Document Identifier: DSP2065

Date: 2026-08-07

Version: 2026.2

Redfish Message Registry Guide

Supersedes: 2026.1

Document Class: Informational

Document Status: Published

Document Language: en-US

Copyright Notice
Copyright © 2022-2026 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patent owner or claimant, and shall have no liability or responsibility for costs or losses incurred if a standard is withdrawn or modified after publication, and shall be indemnified and held harmless by any party implementing the standard from any and all claims of infringement by a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 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 AccountSecurity 1.2.0 7
2.2 AutomationNodeEvent 1.1.1 18
2.3 Base 1.24.0 22
2.4 Composition 1.2.0 88
2.5 Device 1.0.0 95
2.6 Environmental 1.5.1 104
2.7 EthernetFabric 1.1.1 179
2.8 EventService 1.0.0 184
2.9 Fabric 1.1.0 188
2.10 HeartbeatEvent 1.2.0 212
2.11 JobEvent 1.2.1 213
2.12 License 1.1.0 220
2.13 LogService 1.2.1 224
2.14 ManagerEvent 1.2.0 228
2.15 NetworkDevice 1.2.1 235
2.16 Platform 1.5.0 240
2.17 Power 1.4.0 266
2.18 ResourceEvent 1.5.0 319
2.19 SensorEvent 1.2.0 335
2.20 ServiceCommunications 1.2.0 348
2.21 StorageDevice 1.7.0 352
2.22 TaskEvent 1.1.0 375
2.23 Telemetry 1.2.0 380
2.24 Update 1.5.0 386
3 ANNEX A (informative) Change log 403

1 Overview

The Redfish standard comprises a set of specifications maintained by the Redfish Forum, a working group within 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 DMTF and the Redfish Forum by the following means:

- **Redfish User Forum:** https://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 DMTF and the 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 DMTF's Redfish Standards web site: https://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 `GET` responses 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 three 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 AccountSecurity 1.2.0

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.2	2026.1	2024.4

This registry defines the account security messages. These messages are intended for audit logs and security event notifications. Services shall not include these messages in responses to clients.

Message	Severity	Description
AccountCreated	OK	Indicates that an account was created.
AccountDisabled	Warning	Indicates that an account was disabled.
AccountEnabled	OK	Indicates that an account was enabled.
AccountLocked	Critical	Indicates that an account was locked due to failed authorization attempts.
AccountLockoutExpired	OK	Indicates that the lockout timer on an account expired and the account is now unlocked.
AccountRemoved	OK	Indicates that an account was removed.
AccountUnlocked	OK	Indicates that an account was manually unlocked.
InsufficientPrivilege	Critical	Indicates that a user with valid credentials attempted an operation without the necessary privileges.
InsufficientPrivilegeWithAddress *(v1.2+)*	Critical	Indicates that a user with valid credentials attempted an operation without the necessary privileges.
InvalidCredentials	Critical	Indicates that a user provided invalid credentials with a request.
InvalidCredentialsWithAddress *(v1.2+)*	Critical	Indicates that a user provided invalid credentials with a request.
ManagerAccountRoleChanged	OK	Indicates that the role for an account was changed.
PasswordModified	OK	Indicates that the password of an account was changed.

Message	Severity	Description
RolePrivilegeAdded	OK	Indicates that a privilege was added to a role.
RolePrivilegeRemoved	OK	Indicates that a privilege was removed from a role.
SuccessfulLogin	OK	Indicates that a user provided valid credentials and made a successful login.
UserCertificateAdded	OK	Indicates that a user certificate was added to an account.
UserCertificateRemoved	OK	Indicates that a user certificate was removed from an account.
UserCertificateReplaced	OK	Indicates that a user certificate was replaced on an account.
UserKeyAdded	OK	Indicates that a user key was added to an account.
UserKeyRemoved	OK	Indicates that a user key was removed from an account.

2.1.1 AccountCreated

Example: Account 'ContosoJoe' was created.

Indicates that an account was created.

- This message shall indicate that an account was created. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Account '`<1>`' was created."

1. *string*: The username of the account that was created.
   - This argument shall contain the username of the account that was created.

2.1.2 AccountDisabled

Example: Account 'ContosoJoe' was disabled.

Indicates that an account was disabled.

- This message shall indicate that an account was disabled. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "Account '`<1>`' was disabled."

1. *string*: The username of the account that was disabled.
    - This argument shall contain the username of the account that was disabled.

2.1.3 AccountEnabled

Example: Account 'ContosoJoe' was enabled.

Indicates that an account was enabled.

- This message shall indicate that an account was enabled. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Account '`<1>`' was enabled."

1. *string*: The username of the account that was enabled.
    - This argument shall contain the username of the account that was enabled.

2.1.4 AccountLocked

Example: Account 'ContosoJoe' was locked due to excessive failed authorization attempts.

Indicates that an account was locked due to failed authorization attempts.

- This message shall indicate that an account was locked due to failed authorization attempts. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "Account ' `<1>` ' was locked due to excessive failed authorization attempts."

1. *string*: The username of the account that was locked.
    - This argument shall contain the username of the account that was locked.

2.1.5 AccountLockoutExpired

Example: Account 'ContosoJoe' was unlocked following the expiration of a lockout timer.

Indicates that the lockout timer on an account expired and the account is now unlocked.

- This message shall indicate that the lockout timer on an account expired and the account is now unlocked. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Account ' `<1>` ' was unlocked following the expiration of a lockout timer."

1. *string*: The username of the account that was unlocked.
    - This argument shall contain the username of the account that was unlocked.

2.1.6 AccountRemoved

Example: Account 'ContosoJoe' was removed.

Indicates that an account was removed.

- This message shall indicate that an account was removed. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Account ' `<1>` ' was removed."

1. *string*: The username of the account that was removed.
    - This argument shall contain the username of the account that was removed.

2.1.7 AccountUnlocked

Example: Account 'ContosoJoe' was unlocked by an administrator.

Indicates that an account was manually unlocked.

- This message shall indicate that an account was manually unlocked. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Account ' `<1>` ' was unlocked by an administrator."

1. *string*: The username of the account that was unlocked.
    - This argument shall contain the username of the account that was unlocked.

2.1.8 InsufficientPrivilege

Example: 'ContosoJoe' attempted an operation over 'Redfish' with the privileges 'Read', but requires the privileges 'Write'.

Indicates that a user with valid credentials attempted an operation without the necessary privileges.

- This message shall indicate that a user with valid credentials attempted an operation without the necessary privileges. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "' `<1>` ' attempted an operation over ' `<2>` ' with the privileges ' `<3>` ', but requires the privileges ' `<4>` '."

1. *string*: The user that attempted the operation.
    - This argument shall contain the user that attempted the operation.
2. *string*: The protocol or interface over which the operation was attempted. Examples include 'Redfish', 'Web UI'. and 'SSH'.
    - This argument shall contain the protocol or interface over which the operation was attempted.

3. *string*: The privileges of the user that attempted the operation.
    - This argument shall contain the privileges of the user that attempted the operation.
4. *string*: The required privileges for the requested operation.
    - This argument shall contain the required privileges for the requested operation.

2.1.9 InsufficientPrivilegeWithAddress

Example: 'ContosoJoe' from '127.0.0.1' attempted an operation over 'Redfish' with the privileges 'Read', but requires the privileges 'Write'.

Indicates that a user with valid credentials attempted an operation without the necessary privileges.

- This message shall indicate that a user with valid credentials attempted an operation without the necessary privileges. Services shall not include this message in responses to clients.

Version Added: v1.2

Severity: Critical

Resolution: None.

Message and Arguments: "User '`<1>`' from '`<2>`' attempted an operation over '`<3>`' with the privileges '`<4>`', but requires the privileges '`<5>`'."

1. *string*: The user that attempted the operation.
    - This argument shall contain the user that attempted the operation.
2. *string*: The IP address from which the user attempted the operation.
    - This argument shall contain the IP address from which the user attempted the operation.
3. *string*: The protocol or interface over which the operation was attempted. Examples include 'Redfish', 'Web UI'. and 'SSH'.
    - This argument shall contain the protocol or interface over which the operation was attempted.
4. *string*: The privileges of the user that attempted the operation.
    - This argument shall contain the privileges of the user that attempted the operation.
5. *string*: The required privileges for the requested operation.
    - This argument shall contain the required privileges for the requested operation.

2.1.10 InvalidCredentials

Example: 'ContosoJoe' provided invalid credentials over 'HTTPS'.

Indicates that a user provided invalid credentials with a request.

- This message shall indicate that a user provided invalid credentials with a request, such as those found in the

`Authorization` and `X-Auth-Token` HTTP request headers. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "' `<1>` ' provided invalid credentials over ' `<2>` '."

1. *string*: The user that attempted the operation.
   - This argument shall contain the user that attempted the operation.
2. *string*: The protocol or interface over which the operation was attempted. Examples include 'Redfish', 'Web UI'. and 'SSH'.
   - This argument shall contain the protocol or interface over which the operation was attempted.

2.1.11 InvalidCredentialsWithAddress

Example: User 'ContosoJoe' from address '127.0.0.1' provided invalid credentials over 'HTTPS'.

Indicates that a user provided invalid credentials with a request.

- This message shall indicate that a user provided invalid credentials with a request, such as those found in the `Authorization` and `X-Auth-Token` HTTP request headers. Services shall not include this message in responses to clients.

Version Added: v1.2

Severity: Critical

Resolution: None.

Message and Arguments: "User ' `<1>` ' from address ' `<2>` ' provided invalid credentials over ' `<3>` '."

1. *string*: The user that attempted the operation.
   - This argument shall contain the user that attempted the operation.
2. *string*: The IP address from which the user attempted the operation.
   - This argument shall contain the IP address from which the user attempted the operation.
3. *string*: The protocol or interface over which the operation was attempted. Examples include 'Redfish', 'Web UI'. and 'SSH'.
   - This argument shall contain the protocol or interface over which the operation was attempted.

2.1.12 ManagerAccountRoleChanged

Example: Account 'ContosoJoe' has changed from role 'Administrator' to 'ReadOnly'.

Indicates that the role for an account was changed.

- This message shall indicate that the role for an account was changed. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Account '`<1>`' has changed from role '`<2>`' to '`<3>`'."

1. *string*: The username of the account that was modified.
   - This argument shall contain the username of the account that was modified.
2. *string*: The former role for the account.
   - This argument shall contain the former Redfish role for the account.
3. *string*: The new role for the account
   - This argument shall contain the new Redfish role for the account.

2.1.13 PasswordModified

Example: The password for account 'ContosoJoe' was changed.

Indicates that the password of an account was changed.

- This message shall indicate that the password of an account was changed. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The password for account '`<1>`' was changed."

1. *string*: The username of the account with the modified password.
   - This argument shall contain the username of the account with the modified password.

2.1.14 RolePrivilegeAdded

Example: Privilege 'Write' was added to role 'ReadOnly'.

Indicates that a privilege was added to a role.

- This message shall indicate that a privilege was added to a role. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Privilege '`<1>`' was added to role '`<2>`'."

1. *string*: The privilege that was added to the role.
   - This argument shall contain the privilege that was added to the role.
2. *string*: The role that was modified.
   - This argument shall contain the Redfish role that was modified.

2.1.15 RolePrivilegeRemoved

Example: Privilege 'Write' was removed from role 'ReadOnly'.

Indicates that a privilege was removed from a role.

- This message shall indicate that a privilege was removed from a role. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Privilege '`<1>`' was removed from role '`<2>`'."

1. *string*: The privilege that was removed from the role.
   - This argument shall contain the privilege that was removed from the role.
2. *string*: The role that was modified.
   - This argument shall contain the Redfish role that was modified.

2.1.16 SuccessfulLogin

Example: Successful login of user 'ContosoJoe' from '192.168.1.1' over 'HTTPS'.

Indicates that a user provided valid credentials and made a successful login.

- This message shall indicate that a user provided valid credentials and made a successful login. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Successful login of user ' `<1>` ' from ' `<2>` ' over ' `<3>` '."

1. *string*: The username of the account that made a successful login.
   - This argument shall contain the username of the account that made a successful login.
2. *string*: The IP address of the user that made a successful login.
   - This argument shall contain the IP address of the user that made a successful login.
3. *string*: The protocol or interface over which the login was made. Examples include 'Redfish', 'Web UI'. and 'SSH'.
   - This argument shall contain the protocol or interface over which the login was made.

2.1.17 UserCertificateAdded

Example: A user certificate was added to account 'ContosoJoe'.

Indicates that a user certificate was added to an account.

- This message shall indicate that a user certificate was added to an account. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A user certificate was added to account ' `<1>` '."

1. *string*: The username of the account with the certificate that was added.

    - This argument shall contain the username of the account with the certificate that was added.

2.1.18 UserCertificateRemoved

Example: A user certificate was removed from account 'ContosoJoe'.

Indicates that a user certificate was removed from an account.

- This message shall indicate that a user certificate was removed from an account. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A user certificate was removed from account '`<1>`'."

1. *string*: The username of the account with the certificate that was removed.
    - This argument shall contain the username of the account with the certificate that was removed.

2.1.19 UserCertificateReplaced

Example: A user certificate was replaced on account 'ContosoJoe'.

Indicates that a user certificate was replaced on an account.

- This message shall indicate that a user certificate was replaced on an account. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A user certificate was replaced on account '`<1>`'."

1. *string*: The username of the account with the certificate that was replaced.
    - This argument shall contain the username of the account with the certificate that was replaced.

2.1.20 UserKeyAdded

Example: A user key was added to account 'ContosoJoe'.

Indicates that a user key was added to an account.

- This message shall indicate that a user key was added to an account. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A user key was added to account '`<1>`'."

1. *string*: The username of the account with the key that was added.
    - This argument shall contain the username of the account with the key that was added.

2.1.21 UserKeyRemoved

Example: A user key was removed from account 'ContosoJoe'.

Indicates that a user key was removed from an account.

- This message shall indicate that a user key was removed from an account. Services shall not include this message in responses to clients.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A user key was removed from account '`<1>`'."

1. *string*: The username of the account with the key that was removed.
    - This argument shall contain the username of the account with the key that was removed.

2.2 AutomationNodeEvent 1.1.1

Version	*v1.1*	*v1.0*
Release	2026.1	2025.2

This registry defines the messages for Automation node related events.

Message	Severity	Description
NodeDone	OK	A node is done running.
NodeIdle	OK	A node is idle.
NodeRunning	OK	A node is running.
NodeStoppedCondition	Warning	A node has stopped because of conditions.
NodeStoppedError	Warning	A node has stopped with warnings or errors.
NodeWaiting	OK	A node is waiting.

2.2.1 NodeDone

Example: The node with Id 'Node1' is done running.

A node is done running.

- This message shall indicate that a node `NodeState` property transitioned to `Done`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The node with Id '`<1>`' is done running."

1. *string*: The `Id` of the node that is done running.
   - This argument shall contain the value of the `Id` property of the node resource that is done running.

2.2.2 NodeIdle

Example: The node with Id 'Node1' is idle.

A node is idle.

- This message shall indicate that a node `NodeState` property transitioned to `Idle`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The node with Id '`<1>`' is idle."

1. *string*: The `Id` of the node that is idle.
   - This argument shall contain the value of the `Id` property of the node resource that is idle.

2.2.3 NodeRunning

Example: The node with Id 'Node1' is running.

A node is running.

- This message shall indicate that a node `NodeState` property transitioned to one of the `Running` states.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The node with Id '`<1>`' is running."

1. *string*: The `Id` of the node that is running.
   - This argument shall contain the value of the `Id` property of the node resource that is running.

2.2.4 NodeStoppedCondition

Example: The node with Id 'Node1' has stopped because of conditions.

A node has stopped because of conditions.

- This message shall indicate that a node has stopped and the `NodeState` property transitioned to `ConditionStop`.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The node with Id '`<1>`' has stopped because of conditions."

1. *string*: The `Id` of the node that has stopped because of conditions.
   - This argument shall contain the value of the `Id` property of the node resource that has stopped

because of conditions.

2.2.5 NodeStoppedError

Example: The node with Id 'Node1' has stopped with warnings or errors.

A node has stopped with warnings or errors.

- This message shall indicate that a node has stopped and the `NodeState` property transitioned to `ErrorStop`.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The node with Id '`<1>`' has stopped with warnings or errors."

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

2.2.6 NodeWaiting

Example: The node with Id 'Node1' is waiting.

A node is waiting.

- This message shall indicate that a node `NodeState` property transitioned to `Waiting`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The node with Id '`<1>`' is waiting."

1. *string*: The `Id` of the node that is waiting.
   - This argument shall contain the value of the `Id` property of the node resource that is waiting.

2.3 Base 1.24.0

Version	*v1.24*	*v1.23*	*v1.22*	*v1.21*	*v1.20*	*v1.19*	*v1.18*	*v1.17*	*v1.16*	*v1.15*	...
Release	2026.2	2026.1	2025.3	2025.1	2024.4	2024.3	2023.3	2023.2	2024.1	2022.3	...

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.
AccessForbidden *(v1.20+, deprecated v1.22)*	Critical	Indicates that the request contained valid credentials, but the user does not have the necessary privileges to perform the requested operation.
AccessUnauthorized *(v1.20+)*	Critical	Indicates that the request did not contain valid credentials.
AccountForSessionNoLongerExists	OK	Indicates that the account for the session was removed, and so the session was 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 *(v1.10+)*	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 *(v1.12+)*	Warning	Indicates that the requested parameter value could not be completed because of a mismatch with other parameters or properties in the resource.
ActionParameterValueError *(v1.11+)*	Warning	Indicates that a parameter was given an invalid value.

Message	Severity	Description
ActionParameterValueFormatError	Warning	Indicates that a parameter was given the correct value type but the format of the value is not supported.
ActionParameterValueNotInList *(v1.10+)*	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.
ActionParameterValueOutOfRange *(v1.18+)*	Warning	Indicates that a parameter was given the correct value type but the value of that parameter is outside the supported range.
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.
AliasValueReassigned *(v1.24+)*	OK	Indicates that an alias value was reassigned from another resource.
ArraySizeTooLong *(v1.16+)*	Warning	Indicates that the size of the array exceeded the maximum number of elements.
ArraySizeTooShort *(v1.16+)*	Warning	Indicates that the size of the array is less than the minimum number of elements.
AuthenticationTokenRequired *(v1.15+)*	Critical	Indicates that the request could not be performed because an authentication token was not provided.
ChassisPowerStateOffRequired *(v1.6+)*	Warning	Indicates that the request requires a specified chassis to be powered off.
ChassisPowerStateOnRequired *(v1.6+)*	Warning	Indicates that the request requires a specified chassis to be powered on.
ConditionInRelatedResource *(v1.9+)*	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 *(deprecated v1.14)*	Critical	Indicates that a create operation 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 were met.
EmptyJSON *(v1.2+)*	Warning	Indicates that the request body contained an empty JSON object when one or more properties are expected in the body.
EventBufferExceeded *(v1.13+)*	Warning	Indicates undelivered events may have been lost due to a lack of buffer space in the service.
EventSubscriptionLimitExceeded	Critical	Indicates that an event subscription establishment was requested but the operation failed due to the number of simultaneous connections exceeding the limit of the implementation.
FailoverInProgress *(v1.24+)*	Warning	Indicates that a redundancy failover is in progress.

