform this request."

1. *string*: The `Id` of the chassis that requires to be powered off.
 - The value of this argument shall be a string containing the value of the `Id` property of the chassis that requires to be powered off.

2.1.18 ChassisPowerStateOnRequired

Indicates that the request requires a specified chassis to be powered on.

- This message shall be used to indicate that the request requires the specified chassis to be powered on as indicated by its `PowerState` property.

Version Added: 1.6.0

Severity: Warning

Resolution: Power on the specified chassis and resubmit the request.

Message and Arguments: "The chassis with Id '<Arg1>' requires to be powered on to perform this request."

1. *string*: The Id of the chassis that requires to be powered on.
 - The value of this argument shall be a string containing the value of the `Id` property of the chassis that requires to be powered on.

2.1.19 ConditionInRelatedResource

Indicates that one or more fault or error conditions exist in a related resource.

- This message shall be used to indicate that one or more conditions that require attention exist in a resource that affects the Health or HealthRollup of this resource. This message can be used in place of multiple messages in a `Conditions` property to roll up conditions from a single resource. The resource can be located using the `OriginOfCondition` property. The value of the `MessageSeverity` property should match the highest severity of any conditions in the resource referenced by the `OriginOfCondition` property.

Version Added: 1.9.0

Severity: Warning

Resolution: Check the Conditions array in the resource shown in the OriginOfCondition property to determine the conditions that need attention.

Message and Arguments: "One or more conditions exist in a related resource. See the OriginOfCondition property."

2.1.20 CouldNotEstablishConnection

Indicates that the attempt to access the resource, file, or image at the URI was unsuccessful because a session could not be established.

- This message shall be used to indicate that the attempt to access the resource, file, or image at the URI was unsuccessful because a session could not be established.

Version Added: 1.0.0

Severity: Critical

Resolution: Ensure that the URI contains a valid and reachable node name, protocol information and other URI components.

Message and Arguments: "The service failed to establish a connection with the URI '<Arg1>'."

1. *string*: The URI provided.
 - This argument shall contain the URI provided for this operation.

2.1.21 CreateFailedMissingReqProperties

Indicates that a create was attempted on a resource but that properties that are required for the create operation were missing from the request.

- This message shall be used to indicate that a create was attempted on a resource but that properties that are required for the create operation were missing from the request.

Version Added: 1.0.0

Severity: Critical

Resolution: Correct the body to include the required property with a valid value and resubmit the request if the operation failed.

Message and Arguments: "The create operation failed because the required property <Arg1> was missing from the request."

1. *string*: The name of the required property.
 - This argument shall contain the name of the required property.

2.1.22 CreateLimitReachedForResource

Indicates that no more resources can be created on the resource as it has reached its create limit.

- This message shall be used to indicate that no more resources can be created on the resource as it has reached its create limit.

Version Added: 1.0.0

Severity: Critical

Resolution: Either delete resources and resubmit the request if the operation failed or do not resubmit the request.

Message and Arguments: "The create operation failed because the resource has reached the limit of possible resources."

2.1.23 Created

Indicates that all conditions of a successful create operation have been met.

- This message shall be used to indicate that all conditions of a successful create operation have been met.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource has been created successfully."

2.1.24 EmptyJSON

Indicates that the request body contained an empty JSON object when one or more properties are expected in the body.

- This message shall be used to indicate that the request body contained an empty JSON object when one or more properties are expected in the body.

Version Added: 1.2.0

Severity: Warning

Resolution: Add properties in the JSON object and resubmit the request.

Message and Arguments: "The request body submitted contained an empty JSON object and the service is unable to process it."

2.1.25 EventBufferExceeded

Indicates undelivered events may have been lost due to a lack of buffer space in the service.

- This message shall be used to indicate that event buffer space used by the service has been exceeded, and as a result, one or more undelivered events may have been lost.

Version Added: 1.13.0

Severity: Warning

Resolution: None.

Message and Arguments: "Undelivered events may have been lost due to exceeding the event buffer."

2.1.26 EventSubscriptionLimitExceeded

Indicates that a event subscription establishment has been requested but the operation failed due to the number of simultaneous connection exceeding the limit of the implementation.

- This message shall be used to indicate that a event subscription establishment has been requested but the operation failed due to the number of simultaneous connection exceeding the limit of the implementation.

Version Added: 1.0.0

Severity: Critical

Resolution: Reduce the number of other subscriptions before trying to establish the event subscription or increase the limit of simultaneous subscriptions, if supported.

Message and Arguments: "The event subscription failed due to the number of simultaneous subscriptions exceeding the limit of the implementation."

2.1.27 GeneralError

Indicates that a general error has occurred. Use in `@Message.ExtendedInfo` is discouraged. When used in `@Message.ExtendedInfo`, implementations are expected to include a `Resolution` property with this message and provide a service-defined resolution to indicate how to resolve the error.

- This message shall be used to indicate that a general error has occurred. Implementations should not use this message in `@Message.ExtendedInfo`. If used in `@Message.ExtendedInfo`, implementations should include a `Resolution` property with this message and provide a service-defined resolution to indicate how to resolve the error.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "A general error has occurred. See Resolution for information on how to resolve the error, or `@Message.ExtendedInfo` if Resolution is not provided."

2.1.28 HeaderInvalid

Indicates that a request header is invalid.

- This message shall be used to indicate that a request header is known, but is invalid.

Version Added: 1.13.0

Severity: Critical

Resolution: Resubmit the request with a valid request header.

Message and Arguments: "Header '<Arg1>' is invalid."

1. *string*: The request header.
 - This argument shall contain the full request header, including the field name and field value.

2.1.29 HeaderMissing

Indicates that a required request header is missing.

- This message shall be used to indicate that a required request header is missing in the request.

Version Added: 1.13.0

Severity: Critical

Resolution: Resubmit the request with the required request header.

Message and Arguments: "Required header '<Arg1>' is missing in the request."

1. *string*: The missing required request header.
 - This argument shall contain the missing required request header.

2.1.30 InsufficientPrivilege

Indicates that the credentials associated with the established session do not have sufficient privileges for the requested operation.

- This message shall be used to indicate that the credentials associated with the established session do not have sufficient privileges for the requested operation.

Version Added: 1.0.0

Severity: Critical

Resolution: Either abandon the operation or change the associated access rights and resubmit the request if the operation failed.

Message and Arguments: "There are insufficient privileges for the account or credentials associated with the current session to perform the requested operation."

2.1.31 InsufficientStorage

Indicates that the operation could not be completed due to a lack of storage or memory available to the service.

- This message shall be used to indicate that the request exceeds the storage or memory available to the service. For cases where the payload is larger than the maximum size supported, regardless of available storage, the message `PayloadTooLarge` shall be used.

Version Added: 1.13.0

Severity: Critical

Resolution: Increase the free storage space available to the service and resubmit the request.

Message and Arguments: "Insufficient storage or memory available to complete the request."

2.1.32 InternalError

Indicates that the request failed for an unknown internal error but that the service is still operational.

- This message shall be used to indicate that the request failed for an unknown internal error but that the service is still operational.

Version Added: 1.0.0

Severity: Critical

Resolution: Resubmit the request. If the problem persists, consider resetting the service.

Message and Arguments: "The request failed due to an internal service error. The service is still operational."

2.1.33 InvalidIndex

The index is not valid.

- This message shall be used to indicate that the index is not valid.

Version Added: 1.0.0

Severity: Warning

Resolution: Verify the index value provided is within the bounds of the array.

Message and Arguments: "The index <Arg1> is not a valid offset into the array."

1. *number*: The array index.
 - This argument shall contain the array index value provided.

2.1.34 InvalidJSON

Indicates that the request body contains invalid JSON.

- This message shall be used to indicate that the request body contains invalid JSON.

Version Added: 1.10.0

Severity: Critical

Resolution: Ensure that the request body is valid JSON and resubmit the request.

Message and Arguments: "The request body submitted is invalid JSON starting at line <Arg1> and could not be parsed by the receiving service."

1. *number*: The line number where the invalid JSON starts.
 - The value of this argument shall contain the line number where the invalid JSON starts.

2.1.35 InvalidObject

Indicates that the object in question is invalid according to the implementation. Examples include a firmware update malformed URI.

- This message shall be used to indicate that the object in question is invalid according to the implementation.

Version Added: 1.0.0

Severity: Critical

Resolution: Either the object is malformed or the URI is not correct. Correct the condition and resubmit the request if it failed.

Message and Arguments: "The object at '<Arg1>' is invalid."

1. *string*: The URI provided.
 - This argument shall contain the URI provided for this operation.

2.1.36 InvalidURI

Indicates that the operation encountered a URI that does not correspond to a valid resource.

- This message shall be used to indicate that the operation encountered a URI that does not correspond to a valid resource.

Version Added: 1.10.0

Severity: Critical

Resolution: Provide a valid URI and resubmit the request.

Message and Arguments: "The URI <Arg1> was not found."

1. *string*: The supplied URI.
 - This argument shall contain the invalid URI that was encountered.

2.1.37 MalformedJSON

Indicates that the request body was malformed JSON.

- This message shall be used to indicate that the request body was malformed JSON.

Version Added: 1.0.0

Severity: Critical

Resolution: Ensure that the request body is valid JSON and resubmit the request.

Message and Arguments: "The request body submitted was malformed JSON and could not be parsed by the receiving service."

2.1.38 MaximumErrorsExceeded

Indicates that sufficient errors have occurred that the reporting service cannot return them all.

- The maximum number of errors that the implementation can report has been exceeded. This shall not be the only message reported by the service. The use of this message should be limited to RDE implementations.

Version Added: 1.8.0

Severity: Critical

Resolution: Resolve other reported errors and retry the current operation.

Message and Arguments: "Too many errors have occurred to report them all."

2.1.39 MissingOrMalformedPart

Indicates that a multipart request is missing a required part or contains malformed parts.

- This message shall be used to indicate that the request payload supplied as part of an HTTP multipart request is missing one or more required parts or contains malformed parts.

Version Added: 1.13.0

Severity: Critical

Resolution: Add any missing required parts or correct the malformed parts and resubmit the request.

Message and Arguments: "The multipart request contains malformed parts or is missing required parts."

2.1.40 NetworkNameResolutionNotConfigured

Indicates that network-based name resolution has not been configured on the service.

- This message shall be used to indicate that network-based name resolution, such as DNS or WINS protocols, has not been configured on the service.

Version Added: 1.11.0

Severity: Warning

Resolution: Configure the network name resolution protocol support on this service, or update any URI values to include an IP address instead of a network name and resubmit the request.

Message and Arguments: "Network name resolution has not been configured on this service."

2.1.41 NetworkNameResolutionNotSupported

Indicates the service does not support network-based name resolution.

- This message shall be used to indicate that the service does not support network-based name resolution, such as DNS or WINS protocols. URIs provided as property values must contain an IP address as the service cannot resolve a network name.

Version Added: 1.11.0

Severity: Warning

Resolution: Update any URI values to include an IP address instead of a network name and resubmit the request.

Message and Arguments: "Resolution of network-based names is not supported by this service."

2.1.42 NoOperation

Indicates that the requested operation will not perform any changes on the service.

- This message shall be used to indicate that the requested operation will not perform any changes on the service.

Version Added: 1.4.0

Severity: Warning

Resolution: Add properties in the JSON object and resubmit the request.

Message and Arguments: "The request body submitted contain no data to act upon and no changes to the resource took place."

2.1.43 NoValidSession

Indicates that the operation failed because a valid session is required in order to access any resources.

- This message shall be used to indicate that the operation failed because a valid session is required in order to access any resources.

Version Added: 1.0.0

Severity: Critical

Resolution: Establish a session before attempting any operations.

Message and Arguments: "There is no valid session established with the implementation."

2.1.44 OperationFailed

Indicates that one of the internal operations necessary to complete the request failed. Examples of this are when an internal service provider is unable to complete the request, such as in aggregation or RDE.

- This message shall be used to indicate that one of the internal operations necessary to complete the request failed. Partial results of the client operation may be returned.

Version Added: 1.8.0

Severity: Warning

Resolution: Resubmit the request. If the problem persists, consider resetting the service or provider.

Message and Arguments: "An error occurred internal to the service as part of the overall request. Partial results may have been returned."

2.1.45 OperationNotAllowed

Indicates that the HTTP method in the request is not allowed on this resource.

- This message shall be used to indicate that the HTTP method in the request is not allowed on this resource. This error message should be used together with status code 405 Method Not Allowed.

Version Added: 1.13.0

Severity: Critical

Resolution: None.

Message and Arguments: "The HTTP method is not allowed on this resource."

2.1.46 OperationTimeout

Indicates that one of the internal operations necessary to complete the request timed out. Examples of this are when an internal service provider is unable to complete the request, such as in aggregation or RDE.

- This message shall be used to indicate that one of the internal operations necessary to complete the request timed out. Partial results of the client operation may be returned.

Version Added: 1.8.0

Severity: Warning

Resolution: Resubmit the request. If the problem persists, consider resetting the service or provider.

Message and Arguments: "A timeout internal to the service occurred as part of the request. Partial results may have been returned."

2.1.47 PasswordChangeRequired

Indicates that the password for the account provided must be changed before accessing the service. The password can be changed with a PATCH to the `Password` property in the manager account resource instance. Implementations that provide a default password for an account may require a password change prior to first access to the service.

- This message shall be used to indicate that the password for the account provided must be changed before accessing the service. The password can be changed with a PATCH to the `Password` property in the manager

account resource instance. Implementations that provide a default password for an account may require a password change prior to first access to the service.

Version Added: 1.5.0

Severity: Critical

Resolution: Change the password for this account using a PATCH to the Password property at the URI provided.

Message and Arguments: "The password provided for this account must be changed before access is granted. PATCH the Password property for this account located at the target URI '<Arg1>' to complete this process."

1. *string*: The URI of the target resource.
 - This argument shall contain the URI of the target manager account resource instance.

2.1.48 PayloadTooLarge

Indicates that the supplied payload is too large to be accepted by the service.

- This message shall be used to indicate that the payload supplied exceeds the size supported by the service. For cases where the payload size is supported, but there is insufficient available storage space, the message `InsufficientStorage` shall be used.

Version Added: 1.13.0

Severity: Critical

Resolution: Check that the supplied payload is correct and supported by this service.

Message and Arguments: "The supplied payload exceeds the maximum size supported by the service."

2.1.49 PreconditionFailed

Indicates that the ETag supplied did not match the current ETag of the resource.

- Indicates that the ETag supplied either in the header or using that `@odata.etag` annotation in the resource did not match the one expected by the service and thus the operation on this resource failed. The use of this message shall only be used in responses for deep operations, since HTTP status code 412 is typically used for this.

Version Added: 1.8.0

Severity: Critical

Resolution: Try the operation again using the appropriate ETag.

Message and Arguments: "The ETag supplied did not match the ETag required to change this resource."

2.1.50 PreconditionRequired

Indicates that the request did not provide the required precondition such as an `If-Match` or `If-None-Match` header, or `@odata.etag` annotations.

- Indicates that the request did not provide the required precondition such as an `If-Match` or `If-None-Match` header, or `@odata.etag` annotations. The use of this message shall only be used in responses for deep operations, since HTTP status code 428 is typically used for this.

Version Added: 1.8.0

Severity: Critical

Resolution: Try the operation again using an `If-Match` or `If-None-Match` header and appropriate ETag.

Message and Arguments: "A precondition header or annotation is required to change this resource."

2.1.51 PropertyDeprecated

Indicates the property is deprecated.

- This message shall be used to indicate that the property is deprecated.

Version Added: 1.10.0

Severity: Warning

Resolution: Refer to the schema guide for more information.

Message and Arguments: "The deprecated property <Arg1> was included in the request body."

1. *string*: The name of the property.
 - This argument shall contain the name of the property that is deprecated.

2.1.52 PropertyDuplicate

Indicates that a duplicate property was included in the request body.

- This message shall be used to indicate that a duplicate property was included in the request body.

Version Added: 1.0.0

Severity: Warning

Resolution: Remove the duplicate property from the request body and resubmit the request if the operation failed.

Message and Arguments: "The property <Arg1> was duplicated in the request."

1. *string*: The name of the duplicate property.
 - This argument shall contain the name of the duplicate property.

2.1.53 PropertyMissing

Indicates that a required property was not supplied as part of the request.

- This message shall be used to indicate that a required property was not supplied as part of the request.

Version Added: 1.0.0

Severity: Warning

Resolution: Ensure that the property is in the request body and has a valid value and resubmit the request if the operation failed.

Message and Arguments: "The property <Arg1> is a required property and must be included in the request."

1. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.54 PropertyNotUpdated

Indicates that a property was not updated due to an internal service error, but the service is still functional.

- This message shall be used to indicate that a property was not updated due to an internal service error, but the service is still functional.

Version Added: 1.12.0

Severity: Critical

Resolution: Resubmit the request. If the problem persists, check for additional messages and consider resetting the service.

Message and Arguments: "The property <Arg1> was not updated due to an internal service error. The service is still operational."

1. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.55 PropertyNotWritable

Indicates that a property was given a value in the request body, but the property is a readonly property.

- This message shall be used to indicate that a property was given a value in the request body, but the property is a readonly property.

Version Added: 1.0.0

Severity: Warning

Resolution: Remove the property from the request body and resubmit the request if the operation failed.

Message and Arguments: "The property <Arg1> is a read only property and cannot be assigned a value."

1. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.56 PropertyUnknown

Indicates that an unknown property was included in the request body.

- This message shall be used to indicate that an unknown property was included in the request body.

Version Added: 1.0.0

Severity: Warning

Resolution: Remove the unknown property from the request body and resubmit the request if the operation failed.

Message and Arguments: "The property <Arg1> is not in the list of valid properties for the resource."

1. *string*: The name of the unknown property.
 - This argument shall contain the name of the unknown property.

2.1.57 PropertyValueConflict

Indicates that the requested write of a property value could not be completed, because of a conflict with another property value.

- This message shall be used to indicate that the requested write of a property could not be completed, because of a conflict with the value of another property.

Version Added: 1.6.0

Severity: Warning

Resolution: No resolution is required.

Message and Arguments: "The property '<Arg1>' could not be written because its value would conflict with the value of the '<Arg2>' property."

1. *string*: The name of the property for which a write was requested.
 - This argument shall contain the name of the property for which a write was requested.
2. *string*: The name of the property with which there is a conflict.
 - This argument shall contain the name of the property with which there is a conflict.

2.1.58 PropertyValueDeprecated

Indicates that a property was given a deprecated value.

- This message shall be used to indicate that a property was given a deprecated value.

Version Added: 1.10.0

Severity: Warning

Resolution: Refer to the schema guide for more information.

Message and Arguments: "The value '<Arg1>' for the property <Arg2> is deprecated."

1. *string*: The value provided for the property.
 - This argument shall contain the deprecated value provided for the property.
2. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.59 PropertyValueError

Indicates that a property was given an invalid value.

- This message shall be used to indicate that a property was given an invalid value, and the value cannot or should not be included in the message text.

Version Added: 1.11.0

Severity: Warning

Resolution: Correct the value for the property in the request body and resubmit the request if the operation failed.

Message and Arguments: "The value provided for the property <Arg1> is not valid."

1. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.60 PropertyValueExternalConflict

Indicates that the requested write of a property value could not be completed, due to the current state or configuration of the resource. This can include configuration conflicts with other resources or parameters that are not exposed by this interface.

- This message shall be used to indicate that the requested write of a property could not be completed, because the requested value of the property, which is a value supported by the implementation, cannot be accepted due to conflicts in the state or configuration of the resource.

Version Added: 1.10.0

Severity: Warning

Resolution: No resolution is required.

Message and Arguments: "The property '<Arg1>' with the requested value of '<Arg2>' could not be written because the value is not available due to a configuration conflict."

1. *string*: The name of the property for which a write was requested.
 - This argument shall contain the name of the property for which a write was requested.
2. *string*: The value of the property that is in conflict.
 - This argument shall contain the value of the property that is in conflict.

2.1.61 PropertyValueFormatError

Indicates that a property was given the correct value type but the value of that property was not supported.

- This message shall be used to indicate that a property was given the correct value type but the value of that property was not supported.

Version Added: 1.0.0

Severity: Warning

Resolution: Correct the value for the property in the request body and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the property <Arg2> is of a different format than the property can accept."

1. *string*: The value provided for the property.
 - This argument shall contain the value provided for the property. Numeric values shall be converted to a string, and null values shall be represented by the string `null`.
2. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.62 PropertyValueIncorrect

Indicates that the requested write of a property value could not be completed, because of an incorrect value of the property. Examples include values that do not match a regular expression requirement or passwords that do not match the implementation constraints.

- This message shall be used to indicate that the requested write of a property could not be completed, because of an incorrect value of the property such as when it does not meet the constraints of the implementation or match the regular expression requirements.

Version Added: 1.7.0

Severity: Warning

Resolution: No resolution is required.

Message and Arguments: "The property '<Arg1>' with the requested value of '<Arg2>' could not be written because the value does not meet the constraints of the implementation."

1. *string*: Name of the property for which a write was requested.
 - This argument shall contain the name of the property for which a write was requested.
2. *string*: Value of the property that is incorrect.
 - This argument shall contain the value of the property that is incorrect.

2.1.63 PropertyValueModified

Indicates that a property was given the correct value type but the value of that property was modified. Examples are truncated or rounded values.

- This message shall be used to indicate that a property was given the correct value type but the value of that property was modified.

Version Added: 1.0.0

Severity: Warning

Resolution: No resolution is required.

Message and Arguments: "The property <Arg1> was assigned the value '<Arg2>' due to modification by the service."

1. *string*: The name of the property.
 - This argument shall contain the name of the property.
2. *string*: The value assigned for property.
 - This argument shall contain the value assigned for property.

2.1.64 PropertyValueNotInList

Indicates that a property was given the correct value type but the value of that property was not supported. The value is not in an enumeration.

- This message shall be used to indicate that a property was given the correct value type but the value of that property was not supported.

Version Added: 1.0.0

Severity: Warning

Resolution: Choose a value from the enumeration list that the implementation can support and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the property <Arg2> is not in the list of acceptable values."

1. *string*: The value provided for the property.
 - This argument shall contain the value provided for the property.
2. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.65 PropertyValueOutOfRange

Indicates that a property was given the correct value type but the value of that property is outside the supported range.

- This value shall be used to indicate that a property was given the correct value type but the value of that property is outside the supported range.

Version Added: 1.3.0

Severity: Warning

Resolution: Correct the value for the property in the request body and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the property <Arg2> is not in the supported range of acceptable values."

1. *string*: The value provided for the property.
 - This argument shall contain the value provided for the property.
2. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.66 PropertyValueResourceConflict

Indicates that the requested write of a property value could not be completed, due to the current state or configuration of another resource.

- This message shall be used to indicate that the requested write of a property could not be completed, because the requested value of the property, which is a value supported by the implementation, cannot be accepted due to conflicts in the state or configuration of another resource.

Version Added: 1.10.0

Severity: Warning

Resolution: No resolution is required.

Message and Arguments: "The property '<Arg1>' with the requested value of '<Arg2>' could not be written because the value conflicts with the state or configuration of the resource at '<Arg3>'."

1. *string*: The name of the property for which a write was requested.
 - This argument shall contain the name of the property for which a write was requested.
2. *string*: The value of the property that is in conflict.
 - This argument shall contain the value of the property that is in conflict.
3. *string*: The URI of the resource in conflict.
 - This argument shall contain the URI of the resource that is in conflict with the value of the requested property.

2.1.67 PropertyValueTypeError

Indicates that a property was given the wrong value type, such as when a number is supplied for a property that requires a string.

- This message shall be used to indicate that a property was given the wrong value type.

Version Added: 1.0.0

Severity: Warning

Resolution: Correct the value for the property in the request body and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the property <Arg2> is of a different type than the property can accept."

1. *string*: The value provided for the property.
 - This argument shall contain the value provided for the property. Numeric values shall be converted to a string, and null values shall be represented by the string `null`.
2. *string*: The name of the property.
 - This argument shall contain the name of the property.

2.1.68 QueryCombinationInvalid

Indicates the request contains multiple query parameters, and that two or more of them cannot be used together.

- This message shall be used to indicate that the request contains multiple query parameters, and that two or more of them cannot be used together.

Version Added: 1.6.0

Severity: Warning

Resolution: Remove one or more of the query parameters and resubmit the request if the operation failed.

Message and Arguments: "Two or more query parameters in the request cannot be used together."

2.1.69 QueryNotSupported

Indicates that query is not supported on the implementation.

- This message shall be used to indicate that query is not supported on the implementation.

Version Added: 1.0.0

Severity: Warning

Resolution: Remove the query parameters and resubmit the request if the operation failed.

Message and Arguments: "Querying is not supported by the implementation."

2.1.70 QueryNotSupportedOnOperation

Indicates that query is not supported with the given operation, such as when the `$expand` query is attempted with a PATCH operation.

- This message shall be used to indicate that query is not supported with the given operation.

Version Added: 1.6.0

Severity: Warning

Resolution: Remove the query parameters and resubmit the request if the operation failed.

Message and Arguments: "Querying is not supported with the requested operation."

2.1.71 QueryNotSupportedOnResource

Indicates that query is not supported on the given resource, such as when the `$skip` query is attempted on a resource that is not a collection.

- This message shall be used to indicate that query is not supported on the given resource.

Version Added: 1.0.0

Severity: Warning

Resolution: Remove the query parameters and resubmit the request if the operation failed.

Message and Arguments: "Querying is not supported on the requested resource."

2.1.72 QueryParameterOutOfRange

Indicates that a query parameter was provided that is out of range for the given resource. This can happen with values that are too low or beyond that possible for the supplied resource, such as when a page is requested that is beyond the last page.

- This message shall be used to indicate that a query parameter was provided that is out of range for the given resource.

Version Added: 1.0.0

Severity: Warning

Resolution: Reduce the value for the query parameter to a value that is within range, such as a start or count value that is within bounds of the number of resources in a collection or a page that is within the range of valid pages.

Message and Arguments: "The value '<Arg1>' for the query parameter <Arg2> is out of range <Arg3>."

1. *string*: The value provided for the query parameter.
 - This argument shall contain the value provided for the query parameter.

2. *string*: The query parameter.
 - This argument shall contain the name of the query parameter.
3. *string*: The valid range for the query parameter.
 - This argument shall contain a string representation of the valid range of values supported for the query parameter.

2.1.73 QueryParameterValueError

Indicates that a query parameter was given an invalid value.

- This message shall be used to indicate that a query parameter was given an invalid value that can not or should not be included in the message.

Version Added: 1.11.0

Severity: Warning

Resolution: Correct the value for the query parameter in the request and resubmit the request if the operation failed.

Message and Arguments: "The value for the parameter <Arg1> is invalid."

1. *string*: The query parameter.
 - This argument shall contain the name of the query parameter.

2.1.74 QueryParameterValueFormatError

Indicates that a query parameter was given the correct value type but the value of that parameter was not supported. This includes the value size or length has been exceeded.

- This message shall be used to indicate that a query parameter was given the correct value type but the value of that parameter was not supported.

Version Added: 1.0.0

Severity: Warning

Resolution: Correct the value for the query parameter in the request and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the parameter <Arg2> is of a different format than the parameter can accept."

1. *string*: The value provided for the query parameter.
 - This argument shall contain the value provided for the query parameter.
2. *string*: The query parameter.

- This argument shall contain the name of the query parameter.

2.1.75 QueryParameterValueTypeError

Indicates that a query parameter was given the wrong value type, such as when a number is supplied for a query parameter that requires a string.

- This message shall be used to indicate that a query parameter was given the wrong value type, such as when a number is supplied for a query parameter that requires a string.

Version Added: 1.0.0

Severity: Warning

Resolution: Correct the value for the query parameter in the request and resubmit the request if the operation failed.

Message and Arguments: "The value '<Arg1>' for the query parameter <Arg2> is of a different type than the parameter can accept."

1. *string*: The value provided for the query parameter.
 - This argument shall contain the value provided for the query parameter.
2. *string*: The query parameter.
 - This argument shall contain the name of the query parameter.

2.1.76 ResetRecommended

Indicates that a component reset is recommended for error recovery while unaffected applications can continue running without any effects on accuracy and performance.

- This message shall be used to indicate that a component reset is recommended for error recovery while unaffected applications can continue running without any effects on accuracy and performance.

Version Added: 1.12.0

Severity: Warning

Resolution: Perform the recommended reset action on the specified component.

Message and Arguments: "In order to recover from errors, a component reset is recommended with the Reset action URI '<Arg1>' and ResetType '<Arg2>'."

1. *string*: The URI for the reset action of the component that recommends a reset.
 - This argument shall contain the URI for the reset action that is recommended to be invoked for error recovery.

2. *string*: The `ResetType` required in a reset action on the component URI.
 - This argument shall contain the `ResetType` required in a reset action on the component URI for error recovery while unaffected applications can continue running without any effects on accuracy and performance. A reset action POST shall contain this `ResetType` parameter.

2.1.77 ResetRequired

Indicates that a component reset is required for changes, error recovery, or operations to complete.

- This message shall be used to indicate that a component reset is required for changes, error recovery, or operations to complete.

Version Added: 1.6.0

Severity: Warning

Resolution: Perform the required reset action on the specified component.

Message and Arguments: "In order to apply changes, recover from errors, or complete the operation, a component reset is required with the Reset action URI '<Arg1>' and ResetType '<Arg2>'."

1. *string*: The URI for the reset action of the component that requires a reset.
 - This argument shall contain the URI for the reset action that is required to be invoked for changes, error recovery or operations to complete.
2. *string*: The `ResetType` required in a reset action on the component URI.
 - This argument shall contain the `ResetType` required in a reset action on the component URI for changes, error recovery or operations to complete. A reset action POST shall contain this `ResetType` parameter.

2.1.78 ResourceAlreadyExists

Indicates that a resource change or creation was attempted but that the operation cannot proceed because the resource already exists.

- This message shall be used to indicate that a resource change or creation was attempted but that the operation cannot proceed because the resource already exists.

Version Added: 1.0.0

Severity: Critical

Resolution: Do not repeat the create operation as the resource has already been created.

Message and Arguments: "The requested resource of type <Arg1> with the property <Arg2> with the value '<Arg3>' already exists."

1. *string*: The type of resource.
 - This argument shall contain the schema name of the resource.
2. *string*: The name of the property.
 - This argument shall contain the name of the property.
3. *string*: The value of the property.
 - This argument shall contain the value provided for the property. Numeric values shall be converted to a string, and null values shall be represented by the string `null`.

2.1.79 ResourceAtUriInUnknownFormat

Indicates that the URI was valid but the resource or image at that URI was in a format not supported by the service.

- This message shall be used to indicate that the URI was valid but the resource or image at that URI was in a format not supported by the service.

Version Added: 1.0.0

Severity: Critical

Resolution: Place an image or resource or file that is recognized by the service at the URI.

Message and Arguments: "The resource at '<Arg1>' is in a format not recognized by the service."

1. *string*: The URI provided.
 - This argument shall contain the URI provided for this operation.

2.1.80 ResourceAtUriUnauthorized

Indicates that the attempt to access the resource, file, or image at the URI was unauthorized.

- This message shall be used to indicate that the attempt to access the resource, file, or image at the URI was unauthorized.

Version Added: 1.0.0

Severity: Critical

Resolution: Ensure that the appropriate access is provided for the service in order for it to access the URI.

Message and Arguments: "While accessing the resource at '<Arg1>', the service received an authorization error '<Arg2>'."

1. *string*: The URI provided.
 - This argument shall contain the URI provided for this operation.
2. *string*: The error message received from the authorization service.
 - This argument shall contain the error message received from the authorization service or software.

2.1.81 ResourceCannotBeDeleted

Indicates that a delete operation was attempted on a resource that cannot be deleted.

- This message shall be used to indicate that a delete operation was attempted on a resource that cannot be deleted.

Version Added: 1.0.0

Severity: Critical

Resolution: Do not attempt to delete a non-deletable resource.

Message and Arguments: "The delete request failed because the resource requested cannot be deleted."

2.1.82 ResourceCreationConflict

Indicates that the requested resource creation could not be completed because the service has a resource that conflicts with the request.

- This message shall be used to indicate that the requested creation of a resource could not be completed because the service has a resource that conflicts with the request, such as when the resource already exists or an existing resource has similar property values.

Version Added: 1.7.0

Severity: Warning

Resolution: No resolution is required.

Message and Arguments: "The resource could not be created. The service has a resource at URI '<Arg1>' that conflicts with the creation request."

1. *string*: URI of the resource that conflicts with the creation request.
 - This argument shall contain the URI of the resource with which there is a conflict.

2.1.83 ResourceDeprecated

Indicates the resource is deprecated.

- This message shall be used to indicate that the resource is deprecated.

Version Added: 1.10.0

Severity: Warning

Resolution: Refer to the schema guide for more information.

Message and Arguments: "The operation was performed on a deprecated resource '<Arg1>'."

1. *string*: The URI of the resource.
 - This argument shall contain the URI of the resource that is deprecated.

2.1.84 ResourceExhaustion

Indicates that a resource could not satisfy the request due to some unavailability of resources. An example is that available capacity has been allocated.

- This message shall be used to indicate that a resource could not satisfy the request due to some unavailability of resources.

Version Added: 1.2.0

Severity: Critical

Resolution: Ensure that the resources are available and resubmit the request.

Message and Arguments: "The resource '<Arg1>' was unable to satisfy the request due to unavailability of resources."

1. *string*: The URI of the resource.
 - This argument shall contain the URI of the relevant resource.

2.1.85 ResourceInStandby

Indicates that the request could not be performed because the resource is in standby.

- This message shall be used to indicate that the request could not be performed because the resource is in standby.

Version Added: 1.2.0

Severity: Critical

Resolution: Ensure that the resource is in the correct power state and resubmit the request.

Message and Arguments: "The request could not be performed because the resource is in standby."

2.1.86 ResourceInUse

Indicates that a change was requested to a resource but the change was rejected due to the resource being in use or transition.

- This message shall be used to indicate that a change was requested to a resource but the change was rejected due to the resource being in use or transition.