Message	Severity	Description
GeneralError	Critical	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.
GenerateSecretKeyRequired *(v1.19+)*	Critical	Indicates that the Time-based One-Time Password (TOTP) secret key needs to be generated for the account before accessing the service. The secret key can be generated with a `POST` to the `GenerateSecretKey` action for the `ManagerAccount` resource instance.
HeaderInvalid *(v1.13+)*	Critical	Indicates that a request header is invalid.
HeaderMissing *(v1.13+)*	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 *(v1.13+)*	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.
InvalidIndex	Warning	The index is not valid.
InvalidJSON *(v1.10+)*	Critical	Indicates that the request body contains invalid JSON.
InvalidObject	Critical	Indicates that the object in question is invalid according to the implementation. An example is a firmware update malformed URI.
InvalidScheme *(v1.23+)*	Warning	The scheme of the URI of a property or action parameter provided in the request body is not supported. For example, if `tftp` is provided for the scheme, but TFTP is not supported.
InvalidURI *(v1.10+)*	Critical	Indicates that the operation encountered a URI that does not correspond to a valid resource.
LicenseRequired *(v1.16+)*	Critical	Indicates that a license is required to perform the requested operation.
MalformedJSON	Critical	Indicates that the request body was malformed JSON.
MaximumErrorsExceeded *(v1.8+)*	Critical	Indicates that so many errors have occurred that the reporting service cannot return them all.
MissingOrMalformedPart *(v1.13+)*	Critical	Indicates that a multipart request is missing a required part or contains malformed parts.
NetworkNameResolutionNotConfigured *(v1.11+)*	Warning	Indicates that network-based name resolution is not configured on the service.

Message	Severity	Description
NetworkNameResolutionNotSupported *(v1.11+)*	Warning	Indicates the service does not support network-based name resolution.
NoOperation *(v1.4+)*	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.
OneTimePasscodeSent *(v1.17+)*	OK	Indicates that a required one-time passcode was sent to the user. The code should be provided as the Token property in the request to create a session.
OperationFailed *(v1.8+)*	Warning	Indicates that one of the internal operations necessary to complete the request failed. An example of this is when an internal service provider is unable to complete the request, such as in aggregation or RDE.
OperationNotAllowed *(v1.13+)*	Critical	Indicates that the HTTP method in the request is not allowed on this resource.
OperationTimeout *(v1.8+)*	Warning	Indicates that one of the internal operations necessary to complete the request timed out. An example of this is when an internal service provider is unable to complete the request, such as in aggregation or RDE.
PasswordChangeRequired *(v1.5+)*	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 `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.
PasswordComplexityNotMet *(v1.21+)*	Critical	Indicates that the password for the account provided does not meet the password complexity requirements of the service. The password for the account was not changed.
PasswordIncorrectLength *(v1.21+)*	Critical	Indicates that the password for the account provided does not meet the password length requirements of the service. The password for the account was not changed.
PasswordReuseTooRecent *(v1.21+)*	Critical	Indicates that the password for the account provided does not meet the password reuse requirements of the service. The password for the account was not changed.
PayloadTooLarge *(v1.13+)*	Critical	Indicates that the supplied payload is too large to be accepted by the service.
PreconditionFailed *(v1.8+)*	Critical	Indicates that the ETag supplied did not match the current ETag of the resource.
PreconditionRequired *(v1.8+)*	Critical	Indicates that the request did not provide the required precondition such as an `If-Match` or `If-None-Match` header or `@odata.etag` annotations.
PropertyDeprecated *(v1.10+)*	Warning	Indicates the property is deprecated.
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.

Message	Severity	Description
PropertyModified *(v1.19+)*	OK	Indicates that all properties listed in `RelatedProperties` in the message were successfully modified.
PropertyNotApplicable *(v1.24+)*	Warning	Indicates that an unknown property was included in the request body or a read-only property was included in the request body.
PropertyNotUpdated *(v1.12+)*	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 read-only property.
PropertyUnknown	Warning	Indicates that an unknown property was included in the request body.
PropertyValueConflict *(v1.6+)*	Warning	Indicates that the requested write of a property value could not be completed because of a conflict with another property value.
PropertyValueDeprecated *(v1.10+)*	Warning	Indicates that a property was given a deprecated value.
PropertyValueError *(v1.11+)*	Warning	Indicates that a property was given an invalid value.
PropertyValueExternalConflict *(v1.10+)*	Warning	Indicates that the requested write of a property value could not be completed due to conflicts with other configurations in the service managed with non-Redfish interfaces.
PropertyValueFormatError	Warning	Indicates that a property was given the correct value type but the format of the value is not supported.
PropertyValueIncorrect *(v1.7+)*	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, passwords that do not match the implementation constraints, or specifying an invalid URI in a reference property.
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 *(v1.3+)*	Warning	Indicates that a property was given the correct value type but the value of that property is outside the supported range.
PropertyValueResourceConflict *(v1.10+)*	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 *(v1.6+)*	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 *(v1.6+)*	Warning	Indicates that query is not supported with the given operation, such as when the `$expand` query is attempted with a `PATCH` operation.
QueryNotSupportedOnResource	Warning	Indicates that query is not supported on the given resource, such as when the `$skip` query is attempted on a resource that is not a collection.
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 that exceed what is possible for the supplied resource, such as when a page is requested that is beyond the last page.
QueryParameterUnsupported *(v1.15+)*	Warning	Indicates that a query parameter is not supported.
QueryParameterValueError *(v1.11+)*	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 format of the value is not supported.
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.
RedundancyDisabledWrongState *(v1.24+)*	Warning	Indicates that a resource or device is not in the required state for redundancy.
ResetRecommended *(v1.12+)*	Warning	Indicates that a component reset is recommended for error recovery while unaffected applications can continue running without any effect on accuracy and performance.
ResetRequired *(v1.6+)*	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 *(v1.7+)*	Warning	Indicates that the requested resource creation could not be completed because the service has a resource that conflicts with the request.
ResourceDeprecated *(v1.10+)*	Warning	Indicates the resource is deprecated.
ResourceExhaustion *(v1.2+)*	Critical	Indicates that a resource could not satisfy the request due to some unavailability of resources. An example is that available capacity was allocated.
ResourceInStandby *(v1.1+)*	Critical	Indicates that the request could not be performed because the resource is in standby.

Message	Severity	Description
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.
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 such as firmware updates.
ResourceNotFound *(v1.2+)*	Critical	Indicates that the operation expected a resource identifier that corresponds to an existing resource but one was not found.
ResourceTypeIncompatible *(v1.2+)*	Critical	Indicates that the resource type of the operation does not match that of the operation destination. Examples of this are a `POST` to a resource collection using the wrong resource type, an update where the `@odata.type` properties do not match, or a case of major version incompatibility.
RestrictedPrivilege *(v1.9+)*	Warning	Indicates that the operation was not successful because a privilege is restricted.
RestrictedRole *(v1.9+)*	Warning	Indicates that the operation was not successful because the role is restricted.
ServiceDisabled *(v1.11+)*	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 because 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 was requested but the operation failed due to the number of simultaneous sessions exceeding the limit of the implementation.
SessionTerminated *(v1.3+)*	OK	Indicates that the `DELETE` operation on the session resource resulted in the successful termination of the session.
SoftwareRequired *(v1.24+)*	Warning	Indicates that a requested operation cannot be performed because required software is not present.
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 *(v1.9+)*	Warning	Indicates the request failed because a set of `AccountTypes` or `OEMAccountTypes` was not accepted while `StrictAccountTypes` is set to `true`.
StringValueTooLong *(v1.2+)*	Warning	Indicates that a string value passed to the given resource was longer than the maximum allowed length. An example is when an implementation has imposed a shorter maximum length than that allowed by the specification.

Message	Severity	Description
StringValueTooShort *(v1.16+)*	Warning	Indicates that a string value passed to the given resource was shorter than the minimum required length. An example is when an implementation has imposed a greater minimum length than that required by the specification. This message also applies to cases of empty strings where the minimum string length is 1.
SubscriptionTerminated *(v1.5+)*	OK	An event subscription was terminated by the service. No further events will be delivered.
Success	OK	Indicates that all conditions of a successful operation were met.
UndeterminedFault *(v1.9+)*	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.

2.3.1 AccessDenied

Example: While attempting to establish a connection to '/redfish/v1/UpdateService/FirmwareInventory/1/Actions/FirmwareInventory.UpdateImage', the service denied access.

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

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

Version Added: v1.0

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 '`<1>`', the service denied access."

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

2.3.2 AccessForbidden

Example: Forbidden.

Indicates that the request contained valid credentials, but the user does not have the necessary privileges to perform the requested operation.

- This message shall indicate that the request contained valid credentials, but the user does not have the necessary privileges to perform the requested operation.

Deprecated in v1.22 and later. This message was deprecated in favor of the existing `InsufficientPrivilege` message.

Version Added: v1.20

Severity: Critical

Resolution: Resubmit the request with credentials that contain the necessary privileges to perform the requested operation.

Message and Arguments: "Forbidden."

2.3.3 AccessUnauthorized

Example: Unauthorized.

Indicates that the request did not contain valid credentials.

- This message shall indicate that the request did not contain valid credentials, such as those found in the `Authorization` and `X-Auth-Token` HTTP request headers.

Version Added: v1.20

Severity: Critical

Resolution: Resubmit the request with valid credentials.

Message and Arguments: "Unauthorized."

2.3.4 AccountForSessionNoLongerExists

Example: The account for the current session was removed, and so the current session was removed as well.

Indicates that the account for the session was removed, and so the session was removed as well.

- This message shall indicate that the account for the session was removed, and so the session was removed as well.

Version Added: v1.0

Severity: OK

Resolution: Attempt to connect with a valid account.

Message and Arguments: "The account for the current session was removed, and so the current session was removed as well."

2.3.5 AccountModified

Example: The account was successfully modified.

Indicates that the account was successfully modified.

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

Version Added: v1.0

Severity: OK

Resolution: None.

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

2.3.6 AccountNotModified

Example: The account modification request failed.

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

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

Version Added: v1.0

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."

2.3.7 AccountRemoved

Example: The account was successfully removed.

Indicates that the account was successfully removed.

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

Version Added: v1.0

Severity: OK

Resolution: None.

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

2.3.8 ActionDeprecated

Example: The action 'OldAction' is deprecated.

Indicates the action is deprecated.

- This message shall indicate that the action is deprecated.

Version Added: v1.10

Severity: Warning

Resolution: Refer to the schema guide for more information.

Message and Arguments: "The action ' `<1>` ' is deprecated."

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

2.3.9 ActionNotSupported

Example: The action 'DeactivateCloakingDevice' is not supported by the resource.

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

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

Version Added: v1.0

Severity: Critical

Resolution: Check the Actions property in the resource for the supported actions.

Message and Arguments: "The action ' `<1>` ' is not supported by the resource."

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

2.3.10 ActionParameterDuplicate

Example: The action 'Reset' was submitted with more than one value for the parameter 'ResetType'.

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

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

Version Added: v1.0

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 '`<1>`' was submitted with more than one value for the parameter '`<2>`'."

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.

2.3.11 ActionParameterMissing

Example: The action 'Reset' requires the parameter 'ResetType' to be present in the request body.

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

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

Version Added: v1.0

Severity: Critical

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

Message and Arguments: "The action '`<1>`' requires the parameter '`<2>`' 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.

2.3.12 ActionParameterNotSupported

Example: The parameter 'InvalidParameter' for the action 'Reset' is not supported on the target resource.

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

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

Version Added: v1.0

Severity: Warning

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

Message and Arguments: "The parameter ' `<1>` ' for the action ' `<2>` ' 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.

2.3.13 ActionParameterUnknown

Example: The action 'Reset' was submitted with the invalid parameter 'InvalidParameter'.

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

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

Version Added: v1.0

Severity: Warning

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

Message and Arguments: "The action ' `<1>` ' was submitted with the invalid parameter ' `<2>` '."

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.

2.3.14 ActionParameterValueConflict

Example: The parameter 'ResetType' with the requested value of 'ForceRestart' does not meet the constraints of the implementation.

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

Version Added: v1.12

Severity: Warning

Resolution: None.

Message and Arguments: "The parameter '`<1>`' with the requested value of '`<2>`' 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.

2.3.15 ActionParameterValueError

Example: The value 'InvalidValue' for the parameter 'ResetType' in the action 'Reset' is invalid.

Indicates that a parameter was given an invalid value.

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

Version Added: v1.11

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 '`<1>`' in the action '`<2>`' 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.

2.3.16 ActionParameterValueFormatError

Example: The value '2024-01-01' for the parameter 'ManufactureDate' in the action 'UpdateManufactureDate' is not a format that the parameter can accept.

Indicates that a parameter was given the correct value type but the format of the value is not supported.

- This message shall indicate that a parameter was given the correct value type but the format of the value is not supported.

Version Added: v1.0

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 '`<1>`' for the parameter '`<2>`' in the action '`<3>`' is not a format that 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.

2.3.17 ActionParameterValueNotInList

Example: The value 'InvalidValue' for the parameter 'ResetType' in the action 'Reset' is not in the list of acceptable values.

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 indicate that a parameter was given the correct value type but the value of that parameter was not supported.

Version Added: v1.10

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 ' `<1>` ' for the parameter ' `<2>` ' in the action ' `<3>` ' 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.

2.3.18 ActionParameterValueOutOfRange

Example: The value '-5' for the parameter 'MemorySizeGB' in the action 'UpdateMemorySize' is not in the supported range of acceptable values.

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

- This message shall indicate that a parameter was given the correct value type but the value of that parameter is outside the supported range.

Version Added: v1.18

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 ' `<1>` ' for the parameter ' `<2>` ' in the action ' `<3>` ' is not in the supported range of acceptable values."

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

2.3.19 ActionParameterValueTypeError

Example: The value '123' for the parameter 'ResetType' in the action 'Reset' is not a type that the parameter can accept.

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 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: v1.0

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 ' `<1>` ' for the parameter ' `<2>` ' in the action ' `<3>` ' is not a type that 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.

2.3.20 AliasValueReassigned

Example: The alias 'ProductionNetwork' was reassigned from the resource at /redfish/v1/Systems/1/EthernetInterfaces/NIC3.

Indicates that an alias value was reassigned from another resource.

- This message shall indicate that an alias value was reassigned from another resource, as defined in the 'Resource aliases' clause of the Redfish Specification. The URI of the resource to which the alias value was previously assigned is provided in the message.

Version Added: v1.24

Severity: OK

Resolution: None.

Message and Arguments: "The alias ' `<1>` ' was reassigned from the resource at `<2>` ."

1. *string*: The alias value.
   - This argument shall contain the value of the alias.

2. *string*: The URI of the resource to which the alias value was previously assigned.
    - This argument shall contain the URI of the resource to which this alias value was previously assigned.

2.3.21 ArraySizeTooLong

Example: The array provided for property 'AssociatedDrives' exceeds the size limit 100.

Indicates that the size of the array exceeded the maximum number of elements.

- This message shall indicate that the size of the array exceeded the maximum number of elements.

Version Added: v1.16

Severity: Warning

Resolution: Resubmit the request with an appropriate array size.

Message and Arguments: "The array provided for property '`<1>`' exceeds the size limit `<2>`."

1. *string*: The name of the property.
    - This argument shall contain the name of the property.
2. *number*: The maximum number of elements supported for the array.
    - This argument shall contain the maximum number of elements supported for the array for this property.

2.3.22 ArraySizeTooShort

Example: The array provided for property 'AssociatedDrives' is under the minimum size limit 1.

Indicates that the size of the array is less than the minimum number of elements.

- This message shall indicate that the size of the array is less than the minimum number of elements.

Version Added: v1.16

Severity: Warning

Resolution: Resubmit the request with an appropriate array size.

Message and Arguments: "The array provided for property '`<1>`' is under the minimum size limit `<2>`."

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

2. *number*: The minimum number of elements supported for the array.
    - This argument shall contain the minimum number of elements supported for the array for this property.

2.3.23 AuthenticationTokenRequired

Example: The request could not be performed because an authentication token was not provided.

Indicates that the request could not be performed because an authentication token was not provided.

- This message shall indicate that the request could not be performed because the user provided in the request requires multifactor authentication and an authentication token was not provided.

Version Added: v1.15

Severity: Critical

Resolution: Obtain an authentication token and resubmit the request.

Message and Arguments: "The request could not be performed because an authentication token was not provided."

2.3.24 ChassisPowerStateOffRequired

Example: The chassis with Id 'Chassis1' is required to be powered off to perform this request.

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

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

Version Added: v1.6

Severity: Warning

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

Message and Arguments: "The chassis with Id ' `<1>` ' is required to be powered off to perform this request."

1. *string*: The `Id` of the chassis that is required to be powered off.
    - This argument shall contain the value of the `Id` property of the `Chassis` resource that is required to be powered off.

2.3.25 ChassisPowerStateOnRequired

Example: The chassis with Id 'Chassis1' is required to be powered on to perform this request.

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

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

Version Added: v1.6

Severity: Warning

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

Message and Arguments: "The chassis with Id ' `<1>` ' is required to be powered on to perform this request."

1. *string*: The Id of the chassis that is required to be powered on.
   - This argument shall contain the value of the `Id` property of the `Chassis` resource that is required to be powered on.

2.3.26 ConditionInRelatedResource

Example: One or more conditions exist in a related resource. See the OriginOfCondition property.

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

- This message shall 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: v1.9

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."

2.3.27 CouldNotEstablishConnection

Example: The service failed to establish a connection with the URI '/redfish/v1/UpdateService/FirmwareInventory/1/Actions/FirmwareInventory.UpdateImage'.

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 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: v1.0

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 '`<1>`'."

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

2.3.28 CreateFailedMissingReqProperties

Example: The create operation failed because the required property 'Name' was missing from the request.

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

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

Deprecated in v1.14 and later. This message was deprecated in favor of `PropertyMissing`.

Version Added: v1.0

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 '`<1>`' was missing from the request."

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

2.3.29 CreateLimitReachedForResource

Example: The create operation failed because the resource has reached the limit of possible resources.

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

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

Version Added: v1.0

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."

2.3.30 Created

Example: The resource was created successfully.

Indicates that all conditions of a successful create operation were met.

- This message shall indicate that all conditions of a successful create operation were met.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource was created successfully."

2.3.31 EmptyJSON

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

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

- This message shall indicate that the request body contained an empty JSON object when one or more

properties are expected in the body.

Version Added: v1.2

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."

2.3.32 EventBufferExceeded

Example: Undelivered events may have been lost due to exceeding the event buffer.

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

- This message shall 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: v1.13

Severity: Warning

Resolution: None.

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

2.3.33 EventSubscriptionLimitExceeded

Example: The event subscription failed due to the number of simultaneous subscriptions exceeding the limit of the implementation.

Indicates that an event subscription establishment was requested but the operation failed due to the number of simultaneous connections exceeding the limit of the implementation.

- This message shall indicate that an event subscription establishment was requested but the operation failed due to the number of simultaneous connections exceeding the limit of the implementation.

Version Added: v1.0

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."

2.3.34 FailoverInProgress

Example: A failover is already in progress on redundancy member 'Manager1'.

Indicates that a redundancy failover is in progress.

- This message shall indicate that a redundancy failover is in progress and the requested operation cannot be completed at this time.

Version Added: v1.24

Severity: Warning

Resolution: Wait for the failover to complete before retrying the operation.

Message and Arguments: "A failover is already in progress on redundancy member '`<1>`'."

1. *string*: The name or identifier of the redundancy group member reporting the condition.
    - This argument shall contain a string that identifies the redundancy group member that detected the in-progress failover.

2.3.35 GeneralError

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

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 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: v1.0

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."

2.3.36 GenerateSecretKeyRequired

Example: The Time-based One-Time Password (TOTP) secret key for this account must be generated before access is granted. Perform the GenerateSecretKey action at URI '/redfish/v1/AccountService/Accounts/1/GenerateSecretKey' and retain the secret key from the response.

Indicates that the Time-based One-Time Password (TOTP) secret key needs to be generated for the account before accessing the service. The secret key can be generated with a `POST` to the `GenerateSecretKey` action for the `ManagerAccount` resource instance.

- This message shall indicate that the RFC6238-defined Time-based One-Time Password (TOTP) secret key needs to be generated for the account before accessing the service. The secret key can be generated with a `POST` to the `GenerateSecretKey` action for the `ManagerAccount` resource instance.

Version Added: v1.19

Severity: Critical

Resolution: Generate secret key for this account by performing the `GenerateSecretKey` action on the referenced URI and retaining the secret key from the action response to produce a Time-based One-Time Password (TOTP) for the `Token` property in future session creation requests.

Message and Arguments: "The Time-based One-Time Password (TOTP) secret key for this account must be generated before access is granted. Perform the GenerateSecretKey action at URI ' `<1>` ' and retain the secret key from the response."

1. *string*: The URI for the `GenerateSecretKey` action of the account that requires a new Time-based One-Time Password (TOTP) secret key.
    - This argument shall contain the URI for the `GenerateSecretKey` action of the account that requires a new RFC6238-defined Time-based One-Time Password (TOTP) secret key.

2.3.37 HeaderInvalid

Example: Header 'X-Auth-Token' is invalid.

Indicates that a request header is invalid.

- This message shall indicate that a request header is known but is invalid.

Version Added: v1.13

Severity: Critical

Resolution: Resubmit the request with a valid request header.

Message and Arguments: "Header '`<1>`' is invalid."

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

2.3.38 HeaderMissing

Example: Required header 'X-Auth-Token' is missing in the request.

Indicates that a required request header is missing.

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

Version Added: v1.13

Severity: Critical

Resolution: Resubmit the request with the required request header.

Message and Arguments: "Required header '`<1>`' is missing in the request."

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

2.3.39 InsufficientPrivilege

Example: There are insufficient privileges for the account or credentials associated with the current session to perform the requested operation.

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

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

Version Added: v1.0

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."

2.3.40 InsufficientStorage

Example: Insufficient storage or memory available to complete the request.

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

- This message shall 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: v1.13

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."

2.3.41 InternalError

Example: The request failed due to an internal service error. The service is still operational.

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

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

Version Added: v1.0

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."

2.3.42 InvalidIndex

Example: The index -5 is not a valid offset into the array.

The index is not valid.

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

Version Added: v1.0

Severity: Warning

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

Message and Arguments: "The index `<1>` is not a valid offset into the array."

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

2.3.43 InvalidJSON

Example: The request body submitted is invalid JSON starting at line 10 and could not be parsed by the receiving service.

Indicates that the request body contains invalid JSON.

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

Version Added: v1.10

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 `<1>` and could not be parsed by the receiving service."

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

2.3.44 InvalidObject

Example: The object at '/redfish/v1/Systems/InvalidSystem' is invalid.

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

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

Version Added: v1.0

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 ' `<1>` ' is invalid."

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

2.3.45 InvalidScheme

Example: The URI 'tftp://Contoso.com/file' contains a scheme that is not supported.

The scheme of the URI of a property or action parameter provided in the request body is not supported. For example, if `tftp` is provided for the scheme, but TFTP is not supported.

- This message shall indicate the scheme of the URI of a property or action parameter provided in the request body is not supported.

Version Added: v1.23

Severity: Warning

Resolution: Resubmit the request with an appropriate scheme for the URI.

Message and Arguments: "The URI ' `<1>` ' contains a scheme that is not supported."

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

2.3.46 InvalidURI

Example: The URI '/redfish/v1/Systems/InvalidSystem' was not found.

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

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

Version Added: v1.10

Severity: Critical

Resolution: Provide a valid URI and resubmit the request.

Message and Arguments: "The URI ' `<1>` ' was not found."

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

2.3.47 LicenseRequired

Example: A license is required for this operation: 'AdvancedPowerManagement'.

Indicates that a license is required to perform the requested operation.

- This message shall indicate that that a license is required to perform the requested operation.

Version Added: v1.16

Severity: Critical

Resolution: Install the requested license and resubmit the request.

Message and Arguments: "A license is required for this operation: '`<1>`'."

1. *string*: The vendor description of the license required to perform the operation.
    - This argument shall contain the vendor description of the license required to perform the operation.

2.3.48 MalformedJSON

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

Indicates that the request body was malformed JSON.

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

Version Added: v1.0

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."

2.3.49 MaximumErrorsExceeded

Example: Too many errors have occurred to report them all.

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

- The maximum number of errors that the implementation can report was 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: v1.8

Severity: Critical

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

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

2.3.50 MissingOrMalformedPart

Example: The multipart request contains malformed parts or is missing required parts.

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

- This message shall 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: v1.13

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."

2.3.51 NetworkNameResolutionNotConfigured

Example: Network name resolution is not configured on this service.

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

- This message shall indicate that network-based name resolution, such as DNS or WINS protocols, is not configured on the service.

Version Added: v1.11

Severity: Warning

Resolution: Configure the network-based 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 is not configured on this service."

2.3.52 NetworkNameResolutionNotSupported

Example: Resolution of network-based names is not supported by this service.

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

- This message shall 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: v1.11

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."

2.3.53 NoOperation

Example: The request body submitted contains no data to act upon and no changes to the resource took place.

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

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

Version Added: v1.4

Severity: Warning

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

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

2.3.54 NoValidSession

Example: There is no valid session established with the implementation.

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

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

Version Added: v1.0

Severity: Critical

Resolution: Establish a session before attempting any operations.

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

2.3.55 OneTimePasscodeSent

Example: A one-time passcode was sent to: 'joe@contoso.com'. Supply the passcode as the `Token` property in the request to create a session.

Indicates that a required one-time passcode was sent to the user. The code should be provided as the Token property in the request to create a session.

- This message shall indicate that a one-time passcode was generated by the service as the result of a request to create a session. The one-time passcode shall be sent to the delivery address associated with account credentials supplied in the request.

Version Added: v1.17

Severity: OK

Resolution: Obtain the one-time passcode sent to the delivery address and resubmit the request using the passcode as the `Token` property.

Message and Arguments: "A one-time passcode was sent to: '`<1>`'. Supply the passcode as the `Token` property in the request to create a session."

1. *string*: The delivery address for the one-time passcode. The address is at least partially redacted.
    - This argument shall contain the delivery address for the one-time passcode. The value shall be partially or fully redacted to prevent receipt by unauthorized clients.

2.3.56 OperationFailed

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

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

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

Version Added: v1.8

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."

2.3.57 OperationNotAllowed

Example: The HTTP method is not allowed on this resource.

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

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

Version Added: v1.13

Severity: Critical

Resolution: None.

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

2.3.58 OperationTimeout

Example: A timeout internal to the service occurred as part of the request. Partial results may have been returned.

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

- This message shall 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: v1.8

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."

2.3.59 PasswordChangeRequired

Example: 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 '/redfish/v1/Managers/1/Accounts/1' to complete this process.

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 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: v1.5

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 ' `<1>` ' 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.

2.3.60 PasswordComplexityNotMet

Example: The password provided for this account does not meet the password complexity requirements of the service.

Indicates that the password for the account provided does not meet the password complexity requirements of the service. The password for the account was not changed.

- This message shall indicate that the password for the account provided does not meet the password complexity requirements of the service.

Version Added: v1.21

Severity: Critical

Resolution: Resubmit the request with a password that meets the password complexity requirements as specified by the `PasswordGuidanceMessage` property in the `AccountService` resource.

Message and Arguments: "The password provided for this account does not meet the password complexity requirements of the service."

2.3.61 PasswordIncorrectLength

Example: The password provided for this account does not meet the password length requirements of the service.

Indicates that the password for the account provided does not meet the password length requirements of the service. The password for the account was not changed.

- This message shall indicate that the password for the account provided does not meet the length requirements of the service. The length of the supplied password is either too short or too long.

Version Added: v1.21

Severity: Critical

Resolution: Resubmit the request with a password that meets the password length requirements as specified by the `MinPasswordLength` and `MaxPasswordLength` properties in the `AccountService` resource.

Message and Arguments: "The password provided for this account does not meet the password length requirements of the service."

2.3.62 PasswordReuseTooRecent

Example: The password provided for this account does not meet the password reuse requirements of the service.

Indicates that the password for the account provided does not meet the password reuse requirements of the service. The password for the account was not changed.

- This message shall indicate that the password for the account provided does not meet the password reuse requirements of the service. The supplied password was used too recently.

Version Added: v1.21

Severity: Critical

Resolution: Resubmit the request with a password that meets the password reuse requirements specified by the `EnforcePasswordHistoryCount` property in the `AccountService` resource.

Message and Arguments: "The password provided for this account does not meet the password reuse requirements of the service."

2.3.63 PayloadTooLarge

Example: The supplied payload exceeds the maximum size supported by the service.

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

- This message shall 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: v1.13

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."

2.3.64 PreconditionFailed

Example: The ETag supplied did not match the ETag required to change this resource.

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 so 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: v1.8

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."

2.3.65 PreconditionRequired

Example: A precondition header or annotation is required to change this resource.

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: v1.8

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."

2.3.66 PropertyDeprecated

Example: The deprecated property 'OldProperty' was included in the request body.

Indicates the property is deprecated.

- This message shall indicate that the property is deprecated.

Version Added: v1.10

Severity: Warning

Resolution: Refer to the schema guide for more information.

Message and Arguments: "The deprecated property '`<1>`' 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.

2.3.67 PropertyDuplicate

Example: The property 'Name' was duplicated in the request.

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

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

Version Added: v1.0

Severity: Warning

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

Message and Arguments: "The property '`<1>`' was duplicated in the request."

1. *string*: The name of the duplicate property.
    - This argument shall contain the name of the duplicate property. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.68 PropertyMissing

Example: The property 'Name' is a required property and must be included in the request.

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

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

Version Added: v1.0

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 '`<1>`' 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. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.69 PropertyModified

Example: One or more properties were successfully modified.

Indicates that all properties listed in `RelatedProperties` in the message were successfully modified.

- This message shall indicate that the properties listed in RelatedProperties were successfully modified.

Version Added: v1.19

Severity: OK

Resolution: None.

Message and Arguments: "One or more properties were successfully modified."

2.3.70 PropertyNotApplicable

Example: The property 'InvalidPropertyName' is not supported for modification operations for the resource.

Indicates that an unknown property was included in the request body or a read-only property was included in the request body.

- This message shall indicate that an unknown property was included in the request body or a read-only property was included in the request body. Services should use the `PropertyUnknown` or `PropertyNotWritable` messages instead of this message to provide more specific information about the condition.

Version Added: v1.24

Severity: Warning

Resolution: Remove the property that is not applicable from the request body and resubmit the request if the operation failed.

Message and Arguments: "The property '`<1>`' is not supported for modification operations for the resource."

1. *string*: The name of the property that is not applicable.
    - This argument shall contain the property that is not applicable. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.71 PropertyNotUpdated

Example: The property 'Status' was not updated due to an internal service error. The service is still operational.

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

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

Version Added: v1.12

Severity: Critical

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

Message and Arguments: "The property '`<1>`' 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. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.72 PropertyNotWritable

Example: The property 'Manufacturer' is a read-only property and cannot be assigned a value.

Indicates that a property was given a value in the request body, but the property is a read-only property.

- This message shall indicate that a property was given a value in the request body, but the property is a read-only property.

Version Added: v1.0

Severity: Warning

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

Message and Arguments: "The property '`<1>`' 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. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.73 PropertyUnknown

Example: The property 'InvalidPropertyName' is not in the list of valid properties for the resource.

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

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

Version Added: v1.0

Severity: Warning

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

Message and Arguments: "The property '`<1>`' 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. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.74 PropertyValueConflict

Example: The property 'BootSourceOverrideTarget' could not be written because its value would conflict with the value of the 'BootSourceOverrideEnabled' property.

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

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

Version Added: v1.6

Severity: Warning

Resolution: None.

Message and Arguments: "The property '`<1>`' could not be written because its value would conflict with the value of the '`<2>`' 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.

2.3.75 PropertyValueDeprecated

Example: The value 'OldValue' for the property 'OldProperty' is deprecated.

Indicates that a property was given a deprecated value.

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

Version Added: v1.10

Severity: Warning

Resolution: Refer to the schema guide for more information.

Message and Arguments: "The value '`<1>`' for the property '`<2>`' 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.

2.3.76 PropertyValueError

Example: The value provided for the property 'Status' is not valid.

Indicates that a property was given an invalid value.

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

Version Added: v1.11

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 '`<1>`' is not valid."

1. *string*: The name of the property.
    - This argument shall contain the name of the property. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.77 PropertyValueExternalConflict

Example: The property 'BootSourceOverrideTarget' with the requested value of 'Pxe' could not be written because the value is not available due to a configuration conflict.

Indicates that the requested write of a property value could not be completed due to conflicts with other configurations in the service managed with non-Redfish interfaces.

- This message shall 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 with other configurations in the service managed with non-Redfish interfaces.

Version Added: v1.10

Severity: Warning

Resolution: None.

Message and Arguments: "The property '`<1>`' with the requested value of '`<2>`' 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.

2.3.78 PropertyValueFormatError

Example: The value '2024-01-01' for the property 'ManufactureDate' is not a format that the property can accept.

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

- This message shall indicate that a property was given the correct value type but the format of the value is not supported.

Version Added: v1.0

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 '`<1>`' for the property '`<2>`' is not a format that 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. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.79 PropertyValueIncorrect

Example: The property 'UserName' with the requested value of 'Admin!' could not be written because the value is not acceptable for the property.

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, passwords that do not match the implementation constraints, or specifying an invalid URI in a reference property.

- This message shall indicate that the requested write of a property could not be completed because of an incorrect value for the property.

Version Added: v1.7

Severity: Warning

Resolution: None.

Message and Arguments: "The property '<1>' with the requested value of '<2>' could not be written because the value is not acceptable for the property."

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.

2.3.80 PropertyValueModified

Example: The property 'Temperature' was assigned the value '20' due to modification by the service.

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 indicate that a property was given the correct value type but the value of that property was modified.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The property '<1>' was assigned the value '<2>' 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.

2.3.81 PropertyValueNotInList

Example: The value 'InvalidValue' for the property 'Status' is not in the list of acceptable values.

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

Version Added: v1.0

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 '`<1>`' for the property '`<2>`' 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. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.82 PropertyValueOutOfRange

Example: The value '-5' for the property 'MemorySizeGB' is not in the supported range of acceptable values.

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

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

Version Added: v1.3

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 '`<1>`' for the property '`<2>`' 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. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.83 PropertyValueResourceConflict

Example: The property 'BootSourceOverrideTarget' with the requested value of 'Pxe' could not be written because the value conflicts with the state or configuration of the resource at '/redfish/v1/Systems/1'.

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 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: v1.10

Severity: Warning

Resolution: None.

Message and Arguments: "The property '`<1>`' with the requested value of '`<2>`' could not be written because the value conflicts with the state or configuration of the resource at '`<3>`'."

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.

2.3.84 PropertyValueTypeError

Example: The value '123' for the property 'Manufacturer' is not a type that the property can accept.

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 indicate that a property was given the wrong value type.

Version Added: v1.0

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 '`<1>`' for the property '`<2>`' is not a type that 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. The value should follow RFC6901-defined JSON pointer notation rules.

2.3.85 QueryCombinationInvalid

Example: Two or more query parameters in the request cannot be used together.

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

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

Version Added: v1.6

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."

2.3.86 QueryNotSupported

Example: Querying is not supported by the implementation.

Indicates that query is not supported on the implementation.

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

Version Added: v1.0

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."

2.3.87 QueryNotSupportedOnOperation

Example: Querying is not supported with the requested operation.

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 indicate that query is not supported with the given operation.

Version Added: v1.6

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."

2.3.88 QueryNotSupportedOnResource

Example: Querying is not supported on the requested resource.

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 indicate that query is not supported on the given resource.

Version Added: v1.0

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."

2.3.89 QueryParameterOutOfRange

Example: The value '1000' for the query parameter '$top' is out of range 100.

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 that exceed what is possible for the supplied resource, such as when a page is requested that is beyond the last page.

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

Version Added: v1.0

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 number that is within the range of valid pages.

Message and Arguments: "The value ' `<1>` ' for the query parameter ' `<2>` ' is out of range `<3>`."

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.

2.3.90 QueryParameterUnsupported

Example: Query parameter 'InvalidParameter' is not supported.

Indicates that a query parameter is not supported.

- This message shall indicate that a query parameter is not supported by this service.

Version Added: v1.15

Severity: Warning

Resolution: Correct or remove the query parameter and resubmit the request.

Message and Arguments: "Query parameter ' `<1>` ' is not supported."

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

2.3.91 QueryParameterValueError

Example: The value for the parameter '$top' is invalid.

Indicates that a query parameter was given an invalid value.

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

Version Added: v1.11

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 '`<1>`' is invalid."

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

2.3.92 QueryParameterValueFormatError

Example: The value '2024-01-01' for the query parameter '$filter' is not a format that the parameter can accept.

Indicates that a query parameter was given the correct value type but the format of the value is not supported.

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

Version Added: v1.0

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 '`<1>`' for the query parameter '`<2>`' is not a format that 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.

2.3.93 QueryParameterValueTypeError

Example: The value '123' for the query parameter '$filter' is not a type that the parameter can accept.

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 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: v1.0

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 ' `<1>` ' for the query parameter ' `<2>` ' is not a type that 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.