Version Added: 1.0.0

Severity: Warning

Resolution: Remove the condition and resubmit the request if the operation failed.

Message and Arguments: "The change to the requested resource failed because the resource is in use or in transition."

2.1.87 ResourceMissingAtURI

Indicates that the operation expected an image or other resource at the provided URI but none was found. Examples of this are in requests that require URIs like firmware update.

- This message shall be used to indicate that the operation expected an image or other resource at the provided URI but none was found.

Version Added: 1.0.0

Severity: Critical

Resolution: Place a valid resource at the URI or correct the URI and resubmit the request.

Message and Arguments: "The resource at the URI '<Arg1>' was not found."

1. *string*: The URI provided.
 - This argument shall contain the URI provided for this operation.

2.1.88 ResourceNotFound

Indicates that the operation expected a resource identifier that corresponds to an existing resource but one was not found.

- This message shall be used to indicate that the operation expected a resource identifier that corresponds to an existing resource but one was not found.

Version Added: 1.2.0

Severity: Critical

Resolution: Provide a valid resource identifier and resubmit the request.

Message and Arguments: "The requested resource of type <Arg1> named '<Arg2>' was not found."

1. *string*: The type of resource.
 - This argument shall contain the schema name of the resource.
2. *string*: The `Id` of the resource.
 - This argument shall contain the value of the `Id` property of the requested resource.

2.1.89 ResourceTypeIncompatible

Indicates that the resource type of the operation does not match that for the operation destination. Examples of when this can happen include during a POST to a resource collection using the wrong resource type, an update where the `@odata.type` properties do not match, or on a major version incompatibility.

- This message shall be used to indicate that the resource type of the operation does not match that for the operation destination.

Version Added: 1.2.0

Severity: Critical

Resolution: Resubmit the request with a payload compatible with the resource's schema.

Message and Arguments: "The `@odata.type` of the request body <Arg1> is incompatible with the `@odata.type` of the resource, which is <Arg2>."

1. *string*: The `@odata.type` of the request.
 - This argument shall contain the value of the `@odata.type` property provided in the request.
2. *string*: The supported `@odata.type` value.
 - This argument shall contain the value of the `@odata.type` property supported by the resource.

2.1.90 RestrictedPrivilege

Indicates that the operation was not successful because a privilege is restricted.

- This message shall be used to indicate that the operation was not successful because a privilege is restricted.

Version Added: 1.9.0

Severity: Warning

Resolution: Remove restricted privileges from the request body and resubmit the request.

Message and Arguments: "The operation was not successful because the privilege '<Arg1>' is restricted."

1. *string*: The name of the restricted privilege.
 - This argument shall contain the name of the restricted privilege.

2.1.91 RestrictedRole

Indicates that the operation was not successful because the role is restricted.

- This message shall be used to indicate that the operation was not successful because the role is restricted.

Version Added: 1.9.0

Severity: Warning

Resolution: No resolution is required. For standard roles, consider using the role specified in the AlternateRoleId property in the Role resource.

Message and Arguments: "The operation was not successful because the role '<Arg1>' is restricted."

1. *string*: The name of the restricted role.
 - This argument shall contain the name of the restricted role.

2.1.92 ServiceDisabled

Indicates that the operation failed because the service, such as the account service, is disabled and cannot accept requests.

- This message shall be used to indicate that the operation failed because the service, such as the account service, is disabled and cannot accept requests.

Version Added: 1.11.0

Severity: Warning

Resolution: Enable the service and resubmit the request if the operation failed.

Message and Arguments: "The operation failed because the service at <Arg1> is disabled and cannot accept requests."

1. *string*: The URI of the disabled service.
 - This argument shall contain the URI of the disabled service.

2.1.93 ServiceInUnknownState

Indicates that the operation failed because the service is in an unknown state and cannot accept additional requests.

- This message shall be used to indicate that the operation failed because the service is in an unknown state and cannot accept additional requests.

Version Added: 1.0.0

Severity: Critical

Resolution: Restart the service and resubmit the request if the operation failed.

Message and Arguments: "The operation failed because the service is in an unknown state and can no longer take incoming requests."

2.1.94 ServiceShuttingDown

Indicates that the operation failed as the service is shutting down, such as when the service reboots.

- This message shall be used to indicate that the operation failed as the service is shutting down.

Version Added: 1.0.0

Severity: Critical

Resolution: When the service becomes available, resubmit the request if the operation failed.

Message and Arguments: "The operation failed because the service is shutting down and can no longer take incoming requests."

2.1.95 ServiceTemporarilyUnavailable

Indicates the service is temporarily unavailable.

- This message shall be used to indicate that the service is temporarily unavailable.

Version Added: 1.0.0

Severity: Critical

Resolution: Wait for the indicated retry duration and retry the operation.

Message and Arguments: "The service is temporarily unavailable. Retry in <Arg1> seconds."

1. *string*: The retry duration in seconds.
 - This argument shall contain the number of seconds the client should wait before attempting the operation again.

2.1.96 SessionLimitExceeded

Indicates that a session establishment has been requested but the operation failed due to the number of simultaneous sessions exceeding the limit of the implementation.

- This message shall be used to indicate that a session establishment has been requested but the operation failed due to the number of simultaneous sessions exceeding the limit of the implementation.

Version Added: 1.0.0

Severity: Critical

Resolution: Reduce the number of other sessions before trying to establish the session or increase the limit of simultaneous sessions, if supported.

Message and Arguments: "The session establishment failed due to the number of simultaneous sessions exceeding the limit of the implementation."

2.1.97 SessionTerminated

Indicates that the DELETE operation on the session resource resulted in the successful termination of the session.

- This message shall be used to indicate that the DELETE operation on the session resource resulted in the successful termination of the session.

Version Added: 1.3.0

Severity: OK

Resolution: No resolution is required.

Message and Arguments: "The session was successfully terminated."

2.1.98 SourceDoesNotSupportProtocol

Indicates that while attempting to access, connect to or transfer a resource, file, or image from another location that the other end of the connection did not support the protocol.

- This message shall be used to indicate that while attempting to access, connect to or transfer a resource, file, or image from another location that the other end of the connection did not support the protocol.

Version Added: 1.0.0

Severity: Critical

Resolution: Change protocols or URIs.

Message and Arguments: "The other end of the connection at '<Arg1>' does not support the specified protocol <Arg2>."

1. *string*: The URI provided.
 - This argument shall contain the URI provided for this operation.
2. *string*: The protocol specified.
 - This argument shall contain the name of the protocol requested for this operation.

2.1.99 StrictAccountTypes

Indicates the request failed because a set of `AccountTypes` or `OEMAccountTypes` was not accepted while `StrictAccountTypes` is set to `true`.

- This message shall be used to indicate that the request failed because a set of `AccountTypes` or `OEMAccountTypes` was not accepted while `StrictAccountTypes` is set to `true`.

Version Added: 1.9.0

Severity: Warning

Resolution: Resubmit the request either with an acceptable set of `AccountTypes` and `OEMAccountTypes` or with `StrictAccountTypes` set to `false`.

Message and Arguments: "The request was not possible to fulfill with the account types included in property '<Arg1>' and property `StrictAccountTypes` set to `true`."

1. *string*: The name of the property.
 - This argument shall contain the name of the property `AccountTypes` or `OEMAccountTypes`.

2.1.100 StringValueTooLong

Indicates that a string value passed to the given resource exceeded its length limit. An example is when a shorter limit is imposed by an implementation than that allowed by the specification.

- This message shall be used to indicate that a string value passed to the given resource exceeded its length limit.

Version Added: 1.2.0

Severity: Warning

Resolution: Resubmit the request with an appropriate string length.

Message and Arguments: "The string '<Arg1>' exceeds the length limit <Arg2>."

1. *string*: The string provided.
 - This argument shall contain the string value provided for this operation.
2. *number*: The maximum string length.
 - This argument shall contain the maximum supported string length for this property.

2.1.101 SubscriptionTerminated

An event subscription has been terminated by the service. No further events will be delivered.

- This message shall be used to indicate that an event subscription has been terminated by the service.

Version Added: 1.5.0

Severity: OK

Resolution: No resolution is required.

Message and Arguments: "The event subscription has been terminated."

2.1.102 Success

Indicates that all conditions of a successful operation have been met.

- This message shall be used to indicate that all conditions of a successful operation have been met.

Version Added: 1.0.0

Severity: OK

Resolution: None

Message and Arguments: "The request completed successfully."

2.1.103 UndeterminedFault

Indicates that a fault or error condition exists but the source of the fault cannot be determined or is unknown to the service.

- This message shall be used to indicate that a fault or error condition has been reported to the service, but the source of the fault cannot be determined or is unknown to the service.

Version Added: 1.9.0

Severity: Critical

Resolution: None.

Message and Arguments: "A undetermined fault condition has been reported by '<Arg1>'."

1. *string*: The software, provider, or service that reported the fault condition.
 - This argument shall contain the name of the entity that reported the fault condition.

2.1.104 UnrecognizedRequestBody

Indicates that the service encountered an unrecognizable request body that could not even be interpreted as malformed JSON.

- This message shall be used to indicate that the service encountered an unrecognizable request body that could not even be interpreted as malformed JSON.

Version Added: 1.0.0

Severity: Warning

Resolution: Correct the request body and resubmit the request if it failed.

Message and Arguments: "The service detected a malformed request body that it was unable to interpret."

2.2 Composition 1.1.0

This registry defines the messages for composition related events

Message	Severity	Description
ConstrainedResourceAlreadyReserved	Critical	Indicates that the requested resources are already reserved in response to a constrained composition request.
EmptyManifest	Warning	Indicates that the manifest contains no stanzas, or a stanza in the manifest contains no request.
IncompatibleZone	Critical	Indicates that not all referenced resource blocks are in the same resource zone.
NoResourceMatch	Critical	Indicates that the service could not find a matching resource based on the given parameters.
ResourceBlockChanged	OK	Indicates that a <code>ResourceBlock</code> has changed. This is not used whenever there is another event message for that specific change, such as only the state has changed.
ResourceBlockCompositionStateChanged	OK	Indicates that the composition state of a <code>ResourceBlock</code> has changed, specifically the value of the <code>CompositionState</code> property within <code>CompositionStatus</code> has changed.
ResourceBlockInUse	Warning	Indicates that the composition request contains a <code>ResourceBlock</code> that is unable to participate in more compositions.
ResourceBlockInvalid	Critical	Indicates that the <code>Id</code> of a referenced resource block is no longer valid.
ResourceBlockNotFound	Critical	Indicates that the referenced resource block is not found.
ResourceBlockStateChanged	OK	Indicates that the state of a <code>ResourceBlock</code> has changed, specifically the value of the <code>State</code> property within <code>Status</code> has changed.
ResourceZoneMembershipChanged	OK	Indicates that the membership of a resource zone has changed due to <code>ResourceBlock</code> s being added or removed from the resource zone.
SpecifiedResourceAlreadyReserved	Critical	Indicates that a resource block is already reserved in response to a specific composition request.
UnableToProcessStanzaRequest	Critical	Indicates that the manifest provided for the <code>Compose</code> action contains a stanza with <code>Content</code> that could not be processed.

2.2.1 ConstrainedResourceAlreadyReserved

Indicates that the requested resources are already reserved in response to a constrained composition request.

- This message shall be used to indicate that the requested resources are already reserved in response to a constrained composition request.

Version Added: 1.0.0

Severity: Critical

Resolution: Delete the reservation containing the resources and resubmit the request.

Message and Arguments: "The requested resources are reserved under reservation '<Arg1>'."

1. *string*: The `Id` of the reservation.
 - The value of this argument shall contain the value of the `Id` property of the reservation containing the requested resources.

2.2.2 EmptyManifest

Indicates that the manifest contains no stanzas, or a stanza in the manifest contains no request.

- This message shall be used to indicate that the manifest contains no stanzas, or a stanza in the manifest contains no request.

Version Added: 1.0.0

Severity: Warning

Resolution: Provide a request content for the manifest and resubmit.

Message and Arguments: "The provided manifest is empty or a stanza in the manifest contains no request."

2.2.3 IncompatibleZone

Indicates that not all referenced resource blocks are in the same resource zone.

- This message shall be used to indicate that not all referenced resource blocks are in the same resource zone.

Version Added: 1.0.0

Severity: Critical

Resolution: Request resource blocks from the same resource zone.

Message and Arguments: "The requested resource blocks span multiple resource zones."

2.2.4 NoResourceMatch

Indicates that the service could not find a matching resource based on the given parameters.

- This message shall be used to indicate that the service could not find a matching resource based on the given parameters.

Version Added: 1.0.0

Severity: Critical

Resolution: Change parameters associated with the resource, such as quantity or performance, and resubmit the request.

Message and Arguments: "The requested resources of type '<Arg1>' are not available for allocation."

1. *string*: The resource type requested.
 - This argument shall contain the schema name of the requested resource.

2.2.5 ResourceBlockChanged

Indicates that a `ResourceBlock` has changed. This is not used whenever there is another event message for that specific change, such as only the state has changed.

- This message shall be used to indicate that one or more properties of a `ResourceBlock` have changed. This message shall not be used for cases where a more specific message could be used in its place.

Version Added: 1.0.0

Severity: OK

Resolution: Refresh your cached version of the `ResourceBlock` to get the updated information from the service.

Message and Arguments: "The `ResourceBlock` '<Arg1>' has changed on the service."

1. *string*: The `Id` of the `ResourceBlock` that has changed.
 - The value of this argument shall contain the value of the `Id` property of the `ResourceBlock` that has changed.

2.2.6 ResourceBlockCompositionStateChanged

Indicates that the composition state of a `ResourceBlock` has changed, specifically the value of the `CompositionState` property within `CompositionStatus` has changed.

- This message shall be used to indicate that the `CompositionState` property within the `CompositionStatus` object of a `ResourceBlock` has changed.

Version Added: 1.0.0

Severity: OK

Resolution: Refresh your cached version of the `ResourceBlock` to get the updated information from the service.

Message and Arguments: "The composition status of the `ResourceBlock` '<Arg1>' has changed."

1. *string*: The `Id` of the `ResourceBlock` that has changed.
 - The value of this argument shall contain the value of the `Id` property of the `ResourceBlock` that has changed.

2.2.7 ResourceBlockInUse

Indicates that the composition request contains a `ResourceBlock` that is unable to participate in more compositions.

- This message shall be used to indicate that the composition request contains a `ResourceBlock` that is unable to participate in more compositions due to restrictions specified in its `CompositionStatus` property. For example, the `ResourceBlock` has reached its composition limit specified by the `MaxCompositions` property.

Version Added: 1.0.0

Severity: Warning

Resolution: Remove the `ResourceBlock` from the request and resubmit the request.

Message and Arguments: "The `ResourceBlock` with Id '<Arg1>' cannot be part of any new compositions."

1. *string*: The `Id` of the `ResourceBlock` that cannot be part of any new compositions.
 - The value of this argument shall contain the value of the `Id` property of the `ResourceBlock` that cannot be part of any new compositions.

2.2.8 ResourceBlockInvalid

Indicates that the `Id` of a referenced resource block is no longer valid.

- This message shall be used to indicate that the `Id` of a referenced resource block is no longer valid.

Version Added: 1.0.0

Severity: Critical

Resolution: Remove the resource block and resubmit the request.

Message and Arguments: "Resource block '<Arg1>' is not valid."

1. *string*: The `Id` of the resource block.
 - The value of this argument shall contain the value of the `Id` property of the resource block that is no longer valid.

2.2.9 ResourceBlockNotFound

Indicates that the referenced resource block is not found.

- This message shall be used to indicate that the referenced resource block is not found.

Version Added: 1.0.0

Severity: Critical

Resolution: Remove the resource block and resubmit the request.

Message and Arguments: "Resource block '<Arg1>' is not found."

1. *string*: The URI of the resource block.
 - The value of this argument shall contain the value of the URI of the resource block that is not found.

2.2.10 ResourceBlockStateChanged

Indicates that the state of a `ResourceBlock` has changed, specifically the value of the `State` property within `Status` has changed.

- This message shall be used to indicate that the `State` property within the `Status` object of a `ResourceBlock` has changed.

Version Added: 1.0.0

Severity: OK

Resolution: Refresh your cached version of the `ResourceBlock` to get the updated information from the service.

Message and Arguments: "The state of the `ResourceBlock` '<Arg1>' has changed."

1. *string*: The `Id` of the `ResourceBlock` that has changed.
 - The value of this argument shall contain the value of the `Id` property of the `ResourceBlock` that has changed.

2.2.11 ResourceZoneMembershipChanged

Indicates that the membership of a resource zone has changed due to `ResourceBlock`s being added or removed from the resource zone.

- This message shall be used to indicate that entries found in the `ResourceBlocks` array within the `Links` object for a resource zone have been added or removed.

Version Added: 1.0.0

Severity: OK

Resolution: Refresh your cached version of the resource zone to get the updated information from the service.

Message and Arguments: "The membership of resource zone '<Arg1>' has been changed."

1. *string*: The `Id` of the resource zone that has changed.
 - The value of this argument shall contain the value of the `Id` property of the resource zone that has changed.