2.3.94 RedundancyDisabledWrongState

Example: Redundancy member 'Manager1' detected that 'System1' is not in the required state for redundancy.

Indicates that a resource or device is not in the required state for redundancy.

- This message shall indicate that a resource or device is not in the required state to establish or maintain redundancy.

Version Added: v1.24

Severity: Warning

Resolution: Ensure the resource or device is in the required state for redundancy and retry the operation.

Message and Arguments: "Redundancy member ' `<1>` ' detected that ' `<2>` ' is not in the required state for redundancy."

1. *string*: The name or identifier of the redundancy group member reporting the condition.
   - This argument shall contain a string that identifies the redundancy group member that detected the incorrect state.
2. *string*: The name or identifier of the resource or device that is not in the required state.
   - This argument shall contain a string that identifies the resource or device that is not in the required state for redundancy.

2.3.95 ResetRecommended

Example: In order to recover from errors, a component reset is recommended with the Reset action URI '/redfish/v1/Systems/1/Actions/ComputerSystem.Reset' and ResetType 'ForceRestart'.

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

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

Version Added: v1.12

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 '`<1>`' and ResetType '`<2>`'."

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 effect on accuracy and performance. A reset action `POST` shall contain this `ResetType` parameter.

2.3.96 ResetRequired

Example: In order to apply changes, recover from errors, or complete the operation, a component reset is required with the Reset action URI '/redfish/v1/Systems/1/Actions/ComputerSystem.Reset' and ResetType 'ForceRestart'.

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

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

Version Added: v1.6

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 '`<1>`' and ResetType '`<2>`'."

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.

2.3.97 ResourceAlreadyExists

Example: The requested resource of type 'Drive' with the property 'PredictedMediaLifeLeftThreshold' with the value '20' already exists.

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

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

Version Added: v1.0

Severity: Critical

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

Message and Arguments: "The requested resource of type ' `<1>` ' with the property ' `<2>` ' with the value ' `<3>` ' 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`.

2.3.98 ResourceAtUriInUnknownFormat

Example: The resource at '/redfish/v1/UpdateService/FirmwareInventory/1/Actions/FirmwareInventory.UpdateImage' is in a format not recognized by the service.

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 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: v1.0

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 ' `<1>` ' is in a format not recognized by the service."

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

2.3.99 ResourceAtUriUnauthorized

Example: While accessing the resource at '/redfish/v1/UpdateService/FirmwareInventory/1/Actions/FirmwareInventory.UpdateImage', the service received an authorization error 'Unauthorized'.

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

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

Version Added: v1.0

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 ' `<1>` ', the service received an authorization error ' `<2>` '."

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.

2.3.100 ResourceCannotBeDeleted

Example: The delete request failed because the resource requested cannot be deleted.

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

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

Version Added: v1.0

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."

2.3.101 ResourceCreationConflict

Example: The resource could not be created. The service has a resource at URI '/redfish/v1/Systems/1' that conflicts with the creation request.

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

- This message shall 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: v1.7

Severity: Warning

Resolution: None.

Message and Arguments: "The resource could not be created. The service has a resource at URI '`<1>`' 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.

2.3.102 ResourceDeprecated

Example: The operation was performed on a deprecated resource '/redfish/v1/OldResource'.

Indicates the resource is deprecated.

- This message shall indicate that the resource is deprecated.

Version Added: v1.10

Severity: Warning

Resolution: Refer to the schema guide for more information.

Message and Arguments: "The operation was performed on a deprecated resource '`<1>`'."

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

2.3.103 ResourceExhaustion

Example: The resource at '/redfish/v1/UpdateService/FirmwareInventory/1' was unable to satisfy the request due to unavailability of resources.

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

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

Version Added: v1.2

Severity: Critical

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

Message and Arguments: "The resource '`<1>`' 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.

2.3.104 ResourceInStandby

Example: The request could not be performed because the resource is in standby.

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

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

Version Added: v1.1

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."

2.3.105 ResourceInUse

Example: The change to the requested resource failed because the resource is in use or in transition.

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 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: v1.0

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."

2.3.106 ResourceMissingAtURI

Example: The resource at the URI '/redfish/v1/UpdateService/FirmwareInventory/1/Actions/FirmwareInventory.UpdateImage' was not found.

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 such as firmware updates.

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

Version Added: v1.0

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 '`<1>`' was not found."

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

2.3.107 ResourceNotFound

Example: The requested resource of type 'Drive' named 'Drive1' was not found.

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

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

Version Added: v1.2

Severity: Critical

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

Message and Arguments: "The requested resource of type ' `<1>` ' named ' `<2>` ' 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.

2.3.108 ResourceTypeIncompatible

Example: The @odata.type of the request body '#ManagerAccount.v1_3_0.ManagerAccount' is incompatible with the @odata.type of the resource, which is '#ManagerAccount.v1_2_0.ManagerAccount'.

Indicates that the resource type of the operation does not match that of the operation destination. Examples of this are a `POST` to a resource collection using the wrong resource type, an update where the `@odata.type` properties do not match, or a case of major version incompatibility.

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

Version Added: v1.2

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 ' `<1>` ' is incompatible with the @odata.type of the resource, which is ' `<2>` '."

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.

2.3.109 RestrictedPrivilege

Example: The operation was not successful because the privilege 'Login' is restricted.

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

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

Version Added: v1.9

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 '`<1>`' is restricted."

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

2.3.110 RestrictedRole

Example: The operation was not successful because the role 'ReadOnly' is restricted.

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

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

Version Added: v1.9

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 '`<1>`' is restricted."

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

2.3.111 ServiceDisabled

Example: The operation failed because the service at '/redfish/v1/AccountService' is disabled and cannot accept requests.

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

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

Version Added: v1.11

Severity: Warning

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

Message and Arguments: "The operation failed because the service at '`<1>`' is disabled and cannot accept requests."

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

2.3.112 ServiceInUnknownState

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

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

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

Version Added: v1.0

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."

2.3.113 ServiceShuttingDown

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

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

- This message shall indicate that the operation failed because the service is shutting down.

Version Added: v1.0

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."

2.3.114 ServiceTemporarilyUnavailable

Example: The service is temporarily unavailable. Retry in 30 seconds.

Indicates the service is temporarily unavailable.

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

Version Added: v1.0

Severity: Critical

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

Message and Arguments: "The service is temporarily unavailable. Retry in `<1>` 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.

2.3.115 SessionLimitExceeded

Example: The session establishment failed due to the number of simultaneous sessions exceeding the limit of the implementation.

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

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

Version Added: v1.0

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."

2.3.116 SessionTerminated

Example: The session was successfully terminated.

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

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

Version Added: v1.3

Severity: OK

Resolution: None.

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

2.3.117 SoftwareRequired

Example: The operation requires software 'System Software' to be present.

Indicates that a requested operation cannot be performed because required software is not present.

- This message shall indicate that a requested operation cannot be performed because required software is not present in the service.

Version Added: v1.24

Severity: Warning

Resolution: Ensure that the required software is installed and retry the request.

Message and Arguments: "The operation requires software ' `<1>` ' to be present."

1. *string*: The name or identifier of the software.
    - This argument shall contain a string that identifies or describes the software.

2.3.118 SourceDoesNotSupportProtocol

Example: The other end of the connection at '/redfish/v1/UpdateService/FirmwareInventory/1/Actions/FirmwareInventory.UpdateImage' does not support the specified protocol 'HTTPS'.

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 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: v1.0

Severity: Critical

Resolution: Change protocols or URIs.

Message and Arguments: "The other end of the connection at '`<1>`' does not support the specified protocol `<2>`."

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.

2.3.119 StrictAccountTypes

Example: The request could not be fulfilled with the account types included in property 'AccountTypes' because the property StrictAccountTypes is set to true.

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

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

Version Added: v1.9

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 could not be fulfilled with the account types included in property '`<1>`' because the property StrictAccountTypes is set to true."

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

2.3.120 StringValueTooLong

Example: The string 'This is a very long string' exceeds the length limit 10.

Indicates that a string value passed to the given resource was longer than the maximum allowed length. An example is when an implementation has imposed a shorter maximum length than that allowed by the specification.

- This message shall indicate that a string value passed to the given resource was longer than the maximum allowed length.

Version Added: v1.2

Severity: Warning

Resolution: Resubmit the request with an appropriate string length.

Message and Arguments: "The string '`<1>`' exceeds the length limit `<2>`."

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.

2.3.121 StringValueTooShort

Example: The string 'Too short' was under the minimum required length 16.

Indicates that a string value passed to the given resource was shorter than the minimum required length. An example is when an implementation has imposed a greater minimum length than that required by the specification. This message also applies to cases of empty strings where the minimum string length is 1.

- This message shall indicate that a string value passed to the given resource was shorter than the minimum required length.

Version Added: v1.16

Severity: Warning

Resolution: Resubmit the request with an appropriate string length.

Message and Arguments: "The string '`<1>`' was under the minimum required length `<2>`."

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

2. *number*: The minimum string length.
    - This argument shall contain the minimum supported string length for this property.

2.3.122 SubscriptionTerminated

Example: The event subscription was terminated.

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

- This message shall indicate that an event subscription was terminated by the service.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "The event subscription was terminated."

2.3.123 Success

Example: The request completed successfully.

Indicates that all conditions of a successful operation were met.

- This message shall indicate that all conditions of a successful operation were met.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The request completed successfully."

2.3.124 UndeterminedFault

Example: An undetermined fault condition was reported by 'Service1'.

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 indicate that a fault or error condition was reported to the service, but the source of the fault cannot be determined or is unknown to the service.

Version Added: v1.9

Severity: Critical

Resolution: None.

Message and Arguments: "An undetermined fault condition was reported by '`<1>`'."

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.

2.3.125 UnrecognizedRequestBody

Example: The service detected a malformed request body that it was unable to interpret.

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

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

Version Added: v1.0

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."

2.4 Composition 1.2.0

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.1	2021.1	2019.2

This registry defines the messages for composition related events.

Message	Severity	Description
ConstrainedResourceAlreadyReserved *(v1.1+)*	Critical	Indicates that the requested resources are already reserved in response to a constrained composition request.

Message	Severity	Description
EmptyManifest *(v1.1+)*	Warning	Indicates that the manifest contains no stanzas or that a stanza in the manifest contains no request.
IncompatibleZone *(v1.1+)*	Critical	Indicates that not all referenced resource blocks are in the same resource zone.
NoResourceMatch *(v1.1+)*	Critical	Indicates that the service could not find a matching resource based on the given parameters.
ResourceBlockChanged	OK	Indicates that a resource block has changed. This is not used whenever there is another event message for that specific change, such as when only the state has changed.
ResourceBlockCompositionStateChanged	OK	Indicates that the composition state of a resource block has changed, specifically the value of the `CompositionState` property within `CompositionStatus`.
ResourceBlockInUse	Warning	Indicates that the composition request contains a resource block that is unable to participate in more compositions.
ResourceBlockInvalid *(v1.1+)*	Critical	Indicates that the `Id` of a referenced resource block is no longer valid.
ResourceBlockNotFound *(v1.1+)*	Critical	Indicates that the referenced resource block was not found.
ResourceBlockStateChanged	OK	Indicates that the state of a resource block has changed, specifically the value of the `State` property within `Status`.
ResourceZoneMembershipChanged	OK	Indicates that the membership of a resource zone has changed due to resource blocks being added or removed from the resource zone.
SpecifiedResourceAlreadyReserved *(v1.1+)*	Critical	Indicates that a resource block is already reserved in response to a specific composition request.
UnableToProcessStanzaRequest *(v1.1+)*	Critical	Indicates that the manifest provided for the `Compose` action contains a stanza with `Content` that could not be processed.

2.4.1 ConstrainedResourceAlreadyReserved

Example: The requested resources are reserved under reservation 'Reservation1'.

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

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

Version Added: v1.1

Severity: Critical

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

Message and Arguments: "The requested resources are reserved under reservation '`<1>`'."

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

2.4.2 EmptyManifest

Example: The provided manifest is empty or a stanza in the manifest contains no request.

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

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

Version Added: v1.1

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."

2.4.3 IncompatibleZone

Example: The requested resource blocks span multiple resource zones.

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

- This message shall indicate that not all referenced `ResourceBlock` resources are in the same resource zone.

Version Added: v1.1

Severity: Critical

Resolution: Request resource blocks from the same resource zone.

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

2.4.4 NoResourceMatch

Example: The requested resources of type 'Processor' are not available for allocation.

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

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

Version Added: v1.1

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 '`<1>`' are not available for allocation."

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

2.4.5 ResourceBlockChanged

Example: Resource block 'Block1' has changed on the service.

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

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Resource block '`<1>`' has changed on the service."

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

2.4.6 ResourceBlockCompositionStateChanged

Example: The composition status of the resource block 'Block1' has changed.

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

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The composition status of the resource block '`<1>`' has changed."

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

2.4.7 ResourceBlockInUse

Example: Resource block 'Block1' cannot be part of any new compositions.

Indicates that the composition request contains a resource block that is unable to participate in more compositions.

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

Version Added: v1.0

Severity: Warning

Resolution: Remove the resource block from the request and resubmit the request.

Message and Arguments: "Resource block '`<1>`' cannot be part of any new compositions."

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

2.4.8 ResourceBlockInvalid

Example: Resource block 'Block1' is not valid.

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

- This message shall indicate that the `Id` of a referenced `ResourceBlock` resource is no longer valid.

Version Added: v1.1

Severity: Critical

Resolution: Remove the resource block and resubmit the request.

Message and Arguments: "Resource block '`<1>`' is not valid."

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

2.4.9 ResourceBlockNotFound

Example: Resource block 'Block1' was not found.

Indicates that the referenced resource block was not found.

- This message shall indicate that the referenced `ResourceBlock` resource was not found.

Version Added: v1.1

Severity: Critical

Resolution: Remove the resource block and resubmit the request.

Message and Arguments: "Resource block '`<1>`' was not found."

1. *string*: The URI of the resource block.
    - This argument shall contain the value of the URI of the `ResourceBlock` resource that was not found.

2.4.10 ResourceBlockStateChanged

Example: The state of resource block 'Block1' has changed.

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

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The state of resource block '`<1>`' has changed."

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

2.4.11 ResourceZoneMembershipChanged

Example: The membership of resource zone 'Zone1' has been changed.

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

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The membership of resource zone '`<1>`' has been changed."

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

2.4.12 SpecifiedResourceAlreadyReserved

Example: Resource block 'Block1' is already reserved under reservation 'Reservation1'.

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

- This message shall indicate that a `ResourceBlock` resource is already reserved in response to a specific composition request.

Version Added: v1.1

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 '`<1>`' is already reserved under reservation '`<2>`'."

1. *string*: The `Id` of the resource block.

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

2.4.13 UnableToProcessStanzaRequest

Example: The provided manifest for the Compose action of type 'Simple' contains a stanza with Id of value 'Stanza1' with a Content parameter that could not be processed.

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

- This message shall indicate that the manifest provided for the `Compose` action contains a stanza with `Content` that could not be processed.

Version Added: v1.1

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 '`<1>`' contains a stanza with Id of value '`<2>`' with a Content parameter that could not be processed."

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

2.5 Device 1.0.0

Version	*v1.0*
Release	2026.2

This registry defines messages for individual devices or components in a larger system.

Message	Severity	Description
AssemblyDataInaccessible	Warning	Indicates that the assembly data, historically 'FRU data', for a device is inaccessible.
AssemblyDataMalformed	Warning	Indicates that the assembly data, historically 'FRU data', for a device is malformed, such as due to a checksum error.
BootTimeout	Critical	Indicates that a device encountered a boot timeout. This occurs when a device attempts to boot to an image that passed initial verification and other policy checks, but could not complete its boot sequence.
Disabled	Warning	Indicates that a device was disabled or halted.
ErrorDetected	Critical	Indicates that an error was detected on a device.
FirmwareIntegrityFailure	Critical	Indicates that a device failed firmware integrity verification checks. This occurs when the image in the device's storage could not pass checksum, signature, or other validation methods.
FirmwareStorageAccessFailure	Critical	Indicates that a device failed to access its firmware storage.
InRecovery	Critical	Indicates that a device is not operating properly and entered recovery mode to restore firmware or critical data to a state of integrity. This type of condition is a runtime recovery where the device was previously in an operational state and there were no attempts to update the firmware.
InitializationFailure	Critical	Indicates that a device failed to initialize.
ManagementInterfaceError	Critical	Indicates that the management interface for a device encountered an error.
ManagementInterfaceUnresponsive	Critical	Indicates that the management interface for a device is unresponsive.
Missing	Critical	Indicates that the device is unexpectedly missing.
PresenceMismatch	Warning	Indicates that a device presence mismatch was detected.
SecureBootFailure	Critical	Indicates that a device encountered a secure boot failure. This occurs when a device attempts to boot to an image that passed initial verification, but other policies or runtime checks are in place that restrict the device from booting with its current image.
ThermalTrip	Critical	Indicates that a device thermal trip occurred.
Throttled	Warning	Indicates that a device throttling occurred.
WatchdogTimeout	Critical	Indicates that a device encountered a watchdog timeout.

2.5.1 AssemblyDataInaccessible

Example: Unable to access the assembly data for device 'GPU1'.

Indicates that the assembly data, historically 'FRU data', for a device is inaccessible.

- This message shall indicate that the assembly data for a device is inaccessible.

Version Added: v1.0

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Unable to access the assembly data for device '`<1>`'."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.2 AssemblyDataMalformed

Example: Malformed assembly data detected for device 'GPU1'.

Indicates that the assembly data, historically 'FRU data', for a device is malformed, such as due to a checksum error.

- This message shall indicate that the assembly data for a device is malformed, such as due to a checksum error.

Version Added: v1.0

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Malformed assembly data detected for device '`<1>`'."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.3 BootTimeout

Example: Device 'GPU1' encountered a boot timeout.

Indicates that a device encountered a boot timeout. This occurs when a device attempts to boot to an image that passed initial verification and other policy checks, but could not complete its boot sequence.

- This message shall indicate that a device encountered a boot timeout.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available. Verify the correct firmware is installed on the device.

Message and Arguments: "Device '`<1>`' encountered a boot timeout."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.4 Disabled

Example: Device 'GPU1' was disabled or halted.

Indicates that a device was disabled or halted.

- This message shall indicate that a device was disabled or halted.

Version Added: v1.0

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' was disabled or halted."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.5 ErrorDetected

Example: An error was detected on device 'GPU1'.

Indicates that an error was detected on a device.

- This message shall indicate that an error was detected on a device. Services should only use this message if a more specific standard message is not available.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "An error was detected on device '`<1>`'."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.6 FirmwareIntegrityFailure

Example: Device 'GPU1' failed firmware integrity verification checks.

Indicates that a device failed firmware integrity verification checks. This occurs when the image in the device's storage could not pass checksum, signature, or other validation methods.