2.2.12 SpecifiedResourceAlreadyReserved

Indicates that a resource block is already reserved in response to a specific composition request.

- This message shall be used to indicate that a resource block is already reserved in response to a specific composition request.

Version Added: 1.0.0

Severity: Critical

Resolution: Delete the reservation containing the resource block or select a different resource block and resubmit the request.

Message and Arguments: "Resource block '<Arg1>' is already reserved under reservation '<Arg2>'."

1. *string*: The `Id` of the resource block.
 - The value of this argument shall contain the value of the `Id` property of the resource block that is already reserved.
2. *string*: The `Id` of the reservation.
 - The value of this argument shall contain the value of the `Id` property of the reservation containing the requested resource.

2.2.13 UnableToProcessStanzaRequest

Indicates that the manifest provided for the `Compose` action contains a stanza with `Content` that could not be processed.

- This message shall be used to indicate that the manifest provided for the `Compose` action contains a stanza with

`Content` that could not be processed.

Version Added: 1.0.0

Severity: Critical

Resolution: Add the `Content` parameter to the stanza or remove the stanza, and resubmit the request.

Message and Arguments: "The provided manifest for the `Compose` action of type <Arg1> contains a stanza with `Id` of value '<Arg2>' with a `Content` parameter that could not be processed."

1. *string*: The `RequestType` of the `Compose` action.
 - The value of this argument shall contain the value of the `RequestType` parameter of the `Compose` action.
2. *string*: The `Id` of the stanza.
 - The value of this argument shall contain the value of the `Id` parameter of the stanza.

2.3 EthernetFabric 1.0.0

This registry defines messages for Ethernet fabrics

Message	Severity	Description
LLDPInterfaceDisabled	Warning	Indicates that an interface has disabled Link Layer Discovery Protocol (LLDP).
LLDPInterfaceEnabled	OK	Indicates that an interface has enabled Link Layer Discovery Protocol (LLDP).
MLAGInterfaceDegraded	Warning	Indicates that multi-chassis link aggregation group (MLAG) interfaces have been established, but at an unexpectedly low aggregated link speed.
MLAGInterfaceDown	Warning	Indicates that the multi-chassis link aggregation group (MLAG) interface is down on a switch.
MLAGInterfacesUp	OK	Indicates that all multi-chassis link aggregation group (MLAG) interfaces are up.
MLAGPeerDown	Warning	Indicates that the multi-chassis link aggregation group (MLAG) peer is down.
MLAGPeerUp	OK	Indicates that the multi-chassis link aggregation group (MLAG) peer is up.
RoutingFailureThresholdExceeded	Warning	Indicates that a switch has encountered an unusually large number of routing errors.

2.3.1 LLDPInterfaceDisabled

Indicates that an interface has disabled Link Layer Discovery Protocol (LLDP).

- This message shall be used to indicate that an interface has disabled Link Layer Discovery Protocol (LLDP).

Version Added: 1.0.0

Severity: Warning

Resolution: Check that LLDP is enabled on device endpoints.

Message and Arguments: "LLDP has been disabled on switch '<Arg1>' port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch on which an LLDP connection has been disabled.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port on which an LLDP connection has been disabled.

2.3.2 LLDPInterfaceEnabled

Indicates that an interface has enabled Link Layer Discovery Protocol (LLDP).

- This message shall be used to indicate that an interface has enabled Link Layer Discovery Protocol (LLDP).

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "LLDP has been enabled on switch '<Arg1>' port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch on which an LLDP connection has been established.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port on which an LLDP connection has been established.

2.3.3 MLAGInterfaceDegraded

Indicates that multi-chassis link aggregation group (MLAG) interfaces have been established, but at an unexpectedly low aggregated link speed.

- This message shall be used to indicate that multi-chassis link aggregation group (MLAG) interfaces have been established, but the aggregated link speed is lower than provisioned.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "MLAG interface '<Arg1>' is degraded on switch '<Arg2>'."

1. *string*: The MLAG domain ID.
 - This argument shall contain the MLAG domain ID that is degraded.
2. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch for which an MLAG interface is degraded.

2.3.4 MLAGInterfaceDown

Indicates that the multi-chassis link aggregation group (MLAG) interface is down on a switch.

- This message shall be used to indicate that the multi-chassis link aggregation group (MLAG) interface is down on a switch.

Version Added: 1.0.0

Severity: Warning

Resolution: Check physical connectivity and that the MLAG system ID matches on switch pairs.

Message and Arguments: "The MLAG interface '<Arg1>' on switch '<Arg2>' is down."

1. *string*: The MLAG domain ID.
 - This argument shall contain the MLAG domain ID that has been dropped.
2. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch for which an MLAG interface is down.

2.3.5 MLAGInterfacesUp

Indicates that all multi-chassis link aggregation group (MLAG) interfaces are up.

- This message shall be used to indicate all multi-chassis link aggregation group (MLAG) interfaces have been established for one MLAG domain ID.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "All MLAG interfaces have been established for MLAG ID '<Arg1>'."

1. *string*: The MLAG domain ID.
 - This argument shall contain the MLAG domain ID that has been established.

2.3.6 MLAGPeerDown

Indicates that the multi-chassis link aggregation group (MLAG) peer is down.

- This message shall be used to indicate that a multi-chassis link aggregation group (MLAG) peer is down.

Version Added: 1.0.0

Severity: Warning

Resolution: Check physical connectivity and that the port channel ID matches on switch pairs.

Message and Arguments: "MLAG peer switch '<Arg1>' with MLAG ID '<Arg2>' is down."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the peer switch for which MLAG interface has been dropped.
2. *string*: The MLAG domain ID.
 - This argument shall contain the value of the MLAG domain ID that has been dropped.

2.3.7 MLAGPeerUp

Indicates that the multi-chassis link aggregation group (MLAG) peer is up.

- This message shall be used to indicate that a multi-chassis link aggregation group (MLAG) peer is up.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "MLAG peer switch '<Arg1>' with MLAG ID '<Arg2>' is up."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the peer switch for which MLAG

interfaces are up.

2. *string*: The MLAG domain ID.
 - This argument shall contain the value of the MLAG domain ID that has been established.

2.3.8 RoutingFailureThresholdExceeded

Indicates that a switch has encountered an unusually large number of routing errors.

- This message shall be used to indicate that a switch has exceeded a rate threshold for routing errors. This message shall not be emitted more than once in any 24-hour period of time.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' has encountered <Arg2> routing errors in the last <Arg3> minutes."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch that encountered routing errors.
2. *number*: The number of routing errors encountered.
 - This argument shall contain the number of routing errors encountered.
3. *number*: The number of minutes over which routing errors have been encountered.
 - This argument shall contain the number of minutes over which routing errors have been encountered.

2.4 Fabric 1.0.0

This registry defines messages for generic fabrics

Message	Severity	Description
AddressPoolCreated	OK	Indicates that an address pool has been created.
AddressPoolModified	OK	Indicates that an address pool has been modified.
AddressPoolRemoved	OK	Indicates that an address pool has been removed.
CableFailed	Warning	Indicates that a cable has failed.
CableInserted	OK	Indicates that a cable has been inserted into a switch's port.

Message	Severity	Description
CableOK	OK	Indicates that a cable has returned to working condition.
CableRemoved	Warning	Indicates that a cable has been removed from a switch's port.
ConnectionCreated	OK	Indicates that a connection has been created.
ConnectionModified	OK	Indicates that a connection has been modified.
ConnectionRemoved	OK	Indicates that a connection has been removed.
DegradedDownstreamLinkEstablished	Warning	Indicates that a switch's downstream connection is established, but in a degraded state.
DegradedInterswitchLinkEstablished	Warning	Indicates that a switch's interswitch connection is established, but in a degraded state.
DegradedUpstreamLinkEstablished	Warning	Indicates that a switch's upstream connection is established, but in a degraded state.
DownstreamLinkDropped	Warning	Indicates that a switch's downstream connection has gone down.
DownstreamLinkEstablished	OK	Indicates that a switch's downstream connection is established.
DownstreamLinkFlapDetected	Warning	Indicates that a switch's downstream connection is highly unstable.
EndpointCreated	OK	Indicates that an endpoint has been created or discovered.
EndpointModified	OK	Indicates that an endpoint has been modified.
EndpointRemoved	OK	Indicates that an endpoint has been removed.
InterswitchLinkDropped	Warning	Indicates that a switch's interswitch connection has gone down.
InterswitchLinkEstablished	OK	Indicates that a switch's interswitch connection is established.
InterswitchLinkFlapDetected	Warning	Indicates that a switch's interswitch connection is highly unstable.
MaxFrameSizeExceeded	Warning	Indicates that the maximum transmission unit (MTU) for the link has been exceeded.
MediaControllerAdded	OK	Indicates that a media controller has been added.
MediaControllerModified	OK	Indicates that a media controller has been modified.
MediaControllerRemoved	OK	Indicates that a media controller has been removed.
PortAutomaticallyDisabled	OK	Indicates that a switch's port has been automatically disabled.
PortAutomaticallyEnabled	OK	Indicates that a switch's port has been automatically enabled.
PortDegraded	Warning	Indicates that a switch's port is in a degraded state.
PortFailed	Critical	Indicates that a switch's port has become inoperative.
PortManuallyDisabled	OK	Indicates that a switch's port has been manually disabled.

Message	Severity	Description
PortManuallyEnabled	OK	Indicates that a switch's port has been manually enabled.
PortOK	OK	Indicates that a switch's port has returned to a functional state.
SwitchDegraded	Warning	Indicates that a switch is in a degraded state.
SwitchFailed	Critical	Indicates that a switch has become inoperative.
SwitchOK	OK	Indicates that a switch has returned to a functional state.
UpstreamLinkDropped	Warning	Indicates that a switch's upstream connection has gone down.
UpstreamLinkEstablished	OK	Indicates that a switch's upstream connection is established.
UpstreamLinkFlapDetected	Warning	Indicates that a switch's upstream connection is highly unstable.
ZoneCreated	OK	Indicates that a zone has been created.
ZoneModified	OK	Indicates that a zone has been modified.
ZoneRemoved	OK	Indicates that a zone has been removed.

2.4.1 AddressPoolCreated

Indicates that an address pool has been created.

- This message shall be used to indicate that a resource of type `AddressPool1` has been created.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Address pool '<Arg1>' has been created in fabric '<Arg2>'."

1. *string*: The `Id` of the address pool.
 - This argument shall contain the value of the `Id` property of the address pool that was created.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the address pool was created.

2.4.2 AddressPoolModified

Indicates that an address pool has been modified.

- This message shall be used to indicate that a resource of type `AddressPool` has been modified.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Address pool '<Arg1>' in fabric '<Arg2>' has been modified."

1. *string*: The `Id` of the address pool.
 - This argument shall contain the value of the `Id` property of the address pool that was modified.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the address pool was modified.

2.4.3 AddressPoolRemoved

Indicates that an address pool has been removed.

- This message shall be used to indicate that a resource of type `AddressPool` has been removed.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Address pool '<Arg1>' has been removed from fabric '<Arg2>'."

1. *string*: The `Id` of the address pool.
 - This argument shall contain the value of the `Id` property of the address pool that was removed.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the address pool was removed.

2.4.4 CableFailed

Indicates that a cable has failed.

- This message shall be used to indicate that a cable has been detected as failed. This message shall not be emitted more than once in any 24-hour period of time.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "The cable in switch '<Arg1>' port '<Arg2>' has failed."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the failed cable.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port with the failed cable.

2.4.5 CableInserted

Indicates that a cable has been inserted into a switch's port.

- This message shall be used to indicate that a cable has been inserted into a port on a switch.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "A cable has been inserted into switch '<Arg1>' port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the port whose cable was inserted.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port whose cable was inserted.

2.4.6 CableOK

Indicates that a cable has returned to working condition.

- This message shall be used to indicate that a cable has been restored to working condition.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "The cable in switch '<Arg1>' port '<Arg2>' has returned to working condition."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the cable that returned to a functional state.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port with the cable that returned to a functional state.

2.4.7 CableRemoved

Indicates that a cable has been removed from a switch's port.

- This message shall be used to indicate that a cable has been removed from a port on a switch.

Version Added: 1.0.0

Severity: Warning

Resolution: No action is needed.

Message and Arguments: "A cable has been removed from switch '<Arg1>' port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the port whose cable was removed.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port whose cable was removed.

2.4.8 ConnectionCreated

Indicates that a connection has been created.

- This message shall be used to indicate that a resource of type `Connection` has been created.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Connection '<Arg1>' has been created in fabric '<Arg2>'."

1. *string*: The `Id` of the connection.
 - This argument shall contain the value of the `Id` property of the connection that was created.
2. *string*: The `Id` of the fabric.

- This argument shall contain the value of the `Id` property of the fabric in which the connection was created.

2.4.9 ConnectionModified

Indicates that a connection has been modified.

- This message shall be used to indicate that a resource of type `Connection` has been modified.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Connection '<Arg1>' in fabric '<Arg2>' has been modified."

1. *string*: The `Id` of the connection.
 - This argument shall contain the value of the `Id` property of the connection that was modified.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the connection was modified.

2.4.10 ConnectionRemoved

Indicates that a connection has been removed.

- This message shall be used to indicate that a resource of type `Connection` has been removed.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Connection '<Arg1>' has been removed from fabric '<Arg2>'."

1. *string*: The `Id` of the connection.
 - This argument shall contain the value of the `Id` property of the connection that was removed.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the connection was removed.

2.4.11 DegradedDownstreamLinkEstablished

Indicates that a switch's downstream connection is established, but in a degraded state.

- This message shall be used to indicate that a switch's downstream connection is established, but that the link state is degraded.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' downstream link is established on port '<Arg2>', but is running in a degraded state."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the downstream port.

2.4.12 DegradedInterswitchLinkEstablished

Indicates that a switch's interswitch connection is established, but in a degraded state.

- This message shall be used to indicate that a switch's interswitch connection is established, but that the link state is degraded.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' interswitch link is established on port '<Arg2>', but is running in a degraded state."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the interswitch port.

2.4.13 DegradedUpstreamLinkEstablished

Indicates that a switch's upstream connection is established, but in a degraded state.

- This message shall be used to indicate that a switch's upstream connection is established, but that the link state is degraded.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' upstream link is established on port '<Arg2>', but is running in a degraded state."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the upstream port.

2.4.14 DownstreamLinkDropped

Indicates that a switch's downstream connection has gone down.

- This message shall be used to indicate that a switch's downstream connection has gone down.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' downstream link has gone down on port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the downstream port.

2.4.15 DownstreamLinkEstablished

Indicates that a switch's downstream connection is established.

- This message shall be used to indicate that a switch's downstream connection is established.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "Switch '<Arg1>' downstream link is established on port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the downstream port.

2.4.16 DownstreamLinkFlapDetected

Indicates that a switch's downstream connection is highly unstable.

- This message shall be used to indicate that a switch's downstream connection has repeatedly been established and dropped. This message shall not be repeated more than once in any 24-hour period.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' downstream link on port '<Arg2>' has been established and dropped <Arg3> times in the last <Arg4> minutes."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the downstream port that is flapping.
3. *number*: The number of times the link has flapped.
 - This argument shall contain the count of uplink establishment/disconnection cycles.
4. *number*: The number of minutes over which the link has flapped.
 - This argument shall contain the number of minutes over which link flapping activity has been detected.

2.4.17 EndpointCreated

Indicates that an endpoint has been created or discovered.

- This message shall be used to indicate that a resource of type `Endpoint` has been created or discovered.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Endpoint '<Arg1>' has been created in fabric '<Arg2>'."

1. *string*: The `Id` of the endpoint.
 - This argument shall contain the value of the `Id` property of the endpoint that was created.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the endpoint was created.

2.4.18 EndpointModified

Indicates that an endpoint has been modified.

- This message shall be used to indicate that a resource of type `Endpoint` has been modified.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Endpoint '<Arg1>' in fabric '<Arg2>' has been modified."

1. *string*: The `Id` of the endpoint.
 - This argument shall contain the value of the `Id` property of the endpoint that was modified.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the endpoint was modified.

2.4.19 EndpointRemoved

Indicates that an endpoint has been removed.

- This message shall be used to indicate that a resource of type `Endpoint` has been removed.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Endpoint '<Arg1>' has been removed from fabric '<Arg2>'."

1. *string*: The `Id` of the endpoint.
 - This argument shall contain the value of the `Id` property of the endpoint that was removed.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the endpoint was removed.

2.4.20 InterswitchLinkDropped

Indicates that a switch's interswitch connection has gone down.

- This message shall be used to indicate that a switch's interswitch connection has gone down.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' interswitch link has gone down on port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the interswitch port.

2.4.21 InterswitchLinkEstablished

Indicates that a switch's interswitch connection is established.

- This message shall be used to indicate that a switch's interswitch connection is established.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "Switch '<Arg1>' interswitch link is established on port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the interswitch port.

2.4.22 InterswitchLinkFlapDetected

Indicates that a switch's interswitch connection is highly unstable.

- This message shall be used to indicate that a switch's interswitch connection has repeatedly been established and dropped. This message shall not be repeated more than once in any 24-hour period.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' interswitch link on port '<Arg2>' has been established and dropped <Arg3> times in the last <Arg4> minutes."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the interswitch port that is flapping.
3. *number*: The number of times the link has flapped.
 - This argument shall contain the count of uplink establishment/disconnection cycles.
4. *number*: The number of minutes over which the link has flapped.
 - This argument shall contain the number of minutes over which link flapping activity has been detected.

2.4.23 MaxFrameSizeExceeded

Indicates that the maximum transmission unit (MTU) for the link has been exceeded.

- This message shall be used to indicate that packets have been dropped due to the maximum transmission unit (MTU) frame size being exceeded. This message shall not be emitted more than once in any 24-hour period of time.

Version Added: 1.0.0

Severity: Warning

Resolution: Ensure that path MTU discovery is enabled and functioning correctly.

Message and Arguments: "MTU size on switch '<Arg1>' port '<Arg2>' is set to <Arg3>. One or more packets with a larger size have been dropped."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch in which one or more packets have been dropped.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port in which one or more oversized packets were received.
3. *number*: The MTU size.
 - This argument shall contain the MTU size.

2.4.24 MediaControllerAdded

Indicates that a media controller has been added.

- This message shall be used to indicate that a resource of type `MediaController` has been added.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Media controller '<Arg1>' has been added to chassis '<Arg2>'."

1. *string*: The `Id` of the media controller.
 - This argument shall contain the value of the `Id` property of the media controller that was added.
2. *string*: The `Id` of the chassis.
 - This argument shall contain the value of the `Id` property of the chassis in which the media controller was added.

2.4.25 MediaControllerModified

Indicates that a media controller has been modified.

- This message shall be used to indicate that a resource of type `MediaController` has been modified.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Media controller '<Arg1>' in chassis '<Arg2>' has been modified."

1. *string*: The `Id` of the media controller.
 - This argument shall contain the value of the `Id` property of the media controller that was modified.
2. *string*: The `Id` of the chassis.
 - This argument shall contain the value of the `Id` property of the chassis in which the media controller was modified.

2.4.26 MediaControllerRemoved

Indicates that a media controller has been removed.

- This message shall be used to indicate that a resource of type `MediaController` has been removed.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Media controller '<Arg1>' has been removed from chassis '<Arg2>'."

1. *string*: The `Id` of the media controller.
 - This argument shall contain the value of the `Id` property of the media controller that was removed.
2. *string*: The `Id` of the chassis.
 - This argument shall contain the value of the `Id` property of the chassis in which the media controller was removed.

2.4.27 PortAutomaticallyDisabled

Indicates that a switch's port has been automatically disabled.

- This message shall be used to indicate that a switch's port has been automatically disabled, such as by the service.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "Switch '<Arg1>' port '<Arg2>' has been automatically disabled."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the disabled port.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port that has been disabled.

2.4.28 PortAutomaticallyEnabled

Indicates that a switch's port has been automatically enabled.

- This message shall be used to indicate that a switch's port has been automatically enabled, such as by the service.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "Switch '<Arg1>' port '<Arg2>' has been automatically enabled."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the enabled port.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port that has been enabled.

2.4.29 PortDegraded

Indicates that a switch's port is in a degraded state.

- This message shall be used to indicate that a port has entered a degraded state.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' port '<Arg2>' is in a degraded state."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the degraded port.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port that has entered a degraded state.

2.4.30 PortFailed

Indicates that a switch's port has become inoperative.

- This message shall be used to indicate that a port has become inoperative. This message shall not be issued when a port is manually disabled.

Version Added: 1.0.0

Severity: Critical

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' port '<Arg2>' has failed and is inoperative."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the failed port.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port that has failed.

2.4.31 PortManuallyDisabled

Indicates that a switch's port has been manually disabled.

- This message shall be used to indicate that a switch's port has been manually disabled, such as by a network administrator.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "Switch '<Arg1>' port '<Arg2>' has been manually disabled."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the disabled port.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port that has been disabled.

2.4.32 PortManuallyEnabled

Indicates that a switch's port has been manually enabled.

- This message shall be used to indicate that a switch's port has been manually enabled, such as by a network administrator.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "Switch '<Arg1>' port '<Arg2>' has been manually enabled."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the enabled port.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port that has been enabled.

2.4.33 PortOK

Indicates that a switch's port has returned to a functional state.

- This message shall be used to indicate that a port has returned to a functional state.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Switch '<Arg1>' port '<Arg2>' has returned to a functional state."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch with the functional port.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the port that has returned to a functional state.

2.4.34 SwitchDegraded

Indicates that a switch is in a degraded state.

- This message shall be used to indicate that a switch has entered a degraded state.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' is in a degraded state."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch that has entered a degraded state.

2.4.35 SwitchFailed

Indicates that a switch has become inoperative.

- This message shall be used to indicate that a switch has become inoperative.

Version Added: 1.0.0

Severity: Critical

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' has failed and is inoperative."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch that has failed.

2.4.36 SwitchOK

Indicates that a switch has returned to a functional state.

- This message shall be used to indicate that a switch has returned to a functional state.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Switch '<Arg1>' has returned to a functional state."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch that has returned to a functional state.

2.4.37 UpstreamLinkDropped

Indicates that a switch's upstream connection has gone down.

- This message shall be used to indicate that a switch's upstream connection has gone down.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' upstream link has gone down on port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the upstream port.

2.4.38 UpstreamLinkEstablished

Indicates that a switch's upstream connection is established.

- This message shall be used to indicate that a switch's upstream connection is established.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "Switch '<Arg1>' upstream link is established on port '<Arg2>'."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the upstream port.

2.4.39 UpstreamLinkFlapDetected

Indicates that a switch's upstream connection is highly unstable.

- This message shall be used to indicate that a switch's upstream connection has repeatedly been established

and dropped. This message shall not be repeated more than once in any 24-hour period.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '<Arg1>' upstream link on port '<Arg2>' has been established and dropped <Arg3> times in the last <Arg4> minutes."

1. *string*: The `Id` of the switch.
 - This argument shall contain the value of the `Id` property of the switch.
2. *string*: The `Id` of the port.
 - This argument shall contain the value of the `Id` property of the upstream port that is flapping.
3. *number*: The number of times the link has flapped.
 - This argument shall contain the count of uplink establishment/disconnection cycles.
4. *number*: The number of minutes over which the link has flapped.
 - This argument shall contain the number of minutes over which link flapping activity has been detected.

2.4.40 ZoneCreated

Indicates that a zone has been created.

- This message shall be used to indicate that a resource of type `Zone` has been created.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Zone '<Arg1>' has been created in fabric '<Arg2>'."

1. *string*: The `Id` of the zone.
 - This argument shall contain the value of the `Id` property of the zone that was created.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the zone was created.

2.4.41 ZoneModified

Indicates that a zone has been modified.

- This message shall be used to indicate that a resource of type `Zone` has been modified.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Zone '<Arg1>' in fabric '<Arg2>' has been modified."

1. *string*: The `Id` of the zone.
 - This argument shall contain the value of the `Id` property of the zone that was modified.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the zone was modified.

2.4.42 ZoneRemoved

Indicates that a zone has been removed.

- This message shall be used to indicate that a resource of type `Zone` has been removed.

Version Added: 1.0.0

Severity: OK

Resolution: No action is needed.

Message and Arguments: "Zone '<Arg1>' has been removed from fabric '<Arg2>'."

1. *string*: The `Id` of the zone.
 - This argument shall contain the value of the `Id` property of the zone that was removed.
2. *string*: The `Id` of the fabric.
 - This argument shall contain the value of the `Id` property of the fabric in which the zone was removed.

2.5 HeartbeatEvent 1.0.0

This registry defines the messages to use for periodic heartbeat, also known as 'keep alive', events.

Message	Severity	Description
RedfishServiceFunctional	OK	An event sent periodically upon request to indicates that the Redfish service is functional.

2.5.1 RedfishServiceFunctional

An event sent periodically upon request to indicates that the Redfish service is functional.

- This message shall be used to indicate that the Redfish service is functional. This message shall only be sent if specifically requested by an event destination during the creation of a subscription using the `SendHeartbeatMinutes` property.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "Redfish service is functional."

2.6 JobEvent 1.0.0

This registry defines the messages for job related events.

Message	Severity	Description
JobCancelled	Warning	A job has been cancelled.
JobCompletedException	Warning	A job has completed with warnings or errors.
JobCompletedOK	OK	A job has completed.
JobProgressChanged	OK	A job has changed progress.
JobRemoved	Warning	A job has been removed.
JobResumed	OK	A job has resumed.
JobStarted	OK	A job has started.
JobSuspended	Warning	A job has been suspended.

2.6.1 JobCancelled

A job has been cancelled.

- This message shall be used to indicate that a job has been cancelled and the `JobState` property transitioned to `Cancelled`.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '<Arg1>' has been cancelled."

1. *string*: The `Id` of the job.
 - This argument shall contain the value of the `Id` property of the job resource that has been cancelled.

2.6.2 JobCompletedException

A job has completed with warnings or errors.

- This message shall be used to indicate that a job has completed and the `JobState` property transitioned to `Exception`.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '<Arg1>' has completed with warnings or errors."

1. *string*: The `Id` of the job.
 - This argument shall contain the value of the `Id` property of the job resource that has completed with warnings or errors.

2.6.3 JobCompletedOK

A job has completed.

- This message shall be used to indicate that a job has completed and the `JobState` property transitioned to `Completed`.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '<Arg1>' has completed."

1. *string*: The `Id` of the job.
 - This argument shall contain the value of the `Id` property of the job resource that has completed.

2.6.4 JobProgressChanged

A job has changed progress.

- This message shall be used to indicate that a job has changed progress.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '<Arg1>' has changed to progress <Arg2> percent complete."

1. *string*: The `Id` of the job.
 - This argument shall contain the value of the `Id` property of the job resource that has changed progress.
2. *number*: The percent completion of the job.
 - This argument shall contain the percent completion of the job that has changed progress and shall be a value from 0 to 100.

2.6.5 JobRemoved

A job has been removed.

- This message shall be used to indicate that a job has been removed.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '<Arg1>' has been removed."

1. *string*: The `Id` of the job.
 - This argument shall contain the value of the `Id` property of the job resource that has been removed.

2.6.6 JobResumed

A job has resumed.

- This message shall be used to indicate that a job has resumed and the `JobState` property transitioned to `Running` from `Suspended`, `Interrupted`, or `UserIntervention`.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '<Arg1>' has resumed."

1. *string*: The `Id` of the job.
 - This argument shall contain the value of the `Id` property of the job resource that has resumed.

2.6.7 JobStarted

A job has started.

- This message shall be used to indicate that a job has started as the `JobState` property transitions to `Starting` or `Running`.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '<Arg1>' has started."

1. *string*: The `Id` of the job that started.
 - This argument shall contain the value of the `Id` property of the job resource that has started.

2.6.8 JobSuspended

A job has been suspended.

- This message shall be used to indicate that a job has been suspended and the `JobState` property transitioned to `Suspended`, `Interrupted`, or `UserIntervention`.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '<Arg1>' has been susepended."

1. *string*: The `Id` of the job.
 - This argument shall contain the value of the `Id` property of the job resource that has been susepended.

2.7 License 1.0.1

This registry defines the license status and error messages.

Message	Severity	Description
DaysBeforeExpiration	OK	Indicates the number of days remaining on a license before expiration.
Expired	Warning	Indicates that a license has expired and its functionality has been disabled.
GracePeriod	Warning	Indicates that a license has expired and entered its grace period.
InstallFailed	Critical	Indicates that the service failed to install the license.
InvalidLicense	Critical	Indicates that the license was not recognized, is corrupted, or is invalid.
LicenseInstalled	OK	Indicates that a license has been installed.
NotApplicableToTarget	Critical	Indicates that the license is not applicable to the target.
TargetsRequired	Critical	Indicates that one or more targets need to be specified with the license.

2.7.1 DaysBeforeExpiration

Indicates the number of days remaining on a license before expiration.

- This message shall be used to indicate the number of days remaining on a license before expiration.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The license '<Arg1>' will expire in <Arg2> days."

1. *string*: The name of the license.
 - This argument shall contain the human readable identifier of the license.
2. *number*: The number of days before the license expires.
 - This argument shall contain the number of days before the license expires.

2.7.2 Expired

Indicates that a license has expired and its functionality has been disabled.

- This message shall be used to indicate that a license has expired and its functionality has been disabled.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The license '<Arg1>' has expired."

1. *string*: The name of the license.
 - This argument shall contain the human readable identifier of the license.

2.7.3 GracePeriod

Indicates that a license has expired and entered its grace period.

- This message shall be used to indicate that a license has expired and entered its grace period.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The license '<Arg1>' has expired, <Arg2> day grace period before licensed functionality is disabled."

1. *string*: The name of the license.
 - This argument shall contain the human readable identifier of the license.
2. *number*: The number day of grace period.
 - This argument shall contain the number day of grace period for the licensed functionality.

2.7.4 InstallFailed

Indicates that the service failed to install the license.

- This message shall be used to indicate that the service failed to install the license.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "Failed to install the license. Reason: <Arg1>."

1. *String*: The reason for the license installation failure.
 - This argument shall contain the reason for the license installation failure.

2.7.5 InvalidLicense

Indicates that the license was not recognized, is corrupted, or is invalid.

- This message shall be used to indicate that the content of the license was not recognized, is corrupted, or is invalid.

Version Added: 1.0.0

Severity: Critical

Resolution: Verify the license content is correct and resubmit the request.

Message and Arguments: "The content of the license was not recognized, is corrupted, or is invalid."

2.7.6 LicenseInstalled

Indicates that a license has been installed.

- This message shall be used to indicate that a license has been installed. Existing licenses may have been updated or otherwise affected by the installation.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The license '<Arg1>' has been installed."

1. *String*: The name of the license.
 - This argument shall contain the human readable identifier of the license.

2.7.7 NotApplicableToTarget

Indicates that the license is not applicable to the target.

- This message shall be used to indicate that the license is not applicable to the target.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the license compatibility or applicability to the specified target.

Message and Arguments: "The license is not applicable to the target."

2.7.8 TargetsRequired

Indicates that one or more targets need to be specified with the license.

- This message shall be used to indicate that one or more targets need to be specified with the license.

Version Added: 1.0.0

Severity: Critical

Resolution: Add `AuthorizedDevices` to `Links` and resubmit the request.

Message and Arguments: "The license requires targets to be specified."

2.8 LogService 1.0.0

This registry defines the messages for log service related events

Message	Severity	Description
DiagnosticDataCollected	OK	Indicates that diagnostic data has been collected due to a client invoking the <code>CollectDiagnosticData</code> action.

2.8.1 DiagnosticDataCollected

Indicates that diagnostic data has been collected due to a client invoking the `CollectDiagnosticData` action.

- This message shall be used to indicate that diagnostic data has been collected due to a client invoking the `CollectDiagnosticData` action.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "<Arg1>' diagnostic data collected."

1. *string*: The value of the `DiagnosticDataType` parameter.
 - The value of this argument shall be a string containing the `DiagnosticDataType` parameter provided by the client when the `CollectDiagnosticData` action was invoked.

2.9 NetworkDevice 1.0.1

This registry defines the messages for networking devices

Message	Severity	Description
CableInserted	OK	Indicates that a network cable has been inserted.
CableRemoved	OK	Indicates that a network cable has been removed.
ConnectionDropped	OK	Indicates that a network connection has been dropped.
ConnectionEstablished	OK	Indicates that a network connection has been established.
DegradedConnectionEstablished	Warning	Indicates that a network connection has been established, but at an unexpectedly low link speed.
LinkFlapDetected	Warning	Indicates that a network connection is highly unstable.

2.9.1 CableInserted

Indicates that a network cable has been inserted.

- This message shall be used to indicate that a network cable has been inserted. This message shall not be sent for a backplane connection.

Version Added: 1.0.0

Severity: OK

Resolution: Refresh your cached version of the network port to get the updated information from the service.

Message and Arguments: "A network cable has been inserted into network adapter '<Arg1>' port '<Arg2>'."

1. *string*: The `Id` of the network adapter.
 - The value of this argument shall be a string containing the value of the `Id` property of network adapter into which a cable has been plugged.
2. *string*: The `Id` of the network port.
 - The value of this argument shall be a string containing the value of the `Id` property of network port into which a cable has been plugged.

2.9.2 CableRemoved

Indicates that a network cable has been removed.

- This message shall be used to indicate that a network cable has been removed. This message shall not be sent for a backplane connection.

Version Added: 1.0.0

Severity: OK

Resolution: Refresh your cached version of the network port to get the updated information from the service.

Message and Arguments: "A cable has been removed from network adapter '<Arg1>' port '<Arg2>'."

1. *string*: The `Id` of the network adapter.
 - The value of this argument shall be a string containing the value of the `Id` property of network adapter from which a cable has been unplugged.
2. *string*: The `Id` of the network port.
 - The value of this argument shall be a string containing the value of the `Id` property of network port from which a cable has been unplugged.

2.9.3 ConnectionDropped

Indicates that a network connection has been dropped.

- This message shall be used to indicate that the link for a network connection is no longer intact.

Version Added: 1.0.0

Severity: OK

Resolution: Refresh your cached version of the network port to get the updated information from the service.