- This message shall indicate that a device failed firmware integrity verification checks.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available. Verify the correct firmware is installed on the device.

Message and Arguments: "Device '`<1>`' failed firmware integrity verification checks."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.7 FirmwareStorageAccessFailure

Example: Device 'GPU1' failed to access its firmware storage.

Indicates that a device failed to access its firmware storage.

- This message shall indicate that a device failed to access its firmware storage.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' failed to access its firmware storage."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.8 InRecovery

Example: Device 'GPU1' entered recovery mode due to: Firmware corruption.

Indicates that a device is not operating properly and entered recovery mode to restore firmware or critical data to a

state of integrity. This type of condition is a runtime recovery where the device was previously in an operational state and there were no attempts to update the firmware.

- This message shall indicate that a device is not operating properly and entered recovery mode, such as defined by the OCP Secure Firmware Recovery specification, to restore firmware or critical data to a state of integrity.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' entered recovery mode due to: `<2>`."

1. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.
2. *string*: The reason the device entered recovery mode.
   - This argument shall contain a string that describes the reason the device entered recovery mode.

2.5.9 InitializationFailure

Example: Device 'GPU1' initialization failed.

Indicates that a device failed to initialize.

- This message shall indicate that a device failed to complete the initialization sequence during system boot or reset.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the device seating, power delivery, and compatibility.

Message and Arguments: "Device '`<1>`' initialization failed."

1. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.

2.5.10 ManagementInterfaceError

Example: The management interface for device 'GPU1' encountered the error: I2C communication failure.

Indicates that the management interface for a device encountered an error.

- This message shall indicate that the management interface for a device encountered an error.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "The management interface for device '`<1>`' encountered the error: `<2>`."

1. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.
2. *string*: The error encountered on the management interface of the device.
   - This argument shall contain a string that describes error encountered on the management interface of the device.

2.5.11 ManagementInterfaceUnresponsive

Example: The management interface for device 'GPU1' is unresponsive.

Indicates that the management interface for a device is unresponsive.

- This message shall indicate that the management interface for a device is unresponsive.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "The management interface for device '`<1>`' is unresponsive."

1. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.

2.5.12 Missing

Example: Device 'GPU1' could not be detected.

Indicates that the device is unexpectedly missing.

- This message shall be used to indicate the device is unexpectedly missing.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available. Inspect the device to ensure it's present and seated properly.

Message and Arguments: "Device '`<1>`' could not be detected."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.13 PresenceMismatch

Example: Device presence mismatch detected for device 'GPU1'.

Indicates that a device presence mismatch was detected.

- This message shall indicate that the device population does not match the expected or previously recorded configuration. A presence mismatch may indicate an unplanned device replacement, removal, or incorrect installation.

Version Added: v1.0

Severity: Warning

Resolution: Check the device configuration.

Message and Arguments: "Device presence mismatch detected for device '`<1>`'."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.14 SecureBootFailure

Example: Device 'GPU1' encountered a secure boot failure.

Indicates that a device encountered a secure boot failure. This occurs when a device attempts to boot to an image that passed initial verification, but other policies or runtime checks are in place that restrict the device from booting with its current image.

- This message shall indicate that a device encountered a secure boot failure.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available. Verify the correct firmware is installed on the device.

Message and Arguments: "Device '<1>' encountered a secure boot failure."

1. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.

2.5.15 ThermalTrip

Example: A thermal trip occurred on 'GPU1'.

Indicates that a device thermal trip occurred.

- This message shall indicate that the device experienced a thermal trip condition, resulting in the device shutdown.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available. Check thermal sensors, fan health, and cooling device operations.

Message and Arguments: "A thermal trip occurred on '<1>'."

1. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.

2.5.16 Throttled

Example: Throttling occurred on 'GPU1'.

Indicates that a device throttling occurred.

- This message shall indicate that the device experienced a throttling condition, resulting in degraded performance.

Version Added: v1.0

Severity: Warning

Resolution: Check additional diagnostic data if available. Check thermal sensors, fan health, and cooling device operations.

Message and Arguments: "Throttling occurred on '<1>'."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.5.17 WatchdogTimeout

Example: Device 'GPU1' encountered a watchdog timeout.

Indicates that a device encountered a watchdog timeout.

- This message shall indicate that a device encountered a watchdog timeout.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' encountered a watchdog timeout."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.6 Environmental 1.5.1

Version	*v1.5*	*v1.4*	*v1.3*	*v1.2*	*v1.1*	*v1.0*
Release	2026.1	2025.4	2025.3	2025.2	2024.4	2024.1

This registry defines messages related to environmental sensors, heating and cooling equipment, or other environmental conditions.

Message	Severity	Description
CoolantConnectionPairReversed *(v1.5+)*	Critical	Indicates that a supply and return pair of coolant connections are reversed.
CoolantConnectionsCorrected *(v1.5+)*	OK	Indicates that reversed coolant connections were corrected.
CoolantConnectionsReversed *(v1.5+)*	Critical	Indicates that a supply and a return coolant connections are reversed.
CoolingUnitGroupCritical *(v1.4+)*	Critical	Indicates that a cooling unit group has a critical status.
CoolingUnitGroupNormal *(v1.4+)*	OK	Indicates that a cooling unit group has returned to normal operations.
CoolingUnitGroupWarning *(v1.4+)*	Warning	Indicates that a cooling unit group has a warning status.

Message	Severity	Description
DewPointProximityCaution *(v1.5+)*	Warning	Indicates that a temperature reading is below the lower caution dew point proximity threshold.
DewPointProximityCritical *(v1.5+)*	Critical	Indicates that a temperature reading is below the lower critical dew point proximity threshold.
DewPointProximityNoLongerCritical *(v1.5+)*	OK	Indicates that a temperature reading is no longer below the critical dew point proximity threshold, but remains below the caution threshold.
DewPointProximityNormal *(v1.5+)*	OK	Indicates that a dew point proximity temperature reading is now within normal operating range.
FanFailed	Critical	Indicates that a fan has failed.
FanGroupCritical	Critical	Indicates that a fan group has a critical status.
FanGroupNormal	OK	Indicates that a fan group has returned to normal operations.
FanGroupWarning	Warning	Indicates that a fan group has a warning status.
FanInserted	OK	Indicates that a fan was inserted or installed.
FanRemoved	OK	Indicates that a fan was removed.
FanRestored	OK	Indicates that a fan was repaired or restored to normal operation.
FanWarning *(v1.4+)*	Warning	Indicates that a fan has a warning condition.
FilterInserted *(v1.1+)*	OK	Indicates that a filter was inserted or installed.
FilterRemoved *(v1.1+)*	OK	Indicates that a filter was removed.
FilterRequiresService *(v1.1+)*	Warning	Indicates that a filter requires service.
FilterRestored *(v1.1+)*	OK	Indicates that a filter was repaired or restored to normal operation.
FilterSubsystemHealth *(v1.2+)*	OK	Indicates the health of the filter subsystem for the equipment.
FlowRateAboveLowerCriticalThreshold *(v1.1+)*	Warning	Indicates that a flow rate reading is no longer below the lower critical threshold but is still outside of normal operating range.
FlowRateAboveLowerFatalThreshold *(v1.1+)*	Critical	Indicates that a flow rate reading is no longer below the lower fatal threshold but is still outside of normal operating range.
FlowRateAboveUpperCautionThreshold *(v1.1+)*	Warning	Indicates that a flow rate reading is above the upper caution threshold.
FlowRateAboveUpperCriticalThreshold *(v1.1+)*	Critical	Indicates that a flow rate reading is above the upper critical threshold.
FlowRateAboveUpperFatalThreshold *(v1.1+)*	Critical	Indicates that a flow rate reading is above the upper fatal threshold.

Message	Severity	Description
FlowRateBelowLowerCautionThreshold *(v1.1+)*	Warning	Indicates that a flow rate reading is below the lower caution threshold.
FlowRateBelowLowerCriticalThreshold *(v1.1+)*	Critical	Indicates that a flow rate reading is below the lower critical threshold.
FlowRateBelowLowerFatalThreshold *(v1.1+)*	Critical	Indicates that a flow rate reading is below the lower fatal threshold.
FlowRateBelowUpperCriticalThreshold *(v1.1+)*	Warning	Indicates that a flow rate reading is no longer above the upper critical threshold but is still outside of normal operating range.
FlowRateBelowUpperFatalThreshold *(v1.1+)*	Critical	Indicates that a flow rate reading is no longer above the upper fatal threshold but is still outside of normal operating range.
FlowRateCritical *(v1.1+)*	Critical	Indicates that a flow rate reading exceeds an internal critical level.
FlowRateFatal *(v1.1+)*	Critical	Indicates that a flow rate reading exceeds an internal fatal level or flow rate reading is zero (0).
FlowRateNoLongerCritical *(v1.1+)*	Warning	Indicates that a flow rate reading no longer exceeds an internal critical level but still exceeds an internal warning level.
FlowRateNoLongerFatal *(v1.1+)*	Critical	Indicates that a flow rate reading is no longer exceeds an internal fatal level but is still exceeds an internal critical level.
FlowRateNormal *(v1.1+)*	OK	Indicates that a flow rate reading is now within normal operating range.
FlowRateWarning *(v1.1+)*	Warning	Indicates that a flow rate reading exceeds an internal warning level.
FluidLevelAboveLowerCriticalThreshold *(v1.1+)*	Warning	Indicates that a fluid level reading is no longer below the lower critical threshold but is still outside of normal operating range.
FluidLevelAboveLowerFatalThreshold *(v1.1+)*	Critical	Indicates that a fluid level reading is no longer below the lower fatal threshold but is still outside of normal operating range.
FluidLevelAboveUpperCautionThreshold *(v1.1+)*	Warning	Indicates that a fluid level reading is above the upper caution threshold.
FluidLevelAboveUpperCriticalThreshold *(v1.1+)*	Critical	Indicates that a fluid level reading is above the upper critical threshold.
FluidLevelAboveUpperFatalThreshold *(v1.1+)*	Critical	Indicates that a fluid level reading is above the upper fatal threshold.
FluidLevelBelowLowerCautionThreshold *(v1.1+)*	Warning	Indicates that a fluid level reading is below the lower caution threshold.
FluidLevelBelowLowerCriticalThreshold *(v1.1+)*	Critical	Indicates that a fluid level reading is below the lower critical threshold.
FluidLevelBelowLowerFatalThreshold *(v1.1+)*	Critical	Indicates that a fluid level reading is below the lower fatal threshold.

Message	Severity	Description
FluidLevelBelowUpperCriticalThreshold *(v1.1+)*	Warning	Indicates that a fluid level reading is no longer above the upper critical threshold but is still outside of normal operating range.
FluidLevelBelowUpperFatalThreshold *(v1.1+)*	Critical	Indicates that a fluid level reading is no longer above the upper fatal threshold but is still outside of normal operating range.
FluidLevelCritical *(v1.1+)*	Critical	Indicates that a fluid level reading violates an internal critical level.
FluidLevelNoLongerCritical *(v1.1+)*	Warning	Indicates that a fluid level reading no longer violates an internal critical level but still violates an internal warning level.
FluidLevelNormal *(v1.1+)*	OK	Indicates that a fluid level reading is now within normal operating range.
FluidLevelWarning *(v1.1+)*	Warning	Indicates that a fluid level reading violates an internal warning level.
FluidQualityCritical *(v1.1+)*	Critical	Indicates that a fluid quality reading exceeds an internal critical level.
FluidQualityNoLongerCritical *(v1.1+)*	Warning	Indicates that a fluid quality reading no longer exceeds an internal critical level but still exceeds an internal warning level.
FluidQualityNormal *(v1.1+)*	OK	Indicates that a fluid quality reading is now within normal operating range.
FluidQualityWarning *(v1.1+)*	Warning	Indicates that a fluid quality reading exceeds an internal warning level.
HumidityAboveLowerCriticalThreshold	Warning	Indicates that a humidity reading is no longer below the lower critical threshold but is still outside of normal operating range.
HumidityAboveUpperCautionThreshold	Warning	Indicates that a humidity reading is above the upper caution threshold.
HumidityAboveUpperCriticalThreshold	Critical	Indicates that a humidity reading is above the upper critical threshold.
HumidityBelowLowerCautionThreshold	Warning	Indicates that a humidity reading is below the lower caution threshold.
HumidityBelowLowerCriticalThreshold	Critical	Indicates that a humidity reading is below the lower critical threshold.
HumidityBelowUpperCriticalThreshold	Warning	Indicates that a humidity reading is no longer above the upper critical threshold but is still outside of normal operating range.
HumidityNormal	OK	Indicates that a humidity reading is now within normal operating range.
LeakDetectedCritical *(v1.1+)*	Critical	Indicates that a leak detector is in a critical state.
LeakDetectedNormal *(v1.1+)*	OK	Indicates that a leak detector is within normal operating range.
LeakDetectedWarning *(v1.1+)*	Warning	Indicates that a leak detector is in a warning state.
LeakDetectorConnected *(v1.3+)*	OK	Indicates that an external leak detector was connected.
LeakDetectorDisconnected *(v1.3+)*	Warning	Indicates that an external leak detector was disconnected.
LeakDetectorFailure *(v1.3+)*	Warning	Indicates that the service cannot communicate with a leak detector or detected a failure.

Message	Severity	Description
LeakDetectorRestored *(v1.3+)*	OK	Indicates that a leak detector was repaired or communications were restored.
PressureAboveLowerCriticalThreshold *(v1.1+)*	Warning	Indicates that a pressure reading is no longer below the lower critical threshold but is still outside of normal operating range.
PressureAboveLowerFatalThreshold *(v1.1+)*	Critical	Indicates that a pressure reading is no longer below the lower fatal threshold but is still outside of normal operating range.
PressureAboveUpperCautionThreshold *(v1.1+)*	Warning	Indicates that a pressure reading is above the upper caution threshold.
PressureAboveUpperCriticalThreshold *(v1.1+)*	Critical	Indicates that a pressure reading is above the upper critical threshold.
PressureAboveUpperFatalThreshold *(v1.1+)*	Critical	Indicates that a pressure reading is above the upper fatal threshold.
PressureBelowLowerCautionThreshold *(v1.1+)*	Warning	Indicates that a pressure reading is below the lower caution threshold.
PressureBelowLowerCriticalThreshold *(v1.1+)*	Critical	Indicates that a pressure reading is below the lower critical threshold.
PressureBelowLowerFatalThreshold *(v1.1+)*	Critical	Indicates that a pressure reading is below the lower fatal threshold.
PressureBelowUpperCriticalThreshold *(v1.1+)*	Warning	Indicates that a pressure reading is no longer above the upper critical threshold but is still outside of normal operating range.
PressureBelowUpperFatalThreshold *(v1.1+)*	Critical	Indicates that a pressure reading is no longer above the upper fatal threshold but is still outside of normal operating range.
PressureCritical *(v1.1+)*	Critical	Indicates that a pressure reading exceeds an internal critical level.
PressureNoLongerCritical *(v1.1+)*	Warning	Indicates that a pressure reading no longer violates an internal critical level but still violates an internal warning level.
PressureNormal *(v1.1+)*	OK	Indicates that a pressure reading is now within normal operating range.
PressureWarning *(v1.1+)*	Warning	Indicates that a pressure reading exceeds an internal warning level.
PrimaryCoolantConnectorSubsystemHealth *(v1.2+)*	OK	Indicates the health of the primary coolant connector subsystem for the equipment.
PumpFailed *(v1.1+)*	Critical	Indicates that a pump has failed.
PumpInserted *(v1.1+)*	OK	Indicates that a pump was inserted or installed.
PumpRemoved *(v1.1+)*	OK	Indicates that a pump was removed.
PumpRestored *(v1.1+)*	OK	Indicates that a pump was repaired or restored to normal operation.
PumpSubsystemHealth *(v1.2+)*	OK	Indicates the health of the pump subsystem for the equipment.
ReservoirSubsystemHealth *(v1.2+)*	OK	Indicates the health of the reservoir subsystem for the equipment.

Message	Severity	Description
SecondaryCoolantConnectorSubsystemHealth *(v1.2+)*	OK	Indicates the health of the secondary coolant connector subsystem for the equipment.
TemperatureAboveLowerCriticalThreshold	Warning	Indicates that a temperature reading is no longer below the lower critical threshold but is still outside of normal operating range.
TemperatureAboveLowerFatalThreshold	Critical	Indicates that a temperature reading is no longer below the lower fatal threshold but is still outside of normal operating range.
TemperatureAboveUpperCautionThreshold	Warning	Indicates that a temperature reading is above the upper caution threshold.
TemperatureAboveUpperCriticalThreshold	Critical	Indicates that a temperature reading is above the upper critical threshold.
TemperatureAboveUpperFatalThreshold	Critical	Indicates that a temperature reading is above the upper fatal threshold.
TemperatureBelowLowerCautionThreshold	Warning	Indicates that a temperature reading is below the lower caution threshold.
TemperatureBelowLowerCriticalThreshold	Critical	Indicates that a temperature reading is below the lower critical threshold.
TemperatureBelowLowerFatalThreshold	Critical	Indicates that a temperature reading is below the lower fatal threshold.
TemperatureBelowUpperCriticalThreshold	Warning	Indicates that a temperature reading is no longer above the upper critical threshold but is still outside of normal operating range.
TemperatureBelowUpperFatalThreshold	Critical	Indicates that a temperature reading is no longer above the upper fatal threshold but is still outside of normal operating range.
TemperatureCritical	Critical	Indicates that a temperature reading exceeds an internal critical level.
TemperatureNoLongerCritical	Warning	Indicates that a temperature reading no longer exceeds an internal critical level but still exceeds an internal warning level.
TemperatureNormal	OK	Indicates that a temperature reading is now within normal operating range.
TemperatureWarning	Warning	Indicates that a temperature reading exceeds an internal warning level.
ValveClosed *(v1.3+)*	OK	Indicates that a valve was closed.
ValveOpened *(v1.3+)*	OK	Indicates that a valve was opened.
ValvePositionChanged *(v1.3+)*	OK	Indicates that the position of a valve was changed.
ValveUnableToReachSetPoint *(v1.3+)*	Warning	Indicates that a valve is stuck or otherwise unable to reach the desired position.
ValveUnableToReachSetPointPercent *(v1.3+)*	Warning	Indicates that a valve is stuck or otherwise unable to reach the desired percent open position.

2.6.1 CoolantConnectionPairReversed

Example: Coolant connector pair 'Secondary' connections are reversed.

Indicates that a supply and return pair of coolant connections are reversed.

- This message shall indicate that a supply and return coolant connection pair, represented by a `CoolantConnector` resource with a `CoolantConnectorType` valve of `Pair`, are reversed. Services may detect this condition by a negative delta temperature, negative flow rate, or other means.

Version Added: v1.5

Severity: Critical

Resolution: Correct the reversed supply and return coolant connections.

Message and Arguments: "Coolant connector pair '`<1>`' connections are reversed."