Message and Arguments: "The connection is no longer active for network adapter '<Arg1>' port '<Arg2>' function '<Arg3>'."

1. *string*: The `Id` of the network adapter.
 - The value of this argument shall be a string containing the value of the `Id` property of network adapter from which a connection has dropped.
2. *string*: The `Id` of the network port.
 - The value of this argument shall be a string containing the value of the `Id` property of network port from which a connection has dropped.
3. *string*: The `Id` of the network function.
 - The value of this argument shall be a string containing the value of the `Id` property of network device function from which a connection has dropped.

2.9.4 ConnectionEstablished

Indicates that a network connection has been established.

- This message shall be used to indicate that the link for a network connection is newly established.

Version Added: 1.0.0

Severity: OK

Resolution: Refresh your cached version of the network port to get the updated information from the service.

Message and Arguments: "A network connection has been established for network adapter '<Arg1>' port '<Arg2>' function '<Arg3>'."

1. *string*: The `Id` of the network adapter.
 - The value of this argument shall be a string containing the value of the `Id` property of network adapter to which a connection has been established.
2. *string*: The `Id` of the network port.
 - The value of this argument shall be a string containing the value of the `Id` property of network port to which a connection has been established.
3. *string*: The `Id` of the network function.
 - The value of this argument shall be a string containing the value of the `Id` property of network device function to which a connection has been established.

2.9.5 DegradedConnectionEstablished

Indicates that a network connection has been established, but at an unexpectedly low link speed.

- This message shall be used to indicate that a network connection is newly established, but that the link speed is lower than provisioned for the channel.

Version Added: 1.0.0

Severity: Warning

Resolution: Refresh your cached version of the network port to get the updated information from the service.

Message and Arguments: "A degraded network connection has been established for network adapter '<Arg1>' port '<Arg2>' function '<Arg3>'."

1. *string*: The `Id` of the network adapter.
 - The value of this argument shall be a string containing the value of the `Id` property of network adapter to which a connection has been established.
2. *string*: The `Id` of the network port.
 - The value of this argument shall be a string containing the value of the `Id` property of network port to which a connection has been established.
3. *string*: The `Id` of the network function.
 - The value of this argument shall be a string containing the value of the `Id` property of network device function to which a connection has been established.

2.9.6 LinkFlapDetected

Indicates that a network connection is highly unstable.

- This message shall be used to indicate that a network connection has repeatedly been established and dropped. This message shall not be repeated more than once in any 24 hour period.

Version Added: 1.0.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "The network connection for network adapter '<Arg1>' port '<Arg2>' function '<Arg3>' has been established and dropped '<Arg4>' times in the last '<Arg5>' minutes."

1. *string*: The `Id` of the network adapter.

- The value of this argument shall be a string containing the value of the `Id` property of network adapter to which a connection has been established.
- 2. *string*: The `Id` of the network port.
 - The value of this argument shall be a string containing the value of the `Id` property of network port to which a connection has been established.
- 3. *string*: The `Id` of the network function.
 - The value of this argument shall be a string containing the value of the `Id` property of network device function to which a connection has been established.
- 4. *number*: The number of times the link has flapped.
 - The value of this argument shall be a number representing the count of link establishment/disconnection cycles.
- 5. *number*: The number of minutes over which the link has flapped.
 - The value of this argument shall be a number representing the number of minutes over which link flapping activity has been detected.

2.10 ResourceEvent 1.2.1

This registry defines the messages to use for resource events.

Message	Severity	Description
AggregationSourceDiscovered	OK	Indicates that a new aggregation source has been discovered.
LicenseAdded	OK	Indicates that a license has been added.
LicenseChanged	Warning	Indicates that a license has changed.
LicenseExpired	Warning	Indicates that a license has expired.
ResourceChanged	OK	Indicates that one or more resource properties have changed. This is not used whenever there is another event message for that specific change, such as only the state has changed.
ResourceCreated	OK	Indicates that all conditions of a successful creation operation have been met.
ResourceErrorThresholdCleared	OK	Indicates that a specified resource property has cleared its error threshold. Examples would be drive I/O errors, or network link errors.
ResourceErrorThresholdExceeded	Critical	Indicates that a specified resource property has exceeded its error threshold. Examples would be drive I/O errors, or network link errors.
ResourceErrorsCorrected	OK	Indicates that a specified resource property has corrected errors. Examples would be drive I/O errors, or network link errors.
ResourceErrorsDetected	Warning	Indicates that a specified resource property has detected errors. Examples would be drive I/O errors, or network link errors.

Message	Severity	Description
ResourcePaused	OK	Indicates that the power state of a resource has changed to paused.
ResourcePoweredOff	OK	Indicates that the power state of a resource has changed to powered off.
ResourcePoweredOn	OK	Indicates that the power state of a resource has changed to powered on.
ResourcePoweringOff	OK	Indicates that the power state of a resource has changed to powering off.
ResourcePoweringOn	OK	Indicates that the power state of a resource has changed to powering on.
ResourceRemoved	OK	Indicates that all conditions of a successful remove operation have been met.
ResourceSelfTestCompleted	OK	Indicates that a self-test has completed.
ResourceSelfTestFailed	Critical	Indicates that a self-test has failed. Suggested resolution may be provided as OEM data.
ResourceStateChanged	OK	Indicates that the state of a resource has changed.
ResourceStatusChangedCritical	Critical	Indicates that the health of a resource has changed to Critical.
ResourceStatusChangedOK	OK	Indicates that the health of a resource has changed to OK.
ResourceStatusChangedWarning	Warning	Indicates that the health of a resource has changed to Warning.
ResourceVersionIncompatible	Warning	Indicates that an incompatible version of software has been detected. Examples may be after a component or system level software update.
ResourceWarningThresholdCleared	OK	Indicates that a specified resource property has cleared its warning threshold. Examples would be drive I/O errors, or network link errors. Suggested resolution may be provided as OEM data.
ResourceWarningThresholdExceeded	Warning	Indicates that a specified resource property has exceeded its warning threshold. Examples would be drive I/O errors, or network link errors. Suggested resolution may be provided as OEM data.
URIForResourceChanged	OK	Indicates that the URI for a resource has changed. Examples for this would be physical component replacement or redistribution.

2.10.1 AggregationSourceDiscovered

Indicates that a new aggregation source has been discovered.

- This message shall be used to indicate an aggregation source has been discovered in the network or host. The `OriginOfCondition` property shall contain a link to the matching `ConnectionMethod` resource if known.

Version Added: 1.2.0

Severity: OK

Resolution: The aggregation source is available to the service and can be identified using the identified connection method.

Message and Arguments: "A aggregation source of connection method `\<Arg1\>` located at `\<Arg2\>` has been discovered."

1. *string*: The connection method of the discovered aggregation source.
 - This argument shall contain a value from the `ConnectionMethodType` enumeration defined in the `ConnectionMethod` schema that describes connection method of the discovered aggregation source.
2. *string*: The location of the discovered aggregation source, such as an IP address or service entry point.
 - This argument shall contain the location of the discovered aggregation source, such as an IP address or service entry point.

2.10.2 LicenseAdded

Indicates that a license has been added.

- This message shall be used to indicate that a license has been added.

Version Added: 1.0.0

Severity: OK

Resolution: See vendor specific instructions for specific actions.

Message and Arguments: "A license for '<Arg1>' has been added. The following message was returned: '<Arg2>!'."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant resource or service affected by the software license.
2. *string*: The message returned from the license validation process.
 - This argument shall contain the message returned from the license validation process or software.

2.10.3 LicenseChanged

Indicates that a license has changed.

- This message shall be used to indicate that a license has changed.

Version Added: 1.0.0

Severity: Warning

Resolution: See vendor specific instructions for specific actions.

Message and Arguments: "A license for '<Arg1>' has changed. The following message was returned: '<Arg2>'."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant resource or service affected by the software license.
2. *string*: The message returned from the license validation process.
 - This argument shall contain the message returned from the license validation process or software.

2.10.4 LicenseExpired

Indicates that a license has expired.

- This message shall be used to indicate that a license has expired.

Version Added: 1.0.0

Severity: Warning

Resolution: See vendor specific instructions for specific actions.

Message and Arguments: "A license for '<Arg1>' has expired. The following message was returned: '<Arg2>'."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant resource or service affected by the software license.
2. *string*: The message returned from the license validation process.
 - This argument shall contain the message returned from the license validation process or software.

2.10.5 ResourceChanged

Indicates that one or more resource properties have changed. This is not used whenever there is another event message for that specific change, such as only the state has changed.

- This message shall be used to indicate that one or more resource properties have changed.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "One or more resource properties have changed."

2.10.6 ResourceCreated

Indicates that all conditions of a successful creation operation have been met.

- This message shall be used to indicate that all conditions of a successful creation operation have been met.

Version Added: 1.0.0

Severity: OK

Resolution: None

Message and Arguments: "The resource has been created successfully."

2.10.7 ResourceErrorThresholdCleared

Indicates that a specified resource property has cleared its error threshold. Examples would be drive I/O errors, or network link errors.

- This message shall be used to indicate that a specified resource property has cleared its error threshold.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource property <Arg1> has cleared the error threshold of value <Arg2>."

1. *string*: The name of the property.
 - This argument shall contain the name of the relevant property from a Redfish resource.
2. *number*: The value of the threshold.
 - This argument shall contain the value of the relevant error threshold.

2.10.8 ResourceErrorThresholdExceeded

Indicates that a specified resource property has exceeded its error threshold. Examples would be drive I/O errors, or network link errors.

- This message shall be used to indicate that a specified resource property has exceeded its error threshold.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "The resource property <Arg1> has exceeded error threshold of value <Arg2>."

1. *string*: The name of the property.
 - This argument shall contain the name of the relevant property from a Redfish resource.
2. *number*: The value of the threshold.
 - This argument shall contain the value of the relevant error threshold.

2.10.9 ResourceErrorsCorrected

Indicates that a specified resource property has corrected errors. Examples would be drive I/O errors, or network link errors.

- This message shall be used to indicate that a specified resource property has corrected errors.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource property <Arg1> has corrected errors of type '<Arg2>'."

1. *string*: The name of the property.
 - This argument shall contain the name of the relevant property from a Redfish resource.
2. *string*: The type of error encountered.
 - This argument shall contain a description of the type of error encountered.

2.10.10 ResourceErrorsDetected

Indicates that a specified resource property has detected errors. Examples would be drive I/O errors, or network link errors.

- This message shall be used to indicate that a specified resource property has detected errors.

Version Added: 1.0.0

Severity: Warning

Resolution: Resolution dependent upon error type.

Message and Arguments: "The resource property <Arg1> has detected errors of type '<Arg2>'."

1. *string*: The name of the property.
 - This argument shall contain the name of the relevant property from a Redfish resource.
2. *string*: The type of error encountered.
 - This argument shall contain a description of the type of error encountered.

2.10.11 ResourcePaused

Indicates that the power state of a resource has changed to paused.

- This message shall be used to indicate that the value of the PowerState property of a resource has changed to `Paused`.

Version Added: 1.1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource \<Arg1\> has been paused."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.

2.10.12 ResourcePoweredOff

Indicates that the power state of a resource has changed to powered off.

- This message shall be used to indicate that the value of the PowerState property of a resource has changed to `Off`.

Version Added: 1.1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource \<Arg1\> has powered off."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.

2.10.13 ResourcePoweredOn

Indicates that the power state of a resource has changed to powered on.

- This message shall be used to indicate that the value of the PowerState property of a resource has changed to `On`.

Version Added: 1.1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource `\<Arg1\>` has powered on."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.

2.10.14 ResourcePoweringOff

Indicates that the power state of a resource has changed to powering off.

- This message shall be used to indicate that the value of the PowerState property of a resource has changed to `PoweringOff`. This message should not be used if the PowerState transition to `off` will occur immediately.

Version Added: 1.1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource `\<Arg1\>` is powering off."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.

2.10.15 ResourcePoweringOn

Indicates that the power state of a resource has changed to powering on.

- This message shall be used to indicate that the value of the PowerState property of a resource has changed to `PoweringOn`. This message should not be used if the PowerState transition to `on` will occur immediately.

Version Added: 1.1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource \<Arg1\> is powering on."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.

2.10.16 ResourceRemoved

Indicates that all conditions of a successful remove operation have been met.

- This message shall be used to indicates that all conditions of a successful remove operation have been met.

Version Added: 1.0.0

Severity: OK

Resolution: None

Message and Arguments: "The resource has been removed successfully."

2.10.17 ResourceSelfTestCompleted

Indicates that a self-test has completed.

- This message shall be used to indicate that a self-test has completed.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "A self-test has completed."

2.10.18 ResourceSelfTestFailed

Indicates that a self-test has failed. Suggested resolution may be provided as OEM data.

- This message shall be used to indicate that a self-test has failed.

Version Added: 1.0.0

Severity: Critical

Resolution: See vendor specific instructions for specific actions.

Message and Arguments: "A self-test has failed. The following message was returned: '<Arg1>'."

1. *string*: The self-test error message.
 - This argument shall contain the error message received as a result from the self-test.

2.10.19 ResourceStateChanged

Indicates that the state of a resource has changed.

- This message shall be used to indicate that the value of the State property of a resource has changed.

Version Added: 1.1.0

Severity: OK

Resolution: None.

Message and Arguments: "The state of resource \<Arg1\> has changed to <Arg2>."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.
2. *string*: The state of the resource.
 - This argument shall contain the value of the `State` property for the relevant Redfish resource. The values shall be used from the definition of the `State` enumeration in the `Resource` schema.

2.10.20 ResourceStatusChangedCritical

Indicates that the health of a resource has changed to Critical.

- This message shall be used to indicate that the health of a resource has changed to Critical.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "The health of resource \<Arg1\> has changed to <Arg2>."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.

2. *string*: The state of the resource.
 - This argument shall contain the value of the `Health` property for the relevant Redfish resource. The values shall be used from the definition of the `Health` enumeration in the `Resource` schema.

2.10.21 ResourceStatusChangedOK

Indicates that the health of a resource has changed to OK.

- This message shall be used to indicate that the health of a resource has changed to OK.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The health of resource '<Arg1>' has changed to <Arg2>."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.
2. *string*: The state of the resource.
 - This argument shall contain the value of the `Health` property for the relevant Redfish resource. The values shall be used from the definition of the `Health` enumeration in the `Resource` schema.

2.10.22 ResourceStatusChangedWarning

Indicates that the health of a resource has changed to Warning.

- This message shall be used to indicate that the health of a resource has changed to Warning.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The health of resource '`\<Arg1\>`' has changed to <Arg2>."

1. *string*: The relevant resource.
 - This argument shall contain the name of the relevant Redfish resource.
2. *string*: The state of the resource.
 - This argument shall contain the value of the `Health` property for the relevant Redfish resource. The values shall be used from the definition of the `Health` enumeration in the `Resource` schema.

2.10.23 ResourceVersionIncompatible

Indicates that an incompatible version of software has been detected. Examples may be after a component or system level software update.

- This message shall be used to indicate that an incompatible version of software has been detected.

Version Added: 1.0.0

Severity: Warning

Resolution: Compare the version of the resource with the compatible version of the software.

Message and Arguments: "An incompatible version of software '<Arg1>' has been detected."

1. *string*: The name of the software component.
 - This argument shall contain the name of the relevant software component or package. This might include both name and version information.

2.10.24 ResourceWarningThresholdCleared

Indicates that a specified resource property has cleared its warning threshold. Examples would be drive I/O errors, or network link errors. Suggested resolution may be provided as OEM data.

- This message shall be used to indicate that a specified resource property has cleared its warning threshold.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource property <Arg1> has cleared the warning threshold of value <Arg2>."

1. *string*: The name of the property.
 - This argument shall contain the name of the relevant property from a Redfish resource.
2. *number*: The value of the threshold.
 - This argument shall contain the value of the relevant error threshold.

2.10.25 ResourceWarningThresholdExceeded

Indicates that a specified resource property has exceeded its warning threshold. Examples would be drive I/O errors, or network link errors. Suggested resolution may be provided as OEM data.

- This message shall be used to indicate that a specified resource property has exceeded its warning threshold.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The resource property <Arg1> has exceeded its warning threshold of value <Arg2>."

1. *string*: The name of the property.
 - This argument shall contain the name of the relevant property from a Redfish resource.
2. *number*: The value of the threshold.
 - This argument shall contain the value of the relevant error threshold.

2.10.26 URIForResourceChanged

Indicates that the URI for a resource has changed. Examples for this would be physical component replacement or redistribution.

- This message shall be used to indicate that the URI for a resource has changed.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The URI for the resource has changed."

2.11 SensorEvent 1.0.0

This registry defines messages used for general events related to Sensor resources.

Message	Severity	Description
InvalidSensorReading	Warning	Indicates that the service received an invalid reading from a sensor.
ReadingAboveLowerCriticalThreshold	Warning	Indicates that a sensor reading is no longer below the lower critical threshold, but is still outside of normal operating range.
ReadingAboveLowerFatalThreshold	Critical	Indicates that a sensor reading is no longer below the lower fatal threshold, but is still outside of normal operating range.