1. *string*: The name or identifier of the coolant connector pair.
    - This argument shall contain a string that identifies or describes the location or physical context of the coolant connector pair.

2.6.2 CoolantConnectionsCorrected

Example: Coolant connections 'Secondary' have been corrected.

Indicates that reversed coolant connections were corrected.

- This message shall indicate that a set of supply and return coolant connections, previously detected as connected in reverse, were corrected.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "Coolant connections '`<1>`' were corrected."

1. *string*: The name or identifier of the coolant connectors.
    - This argument shall contain a string that identifies or describes the location or physical context of the affected coolant connections.

2.6.3 CoolantConnectionsReversed

Example: Coolant supply 'Secondary Supply' and return 'Secondary Return' connections are reversed.

Indicates that a supply and a return coolant connections are reversed.

- This message shall indicate that a set of supply and return coolant connections, represented by two or more `CoolantConnector` resources with `CoolantConnectorType` values of `Supply`, `Return`, `SupplyGroupMember` or `ReturnGroupMember` are reversed. Services may detect this condition by a negative delta temperature, negative flow rate, or other means. If both the supply and return `CoolantConnector` resources can be identified, the `OriginOfCondition` shall contain a link to the supply connector.

Version Added: v1.5

Severity: Critical

Resolution: Correct the reversed supply and return coolant connections.

Message and Arguments: "Coolant supply '`<1>`' and return '`<2>`' connections are reversed."

1. *string*: The name or identifier of the supply coolant connector.
   - This argument shall contain a string that identifies or describes the location or physical context of the supply coolant connector.
2. *string*: The name or identifier of the return coolant connector.
   - This argument shall contain a string that identifies or describes the location or physical context of the return coolant connector.

2.6.4 CoolingUnitGroupCritical

Example: Cooling unit group 'Row7' is in a critical state.

Indicates that a cooling unit group has a critical status.

- This message shall indicate that a cooling unit group is operating in a critical state due to a fault or a change in configuration.

Version Added: v1.4

Severity: Critical

Resolution: None.

Message and Arguments: "Cooling unit group '`<1>`' is in a critical state."

1. *string*: The name or identifier of the cooling unit group.
   - This argument shall contain a string that identifies or describes the group of `CoolingUnit` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical

2.6.5 CoolingUnitGroupNormal

Example: Cooling unit group 'Row7' is operating normally.

Indicates that a cooling unit group has returned to normal operations.

- This message shall indicate that a cooling unit group has returned to a normal operating state.

Version Added: v1.4

Severity: OK

Resolution: None.

Message and Arguments: "Cooling unit group ' `<1>` ' is operating normally."

1. *string*: The name or identifier of the cooling unit group.
   - This argument shall contain a string that identifies or describes the group of `CoolingUnit` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.6.6 CoolingUnitGroupWarning

Example: Cooling unit group 'Row7' is in a warning state.

Indicates that a cooling unit group has a warning status.

- This message shall indicate that a cooling unit group is operating with a warning status or in a non-redundant state due to a fault or a change in configuration.

Version Added: v1.4

Severity: Warning

Resolution: None.

Message and Arguments: "Cooling unit group ' `<1>` ' is in a warning state."

1. *string*: The name or identifier of the cooling unit group.
   - This argument shall contain a string that identifies or describes the group of `CoolingUnit` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.6.7 DewPointProximityCaution

Example: Dew point proximity of coolant connector 'Secondary' temperature of 1.7 degrees (C) is below the 2.0 degree caution threshold.

Indicates that a temperature reading is below the lower caution dew point proximity threshold.

- This message shall indicate that a temperature reading violates a lower caution threshold for proximity to the dew point. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.5

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Dew point proximity of `<1>` '`<2>`' temperature of `<3>` degrees (C) is below the `<4>` degree caution threshold."

1. *string*: The resource type or description of the resource.
   - This argument shall contain a string that identifies or describes the resource type of the temperature reading.
2. *string*: The name or identifier of the resource.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
3. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
4. *number*: The dew point proximity caution threshold in degrees Celsius.
   - This argument shall contain the dew point proximity caution threshold in degrees Celsius.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.6.8 DewPointProximityCritical

Example: Dew point proximity of coolant connector 'Secondary' temperature of 0.5 degrees (C) is below the 1.0 degrees critical threshold.

Indicates that a temperature reading is below the lower critical dew point proximity threshold.

- This message shall indicate that a temperature reading violates a lower critical threshold for proximity to the dew point. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.5

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Dew point proximity of `<1>` '`<2>`' temperature of `<3>` degrees (C) is below the `<4>` degrees critical threshold."

1. *string*: The resource type or description of the resource.
   - This argument shall contain a string that identifies or describes the resource type of the temperature reading.
2. *string*: The name or identifier of the resource.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
3. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
4. *number*: The dew point proximity critical threshold in degrees Celsius.
   - This argument shall contain the dew point proximity critical threshold in degrees Celsius.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.6.9 DewPointProximityNoLongerCritical

Example: Dew point proximity of coolant connector 'Secondary' temperature of 1.8 degrees (C) no longer exceeds the critical level.

Indicates that a temperature reading is no longer below the critical dew point proximity threshold, but remains below the caution threshold.

- This message shall indicate that a temperature reading no longer violates the critical threshold for proximity to the dew point, but continues to violate the caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties, but above the value of the `LowerCritical` and `LowerCriticalUser` properties.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "Dew point proximity of `<1>` '`<2>`' temperature of `<3>` degrees (C) no longer exceeds the critical level."

1. *string*: The resource type or description of the resource.
   - This argument shall contain a string that identifies or describes the resource type of the temperature reading.
2. *number*: The name or identifier of the resource.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
3. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.6.10 DewPointProximityNormal

Example: Dew point proximity of coolant connector 'Secondary' temperature of 4.0 degrees (C) is normal.

Indicates that a dew point proximity temperature reading is now within normal operating range.

- This message shall indicate that a dew point proximity temperature reading has returned to its normal operating range.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "Dew point proximity of `<1>` '`<2>`' temperature of `<3>` degrees (C) is normal."

1. *string*: The resource type or description of the resource.
   - This argument shall contain a string that identifies or describes the resource type of the temperature reading.
2. *number*: The name or identifier of the resource.
   - This argument shall contain a string that identifies or describes the location or physical context of

the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.

3. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.SensorReadingNormalRange

2.6.11 FanFailed

Example: Fan 'B3' has failed.

Indicates that a fan has failed.

- This message shall indicate that a fan has failed.

Version Added: v1.0

Severity: Critical

Resolution: Check the fan hardware and replace any faulty component.

Message and Arguments: "Fan '`<1>`' has failed."

1. *string*: The name or identifier of the fan.
   - This argument shall contain a string that identifies or describes the `Fan` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical

2.6.12 FanGroupCritical

Example: Fan group 'CPUs+GPUs' is in a critical state.

Indicates that a fan group has a critical status.

- This message shall indicate that a fan group is operating in a critical state due to a fault or a change in configuration.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "Fan group '`<1>`' is in a critical state."

1. *string*: The name or identifier of the fan group.
   - This argument shall contain a string that identifies or describes the group of `Fan` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical

2.6.13 FanGroupNormal

Example: Fan group 'CPUs+GPUs' is operating normally.

Indicates that a fan group has returned to normal operations.

- This message shall indicate that a fan group has returned to a normal operating state.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Fan group '`<1>`' is operating normally."

1. *string*: The name or identifier of the fan group.
   - This argument shall contain a string that identifies or describes the group of `Fan` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.6.14 FanGroupWarning

Example: Fan group 'CPUs+GPUs' is in a warning state.

Indicates that a fan group has a warning status.

- This message shall indicate that a fan group is operating with a warning status or in a non-redundant state due to a fault or a change in configuration.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "Fan group '`<1>`' is in a warning state."

1. *string*: The name or identifier of the fan group.
   - This argument shall contain a string that identifies or describes the group of `Fan` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.6.15 FanInserted

Example: Fan 'B4' was inserted.

Indicates that a fan was inserted or installed.

- This message shall indicate that a fan was inserted or installed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Fan '`<1>`' was inserted."

1. *string*: The name or identifier of the fan.
   - This argument shall contain a string that identifies or describes the `Fan` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStateChanged

2.6.16 FanRemoved

Example: Fan 'B4' was removed.

Indicates that a fan was removed.

- This message shall indicate that a fan was removed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Fan '`<1>`' was removed."

1. *string*: The name or identifier of the fan.
    - This argument shall contain a string that identifies or describes the `Fan` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStateChanged

2.6.17 FanRestored

Example: Fan 'B4' was restored.

Indicates that a fan was repaired or restored to normal operation.

- This message shall indicate that a fan was repaired, replaced, or otherwise returned to normal operation.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Fan '`<1>`' was restored."

1. *string*: The name or identifier of the fan.
    - This argument shall contain a string that identifies or describes the `Fan` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.6.18 FanWarning

Example: Fan 'B3' has a warning condition.

Indicates that a fan has a warning condition.

- This message shall indicate that a fan has a warning condition but continues to operate, potentially in a degraded state.

Version Added: v1.4

Severity: Warning

Resolution: Check the fan hardware and replace any faulty component.

Message and Arguments: "Fan '`<1>`' has a warning condition."

1. *string*: The name or identifier of the fan.
    - This argument shall contain a string that identifies or describes the `Fan` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.6.19 FilterInserted

Example: Filter 'Secondary' was inserted.

Indicates that a filter was inserted or installed.

- This message shall indicate that a filter was inserted or installed.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Filter ' `<1>` ' was inserted."

1. *string*: The name or identifier of the filter.
    - This argument shall contain a string that identifies or describes the `Filter` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStateChanged

2.6.20 FilterRemoved

Example: Filter 'Secondary' was removed.

Indicates that a filter was removed.

- This message shall indicate that a filter was removed.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Filter ' `<1>` ' was removed."

1. *string*: The name or identifier of the filter.

    - This argument shall contain a string that identifies or describes the `Filter` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStateChanged

2.6.21 FilterRequiresService

Example: Filter 'Secondary' requires service.

Indicates that a filter requires service.

- This message shall indicate that a filter requires service. Examples include a dirty filter or a filter reaching the end of its life.

Version Added: v1.1

Severity: Warning

Resolution: Replace the filter or filter media.

Message and Arguments: "Filter '`<1>`' requires service."

1. *string*: The name or identifier of the filter.
    - This argument shall contain a string that identifies or describes the `Filter` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.6.22 FilterRestored

Example: Filter 'Secondary' was restored.

Indicates that a filter was repaired or restored to normal operation.

- This message shall indicate that a filter was repaired, replaced, or otherwise returned to normal operation.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Filter '`<1>`' was restored."

1. *string*: The name or identifier of the filter.

- This argument shall contain a string that identifies or describes the `Filter` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.6.23 FilterSubsystemHealth

Example: The filter subsystem health is OK.

Indicates the health of the filter subsystem for the equipment.

- This message shall indicate the health of the filter subsystem for the equipment.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "The filter subsystem health is `<1>`."

1. *string*: The health of the filter subsystem for the equipment.
   - This argument shall contain the health of the filter subsystem for the equipment. This argument shall contain a value defined by the `Health` enumeration in the `Resource` schema.

2.6.24 FlowRateAboveLowerCriticalThreshold

Example: Flow rate 'Secondary TCS' reading of 18.9 L/min is now above the 15 lower critical threshold but remains outside of normal range.

Indicates that a flow rate reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a flow rate reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the flow rate violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the flow rate reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
    - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerCriticalThreshold

2.6.25 FlowRateAboveLowerFatalThreshold

Example: Flow rate 'Secondary TCS' reading of 14.8 L/min is now above the 12 lower fatal threshold but remains outside of normal range.

Indicates that a flow rate reading is no longer below the lower fatal threshold but is still outside of normal operating range.

- This message shall indicate that a flow rate reading, previously below the lower fatal threshold, no longer violates the lower fatal threshold. However, the flow rate violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is now above the `<3>` lower fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the flow rate reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
    - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerFatalThreshold

2.6.26 FlowRateAboveUpperCautionThreshold

Example: Flow rate 'Facility Water System' reading of 62.4 L/min is above the 60 upper caution threshold.

Indicates that a flow rate reading is above the upper caution threshold.

- This message shall indicate that a flow rate reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties. Otherwise, the message shall indicate that the reading violates an internal level or threshold.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate ' `<1>` ' reading of `<2>` L/min is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
   - If the reading originated from a `Sensor` resource, this argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource. Otherwise, this argument shall contain an internal level or threshold value.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.6.27 FlowRateAboveUpperCriticalThreshold

Example: Flow rate 'Facility Water System' reading of 68.1 L/min is above the 65 upper critical threshold.

Indicates that a flow rate reading is above the upper critical threshold.

- This message shall indicate that a flow rate reading violates an upper critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.6.28 FlowRateAboveUpperFatalThreshold

Example: Flow rate 'Facility Water System' reading of 75.8 L/min is above the 70 upper fatal threshold.

Indicates that a flow rate reading is above the upper fatal threshold.

- This message shall indicate that a flow rate reading violates an upper fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperFatal` property.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is above the `<3>` upper fatal threshold."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperFatalThreshold

2.6.29 FlowRateBelowLowerCautionThreshold

Example: Flow rate 'Secondary TCS' reading of 18.5 L/min is below the 20 lower caution threshold.

Indicates that a flow rate reading is below the lower caution threshold.

- This message shall indicate that a flow rate reading violates a lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate ' `<1>` ' reading of `<2>` L/min is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.6.30 FlowRateBelowLowerCriticalThreshold

Example: Flow rate 'Secondary TCS' reading of 14.7 L/min is below the 15 lower critical threshold.

Indicates that a flow rate reading is below the lower critical threshold.

- This message shall indicate that a flow rate reading violates a lower critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.6.31 FlowRateBelowLowerFatalThreshold

Example: Flow rate 'Secondary TCS' reading of 7.0 L/min is below the 12 lower fatal threshold.

Indicates that a flow rate reading is below the lower fatal threshold.

- This message shall indicate that a flow rate reading violates a lower fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerFatal` property.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is below the `<3>` lower fatal threshold."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerFatalThreshold

2.6.32 FlowRateBelowUpperCriticalThreshold

Example: Flow rate 'Facility Water System' reading of 64.3 L/min is now below the 65 upper critical threshold but remains outside of normal range.

Indicates that a flow rate reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a flow rate reading, previously above the upper critical threshold, no longer violates the upper critical threshold. However, the flow rate violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.6.33 FlowRateBelowUpperFatalThreshold

Example: Flow rate 'Facility Water System' reading of 68.8 L/min is now below the 70 upper fatal threshold but remains outside of normal range.

Indicates that a flow rate reading is no longer above the upper fatal threshold but is still outside of normal operating range.

- This message shall indicate that a flow rate reading, previously above the upper fatal threshold, no longer violates the upper fatal threshold. However, the flow rate violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is now below the `<3>` upper fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperFatalThreshold

2.6.34 FlowRateCritical

Example: Flow rate 'Facility Water System' reading of 67.4 L/min exceeds the critical level.

Indicates that a flow rate reading exceeds an internal critical level.

- This message shall indicate that a flow rate reading violates an internal critical level. This message shall be used for flow rate properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The flow rate properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min exceeds the critical level."

1. *string*: The name or identifier of the flow rate reading.

    - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
    - This argument shall contain the flow rate in litres per minute.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.6.35 FlowRateFatal

Example: Flow rate 'Facility Water System' reading of 73.1 L/min exceeds the fatal level.

Indicates that a flow rate reading exceeds an internal fatal level or flow rate reading is zero (0).

- This message shall indicate that a flow rate reading violates an internal fatal level. This message shall be used for flow rate properties or `Sensor` resources that implement boolean trip points for a fatal limit, such as from hardware mechanisms with no threshold value. The flow rate properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min exceeds the fatal level."

1. *string*: The name or identifier of the flow rate reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
    - This argument shall contain the flow rate in litres per minute.

2.6.36 FlowRateNoLongerCritical

Example: Flow rate 'Facility Water System' reading of 67.1 L/min no longer exceeds the critical level.

Indicates that a flow rate reading no longer exceeds an internal critical level but still exceeds an internal warning level.

- This message shall indicate that a flow rate reading no longer violates an internal critical level but still violates an internal warning level. This message shall be used for flow rate properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The flow rate properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min no longer exceeds the critical level."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.6.37 FlowRateNoLongerFatal

Example: Flow rate 'Facility Water System' reading of 68.1 L/min no longer exceeds the fatal level.

Indicates that a flow rate reading is no longer exceeds an internal fatal level but is still exceeds an internal critical level.

- This message shall indicate that a flow rate reading no longer violates an internal fatal level but still violates an internal critical level. This message shall be used for flow rate properties or `Sensor` resources that implement boolean trip points for a fatal limit, such as from hardware mechanisms with no threshold value. The flow rate properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min no longer exceeds the fatal level."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.

2.6.38 FlowRateNormal

Example: Flow rate 'Secondary TCS' reading of 42.1 L/min is within normal operating range.

Indicates that a flow rate reading is now within normal operating range.

- This message shall indicate that a flow rate reading has returned to its normal operating range.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min is within normal operating range."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.SensorReadingNormalRange

2.6.39 FlowRateWarning

Example: Flow rate 'Facility Water System' reading of 62.8 L/min exceeds the warning level.

Indicates that a flow rate reading exceeds an internal warning level.

- This message shall indicate that a flow rate reading violates an internal warning level. This message shall be used for flow rate properties or `Sensor` resources that implement boolean trip points for a caution limit, such as from hardware mechanisms with no threshold value. The flow rate properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Flow rate '`<1>`' reading of `<2>` L/min exceeds the warning level."

1. *string*: The name or identifier of the flow rate reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the flow rate reading.
2. *number*: The flow rate in litres per minute.
   - This argument shall contain the flow rate in litres per minute.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.6.40 FluidLevelAboveLowerCriticalThreshold

Example: Fluid level 'Fill Reservoir' reading of 15 percent is now above the 10 lower critical threshold but remains outside of normal range.

Indicates that a fluid level reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a fluid level reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the fluid level violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
   - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingAboveLowerCriticalThreshold

2.6.41 FluidLevelAboveLowerFatalThreshold

Example: Fluid level 'Fill Reservoir' reading of 5 percent is now above the 1 lower fatal threshold but remains outside of normal range.

Indicates that a fluid level reading is no longer below the lower fatal threshold but is still outside of normal operating range.

- This message shall indicate that a fluid level reading, previously below the lower fatal threshold, no longer violates the lower fatal threshold. However, the fluid level violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is now above the `<3>` lower fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
   - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerFatalThreshold

2.6.42 FluidLevelAboveUpperCautionThreshold

Example: Fluid level 'Overflow Reservoir' reading of 81 percent is above the 80 upper caution threshold.

Indicates that a fluid level reading is above the upper caution threshold.