Message	Severity	Description
ReadingAboveUpperCautionThreshold	Warning	Indicates that a sensor reading is above the upper caution threshold.
ReadingAboveUpperCriticalThreshold	Critical	Indicates that a sensor reading is above the upper critical threshold.
ReadingAboveUpperFatalThreshold	Critical	Indicates that a sensor reading is above the upper fatal threshold.
ReadingBelowLowerCautionThreshold	Warning	Indicates that a sensor reading is below the lower caution threshold.
ReadingBelowLowerCriticalThreshold	Critical	Indicates that a sensor reading is below the lower critical threshold.
ReadingBelowLowerFatalThreshold	Critical	Indicates that a sensor reading is below the lower fatal threshold.
ReadingBelowUpperCriticalThreshold	Warning	Indicates that a sensor reading is no longer above the upper critical threshold, but is still outside of normal operating range.
ReadingBelowUpperFatalThreshold	Critical	Indicates that a sensor reading is no longer above the upper fatal threshold, but is still outside of normal operating range.
ReadingCritical	Critical	Indicates that a sensor reading exceeds an internal critical level.
ReadingNoLongerCritical	Warning	Indicates that a sensor reading no longer exceeds an internal critical level, but exceeds an internal warning level.
ReadingWarning	Warning	Indicates that a sensor reading exceeds an internal warning level.
SensorFailure	Warning	Indicates that the service cannot communicate with a sensor or has detected a failure.
SensorReadingNormalRange	OK	Indicates that a sensor reading is now within normal operating range.
SensorRestored	OK	Indicates that a sensor has been repaired or communications have been restored. It may also indicate that the service is receiving valid data from a sensor.

2.11.1 InvalidSensorReading

Indicates that the service received an invalid reading from a sensor.

- This message shall be used to indicate that the service received invalid data while attempting to update the value of the `Reading` property in a `Sensor` resource.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the sensor hardware or connection.

Message and Arguments: "Invalid reading received from sensor '<Arg1>'."

- string*: The name or identifier of the sensor.

- This argument shall contain a string that identifies or describes the `Sensor` resource.

2.11.2 ReadingAboveLowerCriticalThreshold

Indicates that a sensor reading is no longer below the lower critical threshold, but is still outside of normal operating range.

- This message shall be used to indicate that the value of the `Reading` in a `Sensor` resource, previously below the threshold, is now above the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties. However, the value of the `Reading` property is still below at least one `Reading` property of another lower threshold and is not within the normal operating range.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is now above the <Arg4> lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

2.11.3 ReadingAboveLowerFatalThreshold

Indicates that a sensor reading is no longer below the lower fatal threshold, but is still outside of normal operating range.

- This message shall be used to indicate that the value of the `Reading` in a `Sensor` resource, previously below the threshold, is now above the value of the `Reading` property within the `LowerFatal` property. However, the value of the `Reading` property is still below at least one `Reading` property of another lower threshold and is not within the normal operating range.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is now above the <Arg4> lower fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

2.11.4 ReadingAboveUpperCautionThreshold

Indicates that a sensor reading is above the upper caution threshold.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is above the <Arg4> upper caution threshold."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.

- This argument shall contain a number that equals the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource.

2.11.5 ReadingAboveUpperCriticalThreshold

Indicates that a sensor reading is above the upper critical threshold.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the condition of the resources listed in `RelatedItem`.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is above the <Arg4> upper critical threshold."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

2.11.6 ReadingAboveUpperFatalThreshold

Indicates that a sensor reading is above the upper fatal threshold.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource is above the value of the `Reading` property within the `UpperFatal` property.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the condition of the resources listed in `RelatedItem`.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is above the <Arg4> upper fatal threshold."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

2.11.7 ReadingBelowLowerCautionThreshold

Indicates that a sensor reading is below the lower caution threshold.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is below the <Arg4> lower caution threshold."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

2.11.8 ReadingBelowLowerCriticalThreshold

Indicates that a sensor reading is below the lower critical threshold.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is below the <Arg4> lower critical threshold."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

2.11.9 ReadingBelowLowerFatalThreshold

Indicates that a sensor reading is below the lower fatal threshold.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource is below the value of the `Reading` property within the `LowerFatal` property.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is below the <Arg4> lower fatal threshold."

1. *string*: The name or identifier of the sensor.

- This argument shall contain a string that identifies or describes the `Sensor` resource.
- 2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
- 3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
- 4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

2.11.10 ReadingBelowUpperCriticalThreshold

Indicates that a sensor reading is no longer above the upper critical threshold, but is still outside of normal operating range.

- This message shall be used to indicate that the value of the `Reading` in a `Sensor` resource, previously above the threshold, is now below the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties. However, the value of the `Reading` property is still above at least one `Reading` property of another upper threshold and is not within the normal operating range.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is now below the <Arg4> upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

2.11.11 ReadingBelowUpperFatalThreshold

Indicates that a sensor reading is no longer above the upper fatal threshold, but is still outside of normal operating range.

- This message shall be used to indicate that the value of the `Reading` in a `Sensor` resource, previously above the threshold, is now below the value of the `Reading` property within the `UpperFatal` property. However, the value of the `Reading` property is still above at least one `Reading` property of another upper threshold and is not within the normal operating range.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the condition of the resources listed in `RelatedItem`.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is now below the <Arg4> upper fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.
4. *number*: The threshold value.
 - This argument shall contain a number that equals the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

2.11.12 ReadingCritical

Indicates that a sensor reading exceeds an internal critical level.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource exceeds the value of an internal critical level. This message shall be used for `Sensor` resources that do not contain readable critical threshold values, such as boolean trip points or similar hardware mechanisms. This message shall not be used if the `Reading` property within a corresponding critical threshold is available.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) exceeds the critical level."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.

2.11.13 ReadingNoLongerCritical

Indicates that a sensor reading no longer exceeds an internal critical level, but exceeds an internal warning level.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource no longer exceeds the value of an internal critical level, but the value still exceeds an internal warning level. This message shall be used for `Sensor` resources that do not contain readable critical threshold values, such as boolean trip points or similar hardware mechanisms. This message shall not be used if the `Reading` property within a corresponding critical threshold is available.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) no longer exceeds the critical level."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.

2.11.14 ReadingWarning

Indicates that a sensor reading exceeds an internal warning level.

- This message shall be used to indicate that the value of the `Reading` property in a `Sensor` resource exceeds the value of an internal warning level. This message shall be used for `Sensor` resources that do not contain readable caution threshold values, such as boolean trip points or similar hardware mechanisms. This message shall not be used if the `Reading` property within a corresponding caution threshold is available.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) exceeds the warning level."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.

2.11.15 SensorFailure

Indicates that the service cannot communicate with a sensor or has detected a failure.

- This message shall be used to indicate that the service is unable to communicate with a sensor or has otherwise detected a fault condition with the sensor.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the sensor hardware or connection.

Message and Arguments: "Sensor '<Arg1>' has failed."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.

2.11.16 SensorReadingNormalRange

Indicates that a sensor reading is now within normal operating range.

- This message shall be used to indicate that the value of the `Reading` property within a `Sensor` resource has returned to its normal operating range.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "Sensor '<Arg1>' reading of <Arg2> (<Arg3>) is within normal operating range."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.
2. *number*: The reading of the sensor.
 - This argument shall contain a number that equals the value of the `Reading` property of the `Sensor` resource.
3. *string*: The reading units of measure.
 - This argument shall contain a string that equals the value of the `ReadingUnits` property of the `Sensor` resource. Unitless readings should use `count`.

2.11.17 SensorRestored

Indicates that a sensor has been repaired or communications have been restored. It may also indicate that the service is receiving valid data from a sensor.

- This message shall be used to indicate that the service is able to communicate with a sensor that had previously failed, or the sensor has been repaired or replaced.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "Sensor '<Arg1>' has been restored."

1. *string*: The name or identifier of the sensor.
 - This argument shall contain a string that identifies or describes the `Sensor` resource.

2.12 StorageDevice 1.1.0

This registry defines the messages for storage devices.

Message	Severity	Description
BatteryCharging	Warning	A battery charging condition has been detected.
BatteryFailure	Critical	A battery failure condition has been detected.
BatteryMissing	Critical	A battery missing condition has been detected.
BatteryOK	OK	The health of a battery has changed to OK.
ControllerDegraded	Warning	A storage controller degraded condition has been detected.
ControllerFailure	Critical	A storage controller failure has been detected.
ControllerOK	OK	The storage controller health has changed to OK.
ControllerPasswordAccepted	OK	The storage controller password was entered.
ControllerPasswordRequired	Critical	The storage controller requires a password.
ControllerPortDegraded	Warning	A controller port degraded condition has been detected.
ControllerPortFailure	Critical	A controller port failure condition has been detected.
ControllerPortOK	OK	The health of a controller port has changed to OK.
ControllerPreviousError	Warning	A storage controller error has been detected prior to reboot.
DriveFailure	Critical	A drive failure condition has been detected.
DriveFailureCleared	OK	A previously detected failure condition on a drive has been cleared.
DriveInserted	OK	A drive has been inserted.
DriveMissing	Critical	A drive missing condition has been detected.
DriveMissingCleared	OK	A previous drive missing condition has been cleared.
DriveOK	OK	The health of a drive has changed to OK.
DriveOffline	Critical	A drive offline condition has been detected.
DriveOfflineCleared	OK	A drive offline condition has been cleared.
DrivePredictiveFailure	Warning	A predictive drive failure condition has been detected.
DrivePredictiveFailureCleared	OK	A previously detected predictive failure condition on a drive has been cleared.
DriveRemoved	Critical	A drive has been removed.
VolumeDegraded	Warning	The storage controller has detected a degraded volume condition.
VolumeFailure	Critical	The storage controller has detected a failed volume condition.

Message	Severity	Description
VolumeOK	OK	A volume health has changed to OK.
VolumeOffline	Critical	The storage controller has detected an offline volume condition.
VolumeOfflineCleared	OK	The storage controller has detected an online volume condition.
WriteCacheDataLoss	Critical	The write cache is reporting loss of data in posted-writes memory.
WriteCacheDegraded	Critical	The write cache state is degraded.
WriteCacheProtected	OK	A storage controller write cache state is in protected mode.
WriteCacheTemporarilyDegraded	Warning	The write cache state is temporarily degraded.

2.12.1 BatteryCharging

A battery charging condition has been detected.

- This message shall be used to indicate a battery is below the required capacity and is charging.

Version Added: 1.1.0

Severity: Warning

Resolution: No action needed. There may be reduced performance and features while the battery is charging.

Message and Arguments: "A charging condition for the battery located in '<Arg1>' has been detected."

- string*: The location of the battery.
 - The value of this argument shall be a string containing the battery location.

2.12.2 BatteryFailure

A battery failure condition has been detected.

- This message shall be used to indicate a battery failure condition has been detected.

Version Added: 1.1.0

Severity: Critical

Resolution: Ensure all cables are properly and securely connected. Replace the failed battery.

Message and Arguments: "A failure condition for the battery located in '<Arg1>' has been detected."

1. *string*: The location of the battery.
 - The value of this argument shall be a string containing the battery location.

2.12.3 BatteryMissing

A battery missing condition has been detected.

- This message shall be used to indicate a battery that is required by the configuration is missing.

Version Added: 1.1.0

Severity: Critical

Resolution: Attach the battery. Ensure all cables are properly and securely connected.

Message and Arguments: "The battery located in '<Arg1>' is missing."

1. *string*: The location of the battery.
 - The value of this argument shall be a string containing the battery location.

2.12.4 BatteryOK

The health of a battery has changed to OK.

- This message shall be used to indicate a battery health has changed to OK.

Version Added: 1.1.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "The health of the battery located in '<Arg1>' has changed to OK."

1. *string*: The location of the battery.
 - The value of this argument shall be a string containing the battery location.

2.12.5 ControllerDegraded

A storage controller degraded condition has been detected.

- This message shall be used to indicate that a storage controller degraded condition has been detected.

Version Added: 1.1.0

Severity: Warning

Resolution: Reseat the storage controller in the PCI slot. Update the controller to the latest firmware version. If the issue persists, replace the controller.

Message and Arguments: "A degraded condition for the storage controller located in '<Arg1>' has been detected due to reason '<Arg2>'."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.
2. *string*: The reason for the degraded controller.
 - The value of this argument shall be a string containing the reason for the degraded controller.

2.12.6 ControllerFailure

A storage controller failure has been detected.

- This message shall be used to indicate that a storage controller failure condition has been detected due to a hardware, communication, or firmware failure.

Version Added: 1.0.0

Severity: Critical

Resolution: Reseat the storage controller in the PCI slot. Update the controller to the latest firmware version. If the issue persists, replace the controller.

Message and Arguments: "A failure condition for the storage controller located in '<Arg1>' has been detected."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.7 ControllerOK

The storage controller health has changed to OK.

- This message shall be used to indicate that the storage controller health has changed to OK.

Version Added: 1.0.0

Severity: OK

Resolution: No action required.

Message and Arguments: "The health of the storage controller located in '<Arg1>' has changed to OK."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.8 ControllerPasswordAccepted

The storage controller password was entered.

- This message shall be used to indicate that the storage controller password was accepted. Normally this is used by storage controllers that require a controller password during boot to unlock encrypted volumes or drives.

Version Added: 1.1.0

Severity: OK

Resolution: No action required.

Message and Arguments: "A password was entered for the storage controller located in '<Arg1>'."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.9 ControllerPasswordRequired

The storage controller requires a password.

- This message shall be used to indicate that the storage controller requires a password. Normally this is used by storage controllers that require a controller password during boot to unlock encrypted volumes or drives.

Version Added: 1.1.0

Severity: Critical

Resolution: Enter the controller password.

Message and Arguments: "The storage controller located in '<Arg1>' requires a password."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.10 ControllerPortDegraded

A controller port degraded condition has been detected.

- This message shall be used to indicate a controller port degraded condition has been detected.

Version Added: 1.1.0

Severity: Warning

Resolution: Ensure all cables are properly and securely connected. Replace faulty cables.

Message and Arguments: "A degraded condition for the controller port located in '<Arg1>' has been detected due to reason '<Arg2>'."

1. *string*: The location of the controller port.
 - The value of this argument shall be a string containing the controller port location.
2. *string*: The reason for the degraded controller port.
 - The value of this argument shall be a string containing the reason for the degraded controller port.

2.12.11 ControllerPortFailure

A controller port failure condition has been detected.

- This message shall be used to indicate a controller port failure condition has been detected.

Version Added: 1.1.0

Severity: Critical

Resolution: Ensure all cables are properly and securely connected. Replace faulty cables.

Message and Arguments: "A failure condition for the controller port located in '<Arg1>' has been detected."

1. *string*: The location of the controller port.
 - The value of this argument shall be a string containing the controller port location.

2.12.12 ControllerPortOK

The health of a controller port has changed to OK.

- This message shall be used to indicate a controller port health has changed to OK.

Version Added: 1.1.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "The health of the controller port located in '<Arg1>' has changed to OK."

1. *string*: The location of the controller port.
 - The value of this argument shall be a string containing the controller port location.

2.12.13 ControllerPreviousError

A storage controller error has been detected prior to reboot.

- This message shall be used to indicate that an unreported storage controller error condition has been detected prior to reboot.

Version Added: 1.1.0

Severity: Warning

Resolution: Update the controller to the latest firmware version. If the issue persists, replace the controller.

Message and Arguments: "A previous error condition for the storage controller located in '<Arg1>' has been detected due to '<Arg2>'."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.
2. *string*: The reason for error.
 - The value of this argument shall be a string containing the reason for failure.

2.12.14 DriveFailure

A drive failure condition has been detected.

- This message shall be used to indicate a drive failure condition detected due to a drive not installed correctly or has failed.

Version Added: 1.0.0

Severity: Critical

Resolution: Ensure all cables are properly and securely connected. Ensure all drives are fully seated. Replace the defective cables, drive, or both.

Message and Arguments: "A failure condition for the drive located in '<Arg1>' has been detected."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.15 DriveFailureCleared

A previously detected failure condition on a drive has been cleared.

- This message shall be used to indicate a failure condition on a drive has been cleared, but any predictive failure or other conditions remain.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "A failure condition for the drive located in '<Arg1>' has been cleared."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.16 DriveInserted

A drive has been inserted.

- This message shall be used to indicate a drive has been inserted.

Version Added: 1.0.0

Severity: OK

Resolution: If the drive is not properly displayed, attempt to refresh the cached data.

Message and Arguments: "The drive located in '<Arg1>' was inserted."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.17 DriveMissing

A drive missing condition has been detected.

- This message shall be used to indicate a drive that is required by the configuration is missing. This message shall exist any time a volume is degraded or failed due to a missing drive.

Version Added: 1.1.0

Severity: Critical

Resolution: Ensure all cables are properly and securely connected. Ensure all drives are fully seated. Replace the defective cables, drive, or both. Delete the volume if no longer needed.

Message and Arguments: "A missing condition for the drive located in '<Arg1>' has been detected."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.18 DriveMissingCleared

A previous drive missing condition has been cleared.

- This message shall be used to indicate a previous drive missing condition has been cleared.

Version Added: 1.1.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "A missing condition for the drive located in '<Arg1>' has been cleared."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.19 DriveOK

The health of a drive has changed to OK.

- This message shall be used to indicate a drive health has changed to OK.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "The health of the drive located in '<Arg1>' has changed to OK."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.20 DriveOffline

A drive offline condition has been detected.

- This message shall be used to indicate a drive offline condition has been detected. This may occur when a drive is unconfigured, encrypted by a foreign controller, offline during a background operation, or unsupported.

Version Added: 1.1.0

Severity: Critical

Resolution: If the drive is unconfigured or needs an import, configure the drive. If the drive operation is in progress, wait for the operation to complete. If the drive is not supported, replace it.

Message and Arguments: "An offline condition for the the drive located in '<Arg1>' has been detected."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.21 DriveOfflineCleared

A drive offline condition has been cleared.

- This message shall be used to indicate a drive offline condition has been cleared.

Version Added: 1.1.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "An offline condition for the the drive located in '<Arg1>' has been cleared."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.22 DrivePredictiveFailure

A predictive drive failure condition has been detected.

- This message shall be used to indicate a drive predictive failure condition has been detected.

Version Added: 1.0.0

Severity: Warning

Resolution: If this drive is not part of a fault-tolerant volume, first back up all data, then replace the drive and restore all data afterward. If this drive is part of a fault-tolerant volume, replace this drive as soon as possible as long as the volume health is OK.

Message and Arguments: "A predictive failure condition for the drive located in '<Arg1>' has been detected."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.23 DrivePredictiveFailureCleared

A previously detected predictive failure condition on a drive has been cleared.

- This message shall be used to indicate a predictive failure condition on a drive has been cleared, but any other failure conditions remain.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "A predictive failure condition for the drive located in '<Arg1>' has been cleared."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.24 DriveRemoved

A drive has been removed.

- This message shall be used to indicate a drive has been removed.

Version Added: 1.0.0

Severity: Critical

Resolution: If the drive is still displayed, attempt to refresh the cached data.

Message and Arguments: "The drive located in '<Arg1>' was removed."

1. *string*: The location of the drive.
 - The value of this argument shall be a string containing the drive location.

2.12.25 VolumeDegraded

The storage controller has detected a degraded volume condition.

- This message shall be used to indicate a volume degraded condition was detected due to one or more drives not installed correctly, missing, or has failed.

Version Added: 1.0.0

Severity: Warning

Resolution: Ensure all cables are properly and securely connected. Replace failed drives.

Message and Arguments: "The volume '<Arg1>' attached to the storage controller located in '<Arg2>' is degraded."

1. *string*: The identifier of the volume.
 - The value of this argument shall be a string containing an identifier for the volume. The identifier may contain `Id`, `Name`, `Identifiers`, `LogicalUnitNumber`, or other data used to uniquely identify the volume.
2. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.26 VolumeFailure

The storage controller has detected a failed volume condition.

- This message shall be used to indicate a volume failure condition was detected due to one or more drives not installed correctly, missing, or has failed.

Version Added: 1.0.0

Severity: Critical

Resolution: Ensure all cables are properly and securely connected. Ensure all drives are fully seated and operational.

Message and Arguments: "The volume '<Arg1>' attached to the storage controller located in '<Arg2>' has failed."

1. *string*: The identifier of the volume.
 - The value of this argument shall be a string containing an identifier for the volume. The identifier may contain `Id`, `Name`, `Identifiers`, `LogicalUnitNumber`, or other data used to uniquely identify the volume.
2. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.27 VolumeOK

A volume health has changed to OK.

- This message shall be used to indicate a volume health has changed to OK.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "The health of volume '<Arg1>' that is attached to the storage controller located in '<Arg2>' has changed to OK."

1. *string*: The identifier of the volume.
 - The value of this argument shall be a string containing an identifier for the volume. The identifier may contain `Id`, `Name`, `Identifiers`, `LogicalUnitNumber`, or other data used to uniquely identify the volume.
2. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.28 VolumeOffline

The storage controller has detected an offline volume condition.

- This message shall be used to indicate a volume offline condition was detected. This may occur if the volume configuration is not supported or needs to be imported.

Version Added: 1.1.0

Severity: Critical

Resolution: Use storage software to enable, repair, or import the volume. You may also delete or move volume back to the original controller.

Message and Arguments: "The volume '<Arg1>' attached to the storage controller located in '<Arg2>' is offline."

1. *string*: The identifier of the volume.
 - The value of this argument shall be a string containing an identifier for the volume. The identifier may contain `Id`, `Name`, `Identifiers`, `LogicalUnitNumber`, or other data used to uniquely identify the volume.
2. *string*: The location of the storage controller.

- The value of this argument shall be a string containing the storage controller location.

2.12.29 VolumeOfflineCleared

The storage controller has detected an online volume condition.

- This message shall be used to indicate a volume online condition was detected.

Version Added: 1.1.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "The volume '<Arg1>' attached to the storage controller located in '<Arg2>' is online."

1. *string*: The identifier of the volume.
 - The value of this argument shall be a string containing an identifier for the volume. The identifier may contain `Id`, `Name`, `Identifiers`, `LogicalUnitNumber`, or other data used to uniquely identify the volume.
2. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.30 WriteCacheDataLoss

The write cache is reporting loss of data in posted-writes memory.

- This message shall be used to indicate the write cache is reporting loss of data in posted-writes cache memory due to loss of battery power when the server lost power or a hardware error.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the controller resource properties to determine cause of the write cache data loss.

Message and Arguments: "The write cache on the storage controller located in '<Arg1>' has data loss."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.31 WriteCacheDegraded

The write cache state is degraded.

- This message shall be used to indicate the write cache state degraded due to a missing battery or hardware failure.

Version Added: 1.0.0

Severity: Critical

Resolution: Check the controller to determine the cause of write cache degraded state, such as a missing battery or hardware failure.

Message and Arguments: "The write cache state on the storage controller located in '<Arg1>' is degraded."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.32 WriteCacheProtected

A storage controller write cache state is in protected mode.

- This message shall be used to indicate a controller write cache state is in protected mode.

Version Added: 1.0.0

Severity: OK

Resolution: No action needed.

Message and Arguments: "The write cache state on the storage controller located in '<Arg1>' is in protected mode."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.12.33 WriteCacheTemporarilyDegraded

The write cache state is temporarily degraded.

- This message shall be used to indicate write cache state temporarily degraded due to battery charging or data recovery rebuild operation pending.

Version Added: 1.0.0

Severity: Warning

Resolution: Check the controller to determine the cause of write cache temporarily degraded state, such as a battery is charging or a data recovery rebuild operation is pending.

Message and Arguments: "The write cache state on the storage controller located in '<Arg1>' is temporarily degraded."

1. *string*: The location of the storage controller.
 - The value of this argument shall be a string containing the storage controller location.

2.13 TaskEvent 1.0.3

This registry defines the messages for task related events.

Message	Severity	Description
TaskAborted	Critical	A task has completed with errors.
TaskCancelled	Warning	A task has been cancelled.
TaskCompletedOK	OK	A task has completed.
TaskCompletedWarning	Warning	A task has completed with warnings.
TaskPaused	Warning	A task has been paused.
TaskProgressChanged	OK	A task has changed progress.
TaskRemoved	Warning	A task has been removed.
TaskResumed	OK	A task has been resumed.
TaskStarted	OK	A task has started.

2.13.1 TaskAborted

A task has completed with errors.

- This message shall be used to indicate that a task has completed with errors and the `TaskState` property transitioned to `Exception`.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "The task with Id '<Arg1>' has completed with errors."

1. *string*: The `Id` of the task.

- This argument shall contain the value of the `Id` property of the task resource that has completed with errors.

2.13.2 TaskCancelled

A task has been cancelled.

- This message shall be used to indicate that work on a task has halted prior to completion due to an explicit request. The `TaskState` property transitioned to `Cancelled`.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "Work on the task with Id '<Arg1>' has been halted prior to completion due to an explicit request."

1. *string*: The `Id` of the task.
 - This argument shall contain the value of the `Id` property of the task resource that has been halted.

2.13.3 TaskCompletedOK

A task has completed.

- This message shall be used to indicate that a task has completed and the `TaskState` property transitioned to `Completed`.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The task with Id '<Arg1>' has completed."

1. *string*: The `Id` of the task.
 - This argument shall contain the value of the `Id` property of the task resource that has completed.

2.13.4 TaskCompletedWarning

A task has completed with warnings.

- This message shall be used to indicate that a task has completed with warnings and the `TaskState` property transitioned to `Completed`.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The task with Id '<Arg1>' has completed with warnings."

1. *string*: The `Id` of the task.
 - This argument shall contain the value of the `Id` property of the task resource that has completed with warnings.

2.13.5 TaskPaused

A task has been paused.

- This message shall be used to indicate that a task has been paused and the `TaskState` property transitioned to `Suspended` or `Interrupted`.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The task with Id '<Arg1>' has been paused."

1. *string*: The `Id` of the task.
 - This argument shall contain the value of the `Id` property of the task resource that has been paused.

2.13.6 TaskProgressChanged

A task has changed progress.

- This message shall be used to indicate that a task has changed progress.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The task with Id '<Arg1>' has changed to progress <Arg2> percent complete."

1. *string*: The `Id` of the task.
 - This argument shall contain the value of the `Id` property of the task resource that has changed progress.
2. *number*: The percent completion of the task.
 - This argument shall contain the percent completion of the task that has changed progress.

2.13.7 TaskRemoved

A task has been removed.

- This message shall be used to indicate that a task has been removed.

Version Added: 1.0.0

Severity: Warning

Resolution: None.

Message and Arguments: "The task with Id '<Arg1>' has been removed."

1. *string*: The `Id` of the task.
 - This argument shall contain the value of the `Id` property of the task resource that has been removed.

2.13.8 TaskResumed

A task has been resumed.

- This message shall be used to indicate that a task has resumed and the `TaskState` property transitioned to `Running` from `Suspended` OR `Interrupted`.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The task with Id '<Arg1>' has been resumed."

1. *string*: The `Id` of the task.
 - This argument shall contain the value of the `Id` property of the task resource that has been resumed.

2.13.9 TaskStarted

A task has started.

- This message shall be used to indicate that a task has started as the `TaskState` property transitions to `Starting` or `Running`.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The task with Id '<Arg1>' has started."

1. *string*: The `Id` of the task.
 - This argument shall contain the value of the `Id` property of the task resource that has started.

2.14 Update 1.0.1

This registry defines the update status and error messages.

Message	Severity	Description
ActivateFailed	Critical	Indicates that the component failed to activate the image.
AllTargetsDetermined	OK	Indicates that all target resources or devices for an update operation have been determined by the service.
ApplyFailed	Critical	Indicates that the component failed to apply an image.
ApplyingOnComponent	OK	Indicates that a component is applying an image.
AwaitToActivate	OK	Indicates that the resource or device is awaiting for an action to proceed with activating an image.
AwaitToUpdate	OK	Indicates that the resource or device is awaiting for an action to proceed with installing an image.
InstallingOnComponent	OK	Indicates that a component is installing an image.
OperationTransitionedToJob	OK	Indicates that the update operation transitioned to a job for managing the progress of the operation.
TargetDetermined	OK	Indicates that a target resource or device for a image has been determined for update.

Message	Severity	Description
TransferFailed	Critical	Indicates that the service failed to transfer an image to a component.
TransferringToComponent	OK	Indicates that the service is transferring an image to a component.
UpdateInProgress	OK	Indicates that an update is in progress.
UpdateSuccessful	OK	Indicates that a resource or device was updated.
VerificationFailed	Critical	Indicates that the component failed to verify an image.
VerifyingAtComponent	OK	Indicates that a component is verifying an image.

2.14.1 ActivateFailed

Indicates that the component failed to activate the image.

- This message shall be used to indicate that the component failed to activate the image.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "Activation of image '<Arg1>' on '<Arg2>' failed."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target activating the image.
 - This argument shall contain the URI of the resource or a unique name of the component activating the image.

2.14.2 AllTargetsDetermined

Indicates that all target resources or devices for an update operation have been determined by the service.

- This message shall be used to indicate that all target resources or devices for an update operation have been determined by the service.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "All the target device to be updated have been determined."

2.14.3 ApplyFailed

Indicates that the component failed to apply an image.

- This message shall be used to indicate that the component failed to apply an image.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "Installation of image '<Arg1>' to '<Arg2>' failed."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target applying the image.
 - This argument shall contain the URI of the resource or a unique name of the component applying the image.

2.14.4 ApplyingOnComponent

Indicates that a component is applying an image.

- This message shall be used to indicate that a component is applying an image.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "Image '<Arg1>' is being applied on '<Arg2>'."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target applying the image.
 - This argument shall contain the URI of the resource or a unique name of the component applying the image.

2.14.5 AwaitToActivate

Indicates that the resource or device is awaiting for an action to proceed with activating an image.

- This message shall be used to indicate that the resource or device is awaiting for an action to proceed with activating an image.

Version Added: 1.0.0

Severity: OK

Resolution: Perform the requested action to advance the update operation.

Message and Arguments: "Awaiting for an action to proceed with activating image '<Arg1>' on '<Arg2>'."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target awaiting for an action to activate the image.
 - This argument shall contain the URI of the resource or a unique name of the component awaiting for an action to activate the image.

2.14.6 AwaitToUpdate

Indicates that the resource or device is awaiting for an action to proceed with installing an image.

- This message shall be used to indicate that the resource or device is awaiting for an action to proceed with installing an image.

Version Added: 1.0.0

Severity: OK

Resolution: Perform the requested action to advance the update operation.

Message and Arguments: "Awaiting for an action to proceed with installing image '<Arg1>' on '<Arg2>'."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target awaiting for an action to install the image.
 - This argument shall contain the URI of the resource or a unique name of the component awaiting for an action to install the image.

2.14.7 InstallingOnComponent

Indicates that a component is installing an image.

- This message shall be used to indicate that a component is installing an image.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "Image '<Arg1>' is being installed on '<Arg2>'."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target installing the image.
 - This argument shall contain the URI of the resource or a unique name of the component installing the image.

2.14.8 OperationTransitionedToJob

Indicates that the update operation transitioned to a job for managing the progress of the operation.

- This message shall be used to indicate that the update operation transitioned to a job for managing the progress of the operation.

Version Added: 1.0.0

Severity: OK

Resolution: Follow the referenced job and monitor the job for further updates.

Message and Arguments: "The update operation has transitioned to the job at URI '<Arg1>'."

1. *string*: The job URI.
 - This argument shall contain the URI of the `Job` resource that is managing the progress of the update operation.

2.14.9 TargetDetermined

Indicates that a target resource or device for a image has been determined for update.

- This message shall be used to indicate that a target resource or device for a image has been determined for update.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "The target device '<Arg1>' will be updated with image '<Arg2>'."

1. *string*: The URI or name of the target that will be updated.
 - This argument shall contain the URI of the resource or a unique name of the device that will be updated.
2. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.

2.14.10 TransferFailed

Indicates that the service failed to transfer an image to a component.

- This message shall be used to indicate that the service failed to transfer an image to a component.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "Transfer of image '<Arg1>' to '<Arg2>' failed."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target receiving the image.
 - This argument shall contain the URI of the resource or a unique name of the component receiving the image.

2.14.11 TransferringToComponent

Indicates that the service is transferring an image to a component.

- This message shall be used to indicate that the service is transferring an image to a component.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "Image '<Arg1>' is being transferred to '<Arg2>'."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target receiving the image.
 - This argument shall contain the URI of the resource or a unique name of the component receiving the image.

2.14.12 UpdateInProgress

Indicates that an update is in progress.

- This message shall be used to indicate that an update is in progress.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "An update is in progress."

2.14.13 UpdateSuccessful

Indicates that a resource or device was updated.

- This message shall be used to indicate that a resource or device was updated.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "Device '<Arg1>' successfully updated with image '<Arg2>'."

1. *string*: The URI or name of the device that was updated.
 - This argument shall contain the URI of the resource or a unique name of the device that was updated.
2. *string*: The identifier for the image.

- This argument shall contain the identifier for the image.

2.14.14 VerificationFailed

Indicates that the component failed to verify an image.

- This message shall be used to indicate that the component failed to verify an image.

Version Added: 1.0.0

Severity: Critical

Resolution: None.

Message and Arguments: "Verification of image '<Arg1>' at '<Arg2>' failed."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target verifying the image.
 - This argument shall contain the URI of the resource or a unique name of the component verifying the image.

2.14.15 VerifyingAtComponent

Indicates that a component is verifying an image.

- This message shall be used to indicate that a component is verifying an image.

Version Added: 1.0.0

Severity: OK

Resolution: None.

Message and Arguments: "Image '<Arg1>' is being verified at '<Arg2>'."

1. *string*: The identifier for the image.
 - This argument shall contain the identifier for the image.
2. *string*: The URI or name of the target verifying the image.
 - This argument shall contain the URI of the resource or a unique name of the component verifying the image.

3 ANNEX A (informative) Change log

Version	Date	Description
2022.2	2022-08-04	Document built from Redfish message registries in DSP8011 release 2022.2.
2022.1	2022-04-07	Document built from Redfish message registries in DSP8011 release 2022.1.