- This message shall indicate that a fluid level reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties. Otherwise, the message shall indicate that the reading violates an internal level or threshold.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
   - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
   - If the reading originated from a `Sensor` resource, this argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource. Otherwise, this argument shall contain an internal level or threshold value.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.6.43 FluidLevelAboveUpperCriticalThreshold

Example: Fluid level 'Overflow Reservoir' reading of 97 percent is above the 95 upper critical threshold.

Indicates that a fluid level reading is above the upper critical threshold.

- This message shall indicate that a fluid level reading violates an upper critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
   - This argument shall contain the fluid level in percent.

3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.6.44 FluidLevelAboveUpperFatalThreshold

Example: Fluid level 'Overflow Reservoir' reading of 100 percent is above the 99.5 upper fatal threshold.

Indicates that a fluid level reading is above the upper fatal threshold.

- This message shall indicate that a fluid level reading violates an upper fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperFatal` property.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level ' `<1>` ' reading of `<2>` percent is above the `<3>` upper fatal threshold."

1. *string*: The name or identifier of the fluid level reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
    - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperFatalThreshold

2.6.45 FluidLevelBelowLowerCautionThreshold

Example: Fluid level 'Fill Reservoir' reading of 19 percent is below the 20 lower caution threshold.

Indicates that a fluid level reading is below the lower caution threshold.

- This message shall indicate that a fluid level reading violates a lower caution threshold. If the reading originates

from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
   - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.6.46 FluidLevelBelowLowerCriticalThreshold

Example: Fluid level 'Fill Reservoir' reading of 7 percent is below the 10 lower critical threshold.

Indicates that a fluid level reading is below the lower critical threshold.

- This message shall indicate that a fluid level reading violates a lower critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.

2. *number*: The fluid level in percent.
    - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.6.47 FluidLevelBelowLowerFatalThreshold

Example: Fluid level 'Fill Reservoir' reading of 0.5 percent is below the 1 lower fatal threshold.

Indicates that a fluid level reading is below the lower fatal threshold.

- This message shall indicate that a fluid level reading violates a lower fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerFatal` property.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is below the `<3>` lower fatal threshold."

1. *string*: The name or identifier of the fluid level reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
    - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerFatalThreshold

2.6.48 FluidLevelBelowUpperCriticalThreshold

Example: Fluid level 'Overflow Reservoir' reading of 91.5 percent is now below the 95 upper critical threshold but remains outside of normal range.

Indicates that a fluid level reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a fluid level reading, previously above the upper critical threshold, no longer violates the upper critical threshold. However, the fluid level violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
   - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.6.49 FluidLevelBelowUpperFatalThreshold

Example: Fluid level 'Overflow Reservoir' reading of 97 percent is now below the 100 upper fatal threshold but remains outside of normal range.

Indicates that a fluid level reading is no longer above the upper fatal threshold but is still outside of normal operating range.

- This message shall indicate that a fluid level reading, previously above the upper fatal threshold, no longer violates the upper fatal threshold. However, the fluid level violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading of `<2>` percent is now below the `<3>` upper fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.
2. *number*: The fluid level in percent.
   - This argument shall contain the fluid level in percent.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingBelowUpperFatalThreshold

2.6.50 FluidLevelCritical

Example: Fluid level 'Overflow Reservoir' reading violates the critical level.

Indicates that a fluid level reading violates an internal critical level.

- This message shall indicate that a fluid level reading violates an internal critical level. This message shall be used for fluid level properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The fluid level properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading violates the critical level."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.6.51 FluidLevelNoLongerCritical

Example: Fluid level 'Overflow Reservoir' reading no longer violates the critical level.

Indicates that a fluid level reading no longer violates an internal critical level but still violates an internal warning level.

- This message shall indicate that a fluid level reading no longer violates an internal critical level but still violates an internal warning level. This message shall be used for fluid level properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The fluid level properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading no longer violates the critical level."

1. *string*: The name or identifier of the fluid level reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.6.52 FluidLevelNormal

Example: Fluid level 'Overflow Reservoir' reading is within normal operating range.

Indicates that a fluid level reading is now within normal operating range.

- This message shall indicate that a fluid level reading has returned to its normal operating range.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Fluid level '`<1>`' reading is within normal operating range."

1. *string*: The name or identifier of the fluid level reading.

    - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.

> **Note:** This message is a more specific version of the following messages: SensorEvent.SensorReadingNormalRange

2.6.53 FluidLevelWarning

Example: Fluid level 'Overflow Reservoir' reading violates the warning level.

Indicates that a fluid level reading violates an internal warning level.

- This message shall indicate that a fluid level reading violates an internal warning level. This message shall be used for fluid level properties or `Sensor` resources that implement boolean trip points for a caution limit, such as from hardware mechanisms with no threshold value. The fluid level properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid level '`<1>`' reading violates the warning level."

1. *string*: The name or identifier of the fluid level reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the fluid level reading.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.6.54 FluidQualityCritical

Example: Fluid quality 'Secondary TCS' reading exceeds the critical level.

Indicates that a fluid quality reading exceeds an internal critical level.

- This message shall indicate that a fluid quality reading violates an internal critical level. This message shall be used for fluid quality properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The fluid quality properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid quality '`<1>`' reading exceeds the critical level."

1. *string*: The name or identifier of the fluid quality reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid quality reading.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.6.55 FluidQualityNoLongerCritical

Example: Fluid quality 'Secondary TCS' reading no longer exceeds the critical level.

Indicates that a fluid quality reading no longer exceeds an internal critical level but still exceeds an internal warning level.

- This message shall indicate that a fluid quality reading no longer violates an internal critical level but still violates an internal warning level. This message shall be used for fluid quality properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The fluid quality properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid quality '`<1>`' reading no longer exceeds the critical level."

1. *string*: The name or identifier of the fluid quality reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid quality reading.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingNoLongerCritical

2.6.56 FluidQualityNormal

Example: Fluid quality 'Secondary TCS' reading is within normal operating range.

Indicates that a fluid quality reading is now within normal operating range.

- This message shall indicate that a fluid quality reading has returned to its normal operating range.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Fluid quality '`<1>`' reading is within normal operating range."

1. *string*: The name or identifier of the fluid quality reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid quality reading.

> **Note:** This message is a more specific version of the following messages: SensorEvent.SensorReadingNormalRange

2.6.57 FluidQualityWarning

Example: Fluid quality 'Secondary TCS' reading exceeds the warning level.

Indicates that a fluid quality reading exceeds an internal warning level.

- This message shall indicate that a fluid quality reading violates an internal warning level. This message shall be used for fluid quality properties or `Sensor` resources that implement boolean trip points for a caution limit, such as from hardware mechanisms with no threshold value. The fluid quality properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Fluid quality '`<1>`' reading exceeds the warning level."

1. *string*: The name or identifier of the fluid quality reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the fluid quality reading.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.6.58 HumidityAboveLowerCriticalThreshold

Example: Humidity 'Room Ambient' reading of 26.3 percent is now above the 25 lower critical threshold but remains outside of normal range.

Indicates that a humidity reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a humidity reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the humidity violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Humidity '`<1>`' reading of `<2>` percent is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the humidity reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the humidity reading.
2. *number*: The percent humidity.
   - This argument shall contain the humidity as a percentage.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerCriticalThreshold

2.6.59 HumidityAboveUpperCautionThreshold

Example: Humidity 'Room Ambient' reading of 57 percent is above the 55 upper caution threshold.

Indicates that a humidity reading is above the upper caution threshold.

- This message shall indicate that a humidity reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Humidity '`<1>`' reading of `<2>` percent is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the humidity reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the humidity reading.
2. *number*: The humidity in percent units.
   - This argument shall contain the humidity as a percentage.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.6.60 HumidityAboveUpperCriticalThreshold

Example: Humidity 'Room Ambient' reading of 62.1 percent is above the 60 upper critical threshold.

Indicates that a humidity reading is above the upper critical threshold.

- This message shall indicate that a humidity reading violates an upper critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Humidity '`<1>`' reading of `<2>` percent is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the humidity reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the humidity reading.
2. *number*: The humidity in percent units.
   - This argument shall contain the humidity as a percentage.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.6.61 HumidityBelowLowerCautionThreshold

Example: Humidity 'Room Ambient' reading of 27.8 percent is below the 30 lower caution threshold.

Indicates that a humidity reading is below the lower caution threshold.

- This message shall indicate that a humidity reading violates an lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Humidity '`<1>`' reading of `<2>` percent is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the humidity reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the humidity reading.
2. *number*: The humidity in percent units.
   - This argument shall contain the humidity as a percentage.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.6.62 HumidityBelowLowerCriticalThreshold

Example: Humidity 'Room Ambient' reading of 24.8 percent is below the 25 lower critical threshold.

Indicates that a humidity reading is below the lower critical threshold.

- This message shall indicate that a humidity reading violates a lower critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Humidity '`<1>`' reading of `<2>` percent is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the humidity reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the humidity reading.
2. *number*: The humidity in percent units.
   - This argument shall contain the humidity as a percentage.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.6.63 HumidityBelowUpperCriticalThreshold

Example: Humidity 'Room Ambient' reading of 59.2 percent is now below the 60 upper critical threshold but remains outside of normal range.

Indicates that a humidity reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a humidity reading, previously above the critical threshold, no longer violates the critical threshold. However, the humidity violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Humidity '`<1>`' reading of `<2>` percent is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the humidity reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the humidity reading.
2. *number*: The percent humidity.
   - This argument shall contain the humidity as a percentage.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or

`UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.6.64 HumidityNormal

Example: Humidity 'Room Ambient' reading of 51.5 percent is within normal operating range.

Indicates that a humidity reading is now within normal operating range.

- This message shall indicate that a humidity reading has returned to its normal operating range.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Humidity '`<1>`' reading of `<2>` percent is within normal operating range."

1. *string*: The name or identifier of the humidity reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the humidity reading.
2. *number*: The percent humidity.
   - This argument shall contain the humidity as a percentage.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.SensorReadingNormalRange

2.6.65 LeakDetectedCritical

Example: Leak detector 'Rack #42 Spot' reports a critical level leak.

Indicates that a leak detector is in a critical state.

- This message shall indicate a leak detector is in a critical state.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Leak detector '`<1>`' reports a critical level leak."

1. *string*: The name or identifier of the leak detector.
    - This argument shall contain a string that identifies or describes the location or physical context of the leak detector.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.6.66 LeakDetectedNormal

Example: Leak detector 'Rack #42 Spot' has returned to normal.

Indicates that a leak detector is within normal operating range.

- This message shall indicate that a leak detector returned to its normal operating state.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Leak detector '`<1>`' has returned to normal."

1. *string*: The name or identifier of the leak detector.
    - This argument shall contain a string that identifies or describes the location or physical context of the leak detector.

> **Note:** This message is a more specific version of the following messages: SensorEvent.SensorReadingNormalRange

2.6.67 LeakDetectedWarning

Example: Leak detector 'Rack #19 Rope' reports a warning level leak.

Indicates that a leak detector is in a warning state.

- This message shall indicate a leak detector is in a warning state.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Leak detector '`<1>`' reports a warning level leak."

1. *string*: The name or identifier of the leak detector.
    - This argument shall contain a string that identifies or describes the location or physical context of the leak detector.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.6.68 LeakDetectorConnected

Example: Leak detector 'Rack #19 Rope' was connected.

Indicates that an external leak detector was connected.

- This message shall indicate that an external leak detector was connected.

Version Added: v1.3

Severity: OK

Resolution: None.

Message and Arguments: "Leak detector '`<1>`' was connected."

1. *string*: The name or identifier of the leak detector.
    - This argument shall contain a string that identifies or describes the `LeakDetector` resource.

2.6.69 LeakDetectorDisconnected

Example: Leak detector 'Rack #19 Rope' was disconnected.

Indicates that an external leak detector was disconnected.

- This message shall indicate that an external leak detector was disconnected.

Version Added: v1.3

Severity: Warning

Resolution: Check the leak detector connection or re-connect the leak detector.

Message and Arguments: "Leak detector '`<1>`' was disconnected."

1. *string*: The name or identifier of the leak detector.
    - This argument shall contain a string that identifies or describes the `LeakDetector` resource.

2.6.70 LeakDetectorFailure

Example: Leak detector 'Rack #19 Rope' failed.

Indicates that the service cannot communicate with a leak detector or detected a failure.

- This message shall indicate that the service is unable to communicate with a leak detector or otherwise detected a fault condition with the leak detector.

Version Added: v1.3

Severity: Warning

Resolution: Check the leak detector hardware or connection.

Message and Arguments: "Leak detector '`<1>`' failed."

1. *string*: The name or identifier of the leak detector.
   - This argument shall contain a string that identifies or describes the `LeakDetector` resource.

2.6.71 LeakDetectorRestored

Example: Leak detector 'Rack #42 Spot' was restored.

Indicates that a leak detector was repaired or communications were restored.

- This message shall indicate that the service is able to communicate with a leak detector that had previously failed, or the leak detector was repaired or replaced.

Version Added: v1.3

Severity: OK

Resolution: None.

Message and Arguments: "Leak detector '`<1>`' was restored."

1. *string*: The name or identifier of the leak detector.
   - This argument shall contain a string that identifies or describes the `LeakDetector` resource.

2.6.72 PressureAboveLowerCriticalThreshold

Example: Pressure 'Secondary TCS' reading of 671 kPa is now above the 650 lower critical threshold but remains outside of normal range.

Indicates that a pressure reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a pressure reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the pressure violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerCriticalThreshold

2.6.73 PressureAboveLowerFatalThreshold

Example: Pressure 'Secondary TCS' reading of 528 kPa is now above the 450 lower fatal threshold but remains outside of normal range.

Indicates that a pressure reading is no longer below the lower fatal threshold but is still outside of normal operating range.

- This message shall indicate that a pressure reading, previously below the lower fatal threshold, no longer

violates the lower fatal threshold. However, the pressure violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is now above the `<3>` lower fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerFatalThreshold

2.6.74 PressureAboveUpperCautionThreshold

Example: Pressure 'Facility Water System' reading of 928 kPa is above the 900 upper caution threshold.

Indicates that a pressure reading is above the upper caution threshold.

- This message shall indicate that a pressure reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties. Otherwise, the message shall indicate that the reading violates an internal level or threshold.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
   - If the reading originated from a `Sensor` resource, this argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource. Otherwise, this argument shall contain an internal level or threshold value.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.6.75 PressureAboveUpperCriticalThreshold

Example: Pressure 'Facility Water System Delta Pressure' reading of 76 kPa is above the 50 upper critical threshold.

Indicates that a pressure reading is above the upper critical threshold.

- This message shall indicate that a pressure reading violates an upper critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.6.76 PressureAboveUpperFatalThreshold

Example: Pressure 'Facility Water System' reading of 1138 kPa is above the 1100 upper fatal threshold.

Indicates that a pressure reading is above the upper fatal threshold.

- This message shall indicate that a pressure reading violates an upper fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperFatal` property.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure ' `<1>` ' reading of `<2>` kPa is above the `<3>` upper fatal threshold."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperFatalThreshold

2.6.77 PressureBelowLowerCautionThreshold

Example: Pressure 'Secondary TCS' reading of 689 kPa is below the 700 lower caution threshold.

Indicates that a pressure reading is below the lower caution threshold.

- This message shall indicate that a pressure reading violates a lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.6.78 PressureBelowLowerCriticalThreshold

Example: Pressure 'Secondary TCS' reading of 541 kPa is below the 600 lower critical threshold.

Indicates that a pressure reading is below the lower critical threshold.

- This message shall indicate that a pressure reading violates a lower critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.

    - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.6.79 PressureBelowLowerFatalThreshold

Example: Pressure 'Secondary TCS' reading of 389 kPa is below the 450 lower fatal threshold.

Indicates that a pressure reading is below the lower fatal threshold.

- This message shall indicate that a pressure reading violates a lower fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerFatal` property.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is below the `<3>` lower fatal threshold."

1. *string*: The name or identifier of the pressure reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
    - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerFatalThreshold

2.6.80 PressureBelowUpperCriticalThreshold

Example: Pressure 'Facility Water System' reading of 937 kPa is now below the 1000 upper critical threshold but remains outside of normal range.

Indicates that a pressure reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a pressure reading, previously above the upper critical threshold, no longer violates the upper critical threshold. However, the pressure violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.6.81 PressureBelowUpperFatalThreshold

Example: Pressure 'Facility Water System' reading of 1019 kPa is now below the 1100 upper fatal threshold but remains outside of normal range.

Indicates that a pressure reading is no longer above the upper fatal threshold but is still outside of normal operating range.

- This message shall indicate that a pressure reading, previously above the upper fatal threshold, no longer violates the upper fatal threshold. However, the pressure violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is now below the `<3>` upper fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingBelowUpperFatalThreshold

2.6.82 PressureCritical

Example: Pressure 'Facility Water System' reading of 1026 kPa exceeds the critical level.

Indicates that a pressure reading exceeds an internal critical level.

- This message shall indicate that a pressure reading violates an internal critical level. This message shall be used for pressure properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The pressure properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa exceeds the critical level."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.6.83 PressureNoLongerCritical

Example: Pressure 'Facility Water System' reading of 987 kPa no longer violates the critical level.

Indicates that a pressure reading no longer violates an internal critical level but still violates an internal warning level.

- This message shall indicate that a pressure reading no longer violates an internal critical level but still violates an internal warning level. This message shall be used for pressure properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The pressure properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa no longer violates the critical level."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.6.84 PressureNormal

Example: Pressure 'Facility Water System' reading of 781 kPa is within normal operating range.

Indicates that a pressure reading is now within normal operating range.

- This message shall indicate that a pressure reading has returned to its normal operating range.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa is within normal operating range."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.

> **Note:** This message is a more specific version of the following messages: SensorEvent.SensorReadingNormalRange

2.6.85 PressureWarning

Example: Pressure 'Facility Water System' reading of 972 kPa exceeds the warning level.

Indicates that a pressure reading exceeds an internal warning level.

- This message shall indicate that a pressure reading violates an internal warning level. This message shall be used for pressure properties or `Sensor` resources that implement boolean trip points for a caution limit, such as from hardware mechanisms with no threshold value. The pressure properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.1

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Pressure '`<1>`' reading of `<2>` kPa exceeds the warning level."

1. *string*: The name or identifier of the pressure reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the pressure reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The pressure in kilopascals.
   - This argument shall contain the pressure in kilopascals.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.6.86 PrimaryCoolantConnectorSubsystemHealth

Example: The primary coolant connector subsystem health is OK.

Indicates the health of the primary coolant connector subsystem for the equipment.

- This message shall indicate the health of the primary coolant connector subsystem for the equipment.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "The primary coolant connector subsystem health is `<1>`."

1. *string*: The health of the primary coolant connector subsystem for the equipment.
   - This argument shall contain the health of the primary coolant connector subsystem for the equipment. This argument shall contain a value defined by the `Health` enumeration in the `Resource` schema.

2.6.87 PumpFailed

Example: Pump 'Output #3' has failed.

Indicates that a pump has failed.

- This message shall indicate that a pump has failed.

Version Added: v1.1

Severity: Critical

Resolution: Check the pump hardware and replace any faulty component.

Message and Arguments: "Pump ' `<1>` ' has failed."

1. *string*: The name or identifier of the pump.
   - This argument shall contain a string that identifies or describes the `Pump` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical

2.6.88 PumpInserted

Example: Pump 'Output #3' was inserted.

Indicates that a pump was inserted or installed.

- This message shall indicate that a pump was inserted or installed.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Pump '`<1>`' was inserted."

1. *string*: The name or identifier of the pump.
   - This argument shall contain a string that identifies or describes the `Pump` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStateChanged

2.6.89 PumpRemoved

Example: Pump 'Output #3' was removed.

Indicates that a pump was removed.

- This message shall indicate that a pump was removed.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Pump '`<1>`' was removed."

1. *string*: The name or identifier of the pump.
   - This argument shall contain a string that identifies or describes the `Pump` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStateChanged

2.6.90 PumpRestored

Example: Pump 'Output #3' was restored.

Indicates that a pump was repaired or restored to normal operation.

- This message shall indicate that a pump was repaired, replaced, or otherwise returned to normal operation.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Pump '`<1>`' was restored."

1. *string*: The name or identifier of the pump.
   - This argument shall contain a string that identifies or describes the `Pump` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.6.91 PumpSubsystemHealth

Example: The pump subsystem health is Critical.

Indicates the health of the pump subsystem for the equipment.

- This message shall indicate the health of the pump subsystem for the equipment.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "The pump subsystem health is `<1>`."

1. *string*: The health of the pump subsystem for the equipment.
   - This argument shall contain the health of the pump subsystem for the equipment. This argument shall contain a value defined by the `Health` enumeration in the `Resource` schema.

2.6.92 ReservoirSubsystemHealth

Example: The reservoir subsystem health is Warning.

Indicates the health of the reservoir subsystem for the equipment.

- This message shall indicate the health of the reservoir subsystem for the equipment.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "The reservoir subsystem health is `<1>`."

1. *string*: The health of the reservoir subsystem for the equipment.
   - This argument shall contain the health of the reservoir subsystem for the equipment. This argument shall contain a value defined by the `Health` enumeration in the `Resource` schema.

2.6.93 SecondaryCoolantConnectorSubsystemHealth

Example: The secondary coolant connector subsystem health is Critical.

Indicates the health of the secondary coolant connector subsystem for the equipment.

- This message shall indicate the health of the secondary coolant connector subsystem for the equipment.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "The secondary coolant connector subsystem health is `<1>`."

1. *string*: The health of the secondary coolant connector subsystem for the equipment.
   - This argument shall contain the health of the secondary coolant connector subsystem for the equipment. This argument shall contain a value defined by the `Health` enumeration in the `Resource` schema.

2.6.94 TemperatureAboveLowerCriticalThreshold

Example: Temperature 'Facility Water Supply' reading of 16.1 degrees (C) is now above the 15 lower critical threshold but remains outside of normal range.

Indicates that a temperature reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a temperature reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the temperature violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerCriticalThreshold

2.6.95 TemperatureAboveLowerFatalThreshold

Example: Temperature 'Facility Water Supply' reading of 10.7 degrees (C) is now above the 10 lower fatal threshold but remains outside of normal range.

Indicates that a temperature reading is no longer below the lower fatal threshold but is still outside of normal operating range.

- This message shall indicate that a temperature reading, previously below the lower fatal threshold, no longer violates the lower fatal threshold. However, the temperature violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is now above the `<3>` lower fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.

- This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerFatalThreshold

2.6.96 TemperatureAboveUpperCautionThreshold

Example: Temperature 'CPU1' reading of 43.8 degrees (C) is above the 42 upper caution threshold.

Indicates that a temperature reading is above the upper caution threshold.

- This message shall indicate that a temperature reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties. Otherwise, the message shall indicate that the reading violates an internal level or threshold.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature ' `<1>` ' reading of `<2>` degrees (C) is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - If the reading originated from a `Sensor` resource, this argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource. Otherwise, this argument shall contain an internal level or threshold value.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.6.97 TemperatureAboveUpperCriticalThreshold

Example: Temperature 'CPU1' reading of 47.4 degrees (C) is above the 45 upper critical threshold.

Indicates that a temperature reading is above the upper critical threshold.

- This message shall indicate that a temperature reading violates an upper critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature ' `<1>` ' reading of `<2>` degrees (C) is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.6.98 TemperatureAboveUpperFatalThreshold

Example: Temperature 'CPU1' reading of 50.5 degrees (C) is above the 50 upper fatal threshold.

Indicates that a temperature reading is above the upper fatal threshold.

- This message shall indicate that a temperature reading violates an upper fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperFatal` property.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is above the `<3>` upper fatal threshold."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperFatalThreshold

2.6.99 TemperatureBelowLowerCautionThreshold

Example: Temperature 'Facility Water Supply' reading of 14.7 degrees (C) is below the 15 lower caution threshold.

Indicates that a temperature reading is below the lower caution threshold.

- This message shall indicate that a temperature reading violates a lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.6.100 TemperatureBelowLowerCriticalThreshold

Example: Temperature 'Facility Water Supply' reading of 11.8 degrees (C) is below the lower critical threshold.

Indicates that a temperature reading is below the lower critical threshold.

- This message shall indicate that a temperature reading violates a lower critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.6.101 TemperatureBelowLowerFatalThreshold

Example: Temperature 'Facility Water Supply' reading of 9.6 degrees (C) is below the 10 lower fatal threshold.

Indicates that a temperature reading is below the lower fatal threshold.

- This message shall indicate that a temperature reading violates a lower fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerFatal` property.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is below the `<3>` lower fatal threshold."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerFatalThreshold

2.6.102 TemperatureBelowUpperCriticalThreshold

Example: Temperature 'CPU1' reading of 44.1 degrees (C) is now below the 45 upper critical threshold but remains outside of normal range.

Indicates that a temperature reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a temperature reading, previously above the upper critical threshold, no longer violates the upper critical threshold. However, the temperature violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of

the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.

2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.6.103 TemperatureBelowUpperFatalThreshold

Example: Temperature 'CPU1' reading of 49.5 degrees (C) is now below the 50 upper fatal threshold but remains outside of normal range.

Indicates that a temperature reading is no longer above the upper fatal threshold but is still outside of normal operating range.

- This message shall indicate that a temperature reading, previously above the upper fatal threshold, no longer violates the upper fatal threshold. However, the temperature violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is now below the `<3>` upper fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperFatalThreshold

2.6.104 TemperatureCritical

Example: Temperature 'CPU1' reading of 49.2 degrees (C) exceeds the critical level.

Indicates that a temperature reading exceeds an internal critical level.

- This message shall indicate that a temperature reading violates an internal critical level. This message shall be used for temperature properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The temperature properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature ' `<1>` ' reading of `<2>` degrees (C) exceeds the critical level."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.6.105 TemperatureNoLongerCritical

Example: Temperature 'CPU1' reading of 41.8 degrees (C) no longer exceeds the critical level.

Indicates that a temperature reading no longer exceeds an internal critical level but still exceeds an internal warning level.

- This message shall indicate that a temperature reading no longer violates an internal critical level but still violates an internal warning level. This message shall be used for temperature properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The temperature properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) no longer exceeds the critical level."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.6.106 TemperatureNormal

Example: Temperature 'CPU1' reading of 39.5 degrees (C) is within normal operating range.

Indicates that a temperature reading is now within normal operating range.

- This message shall indicate that a temperature reading has returned to its normal operating range.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) is within normal operating range."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.SensorReadingNormalRange

2.6.107 TemperatureWarning

Example: Temperature 'CPU1' reading of 43.8 degrees (C) exceeds the warning level.

Indicates that a temperature reading exceeds an internal warning level.

- This message shall indicate that a temperature reading violates an internal warning level. This message shall be used for temperature properties or `Sensor` resources that implement boolean trip points for a caution limit, such as from hardware mechanisms with no threshold value. The temperature properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Temperature '`<1>`' reading of `<2>` degrees (C) exceeds the warning level."

1. *string*: The name or identifier of the temperature reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the temperature reading. For `Sensor` readings with `ReadingBasis` of `Delta`, this identifier should include an indication that the value is differential, such as the word `delta`.
2. *number*: The temperature in degrees Celsius.
   - This argument shall contain the temperature in degrees Celsius.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.6.108 ValveClosed

Example: Valve 'Secondary Shutoff' was closed.

Indicates that a valve was closed.

- This message shall indicate that a valve, previously in an opened state, was closed. This message may be used for any valve type.

Version Added: v1.3

Severity: OK

Resolution: None.

Message and Arguments: "Valve '`<1>`' was closed."

1. *string*: The name or identifier of the valve.
    - This argument shall contain a string that identifies or describes the location or physical context of the valve.

2.6.109 ValveOpened

Example: Valve 'Secondary Shutoff' was opened.

Indicates that a valve was opened.

- This message shall indicate that a valve, previously in a closed state, was opened. This message may be used for any valve type. The `ValvePositionPercent` property in the related resource may provide more detailed information about the position of the valve.

Version Added: v1.3

Severity: OK

Resolution: None.

Message and Arguments: "Valve '`<1>`' was opened."

1. *string*: The name or identifier of the valve.
    - This argument shall contain a string that identifies or describes the location or physical context of the valve.

2.6.110 ValvePositionChanged

Example: Valve 'Secondary Flow' was changed to 50 percent open.

Indicates that the position of a valve was changed.

- This message shall indicate that the position of the valve, in percent open units, was changed. This message may be used for valves that support variable percent open positions. This message is intended to indicate a valve position change due to a updated set point, not for changes due to the normal operation of a control loop. Services should use the `ValveClosed` for a change to a value of zero percent.

Version Added: v1.3

Severity: OK

Resolution: None.

Message and Arguments: "Valve '`<1>`' was changed to `<1>` percent open."

1. *string*: The name or identifier of the valve.
    - This argument shall contain a string that identifies or describes the location or physical context of the valve.
2. *number*: The new position of the valve, in percent open units.
    - This argument shall contain a number that indicates the new valve position, in percent open units.

2.6.111 ValveUnableToReachSetPoint

Example: Valve 'Secondary Flow' is unable to reach the desired set point.

Indicates that a valve is stuck or otherwise unable to reach the desired position.

- This message shall indicate that the valve is unable to attain the desired position set point, due to an issue with the valve or an obstruction. Services should use the `ValveUnableToReachSetPointPercent` message if the valve contains percent-based position and set point information.

Version Added: v1.3

Severity: Warning

Resolution: Cycle the valve to a different position. Inspect the valve, control unit, motor, or other components.

Message and Arguments: "Valve '`<1>`' is unable to reach the desired set point."

1. *string*: The name or identifier of the valve.
    - This argument shall contain a string that identifies or describes the location or physical context of the valve.

2.6.112 ValveUnableToReachSetPointPercent

Example: Valve 'Secondary' is unable to reach the desired 80 percent open set point and is currently at 58 percent open.

Indicates that a valve is stuck or otherwise unable to reach the desired percent open position.

- This message shall indicate that the valve is unable to attain the desired position set point, in percent open units, due to an issue with the valve or an obstruction. Services should use the `ValveUnableToReachSetPoint` message if the valve does not contain percent-based position and set point position information.

Version Added: v1.3

Severity: Warning

Resolution: Cycle the valve to a different position. Inspect the valve, control unit, motor, or other components.

Message and Arguments: "Valve '<1>' is unable to reach the desired <2> percent open set point and is currently at <3> percent open."

1. *string*: The name or identifier of the valve.
   - This argument shall contain a string that identifies or describes the location or physical context of the valve.
2. *number*: The desired valve set point, in percent open units.
   - This argument shall contain the desired set point of the valve, in percent open units.
3. *number*: The current valve position, in percent open units.
   - This argument shall contain the current position of the valve, in percent open units.

2.7 EthernetFabric 1.1.1

Version	*v1.1*	*v1.0*
Release	2026.1	2020.4

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 were 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.

2.7.1 LLDPInterfaceDisabled

Example: LLDP was disabled on switch 'Switch1' port 'Port1'.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Check that LLDP is enabled on device endpoints.

Message and Arguments: "LLDP was disabled on switch ' `<1>` ' port ' `<2>` '."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource on which LLDP was disabled.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource on which LLDP was disabled.

2.7.2 LLDPInterfaceEnabled

Example: LLDP was enabled on switch 'Switch1' port 'Port1'.

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

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "LLDP was enabled on switch ' `<1>` ' port ' `<2>` '."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource on which LLDP was enabled.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource on which LLDP was enabled.

2.7.3 MLAGInterfaceDegraded

Example: MLAG interface 'MLAG1' is degraded on switch 'Switch1'.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "MLAG interface '`<1>`' is degraded on switch '`<2>`'."

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` resource for which an MLAG interface is degraded.

2.7.4 MLAGInterfaceDown

Example: The MLAG interface 'MLAG1' on switch 'Switch1' is down.

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

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

Version Added: v1.0

Severity: Warning

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

Message and Arguments: "The MLAG interface '`<1>`' on switch '`<2>`' is down."

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

2.7.5 MLAGInterfacesUp

Example: All MLAG interfaces were established for MLAG ID 'MLAG1'.

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

- This message shall indicate all multi-chassis link aggregation group (MLAG) interfaces were established for one MLAG domain ID.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "All MLAG interfaces were established for MLAG ID '`<1>`'."

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

2.7.6 MLAGPeerDown

Example: MLAG peer switch 'Switch2' with MLAG ID 'MLAG1' is down.

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

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

Version Added: v1.0

Severity: Warning

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

Message and Arguments: "MLAG peer switch '`<1>`' with MLAG ID '`<2>`' is down."

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

2.7.7 MLAGPeerUp

Example: MLAG peer switch 'Switch2' with MLAG ID 'MLAG1' is up.

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

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "MLAG peer switch ' `<1>` ' with MLAG ID ' `<2>` ' is up."

1. *string*: The `Id` of the peer switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource that represents the peer switch for which MLAG interface is up.
2. *string*: The MLAG domain ID.
   - This argument shall contain the value of the MLAG domain ID that is established.

2.7.8 RoutingFailureThresholdExceeded

Example: Switch 'Switch1' has encountered 100 routing errors in the last 60 minutes.

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

- This message shall 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: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch ' `<1>` ' has encountered `<2>` routing errors in the last `<3>` minutes."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource 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 were encountered.
    - This argument shall contain the number of minutes over which routing errors were encountered.

2.8 EventService 1.0.0

Version	*v1.0*
Release	2026.2

This registry defines messages for the lifecycle of Redfish event subscriptions and the health of event delivery.

Message	Severity	Description
EventDeliveryFailed	Warning	Indicates that an attempt to deliver an event to a subscriber failed.
EventDeliverySucceededWithRetries	Warning	Indicates that an attempt to deliver an event to a subscriber succeeded, but required retries.
EventSubscriptionCreated	OK	Indicates that an event subscription was created and is now active.
EventSubscriptionRemoved	Warning	Indicates that an event subscription was removed and is no longer active.
EventSubscriptionResumed	OK	Indicates that an event subscription was resumed.
EventSubscriptionSuspended	Warning	Indicates that an event subscription was suspended.

2.8.1 EventDeliveryFailed

Example: Event delivery to subscriber 'contoso-listener' failed with protocol 'Redfish' and destination 'https://contoso.com/events'.

Indicates that an attempt to deliver an event to a subscriber failed.

- This message shall indicate that an attempt to deliver an event to a subscriber failed. This message shall only be used after all retries are performed. This message shall not be used for subscriptions configured to retry forever.

Version Added: v1.0

Severity: Warning

Resolution: Check the subscriber endpoint availability and network connectivity. Check the configuration of the event subscription.

Message and Arguments: "Event delivery to subscriber '`<1>`' failed with protocol '`<2>`' and destination '`<3>`'."

1. *string*: The identifier of the event subscription.
   - This argument shall contain a string that identifies the event subscription for which the event delivery failed.
2. *string*: The protocol of the event subscription.
   - This argument shall contain the protocol from the `Protocol` property of the event subscription.
3. *string*: The destination URI of the event subscription.
   - This argument shall contain the destination URI from the `Destination` property of the event subscription.

2.8.2 EventDeliverySucceededWithRetries

Example: Event delivery to subscriber 'contoso-listener' succeeded with protocol 'Redfish' and destination 'https://contoso.com/events' after 3 retries.

Indicates that an attempt to deliver an event to a subscriber succeeded, but required retries.

- This message shall indicate that an attempt to deliver an event to a subscriber succeeded, but required retries. This message shall only be used when retries were performed.

Version Added: v1.0

Severity: Warning

Resolution: Check the subscriber endpoint availability and network connectivity. Check the configuration of the event subscription.

Message and Arguments: "Event delivery to subscriber '`<1>`' succeeded with protocol '`<2>`' and destination '`<3>`' after `<4>` retries."

1. *string*: The identifier of the event subscription.
   - This argument shall contain a string that identifies the event subscription for which the event delivery failed.
2. *string*: The protocol of the event subscription.
   - This argument shall contain the protocol from the `Protocol` property of the event subscription.
3. *string*: The destination URI of the event subscription.
   - This argument shall contain the destination URI from the `Destination` property of the event subscription.
4. *number*: The number of retries that were performed.
   - This argument shall contain the number of retries that were performed.

2.8.3 EventSubscriptionCreated

Example: Event subscription 'contoso-listener' was created successfully with protocol 'Redfish' and destination 'https://contoso.com/events'.

Indicates that an event subscription was created and is now active.

- This message shall indicate that an event subscription was created and is now active.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Event subscription '`<1>`' was created successfully with protocol '`<2>`' and destination '`<3>`'."

1. *string*: The identifier of the event subscription.
   - This argument shall contain a string that identifies the event subscription that was removed successfully.
2. *string*: The protocol of the event subscription.
   - This argument shall contain the protocol from the `Protocol` property of the event subscription.
3. *string*: The destination URI of the event subscription.
   - This argument shall contain the destination URI from the `Destination` property of the event subscription.

2.8.4 EventSubscriptionRemoved

Example: Event subscription 'contoso-listener' was removed successfully with protocol 'Redfish' and destination 'https://contoso.com/events'.

Indicates that an event subscription was removed and is no longer active.

- This message shall indicate that an event subscription was removed and is no longer active.

Version Added: v1.0

Severity: Warning

Resolution: Crated a new event subscription if continued event delivery is required.

Message and Arguments: "Event subscription '`<1>`' was removed successfully with protocol '`<2>`' and destination '`<3>`'."

1. *string*: The identifier of the event subscription.
   - This argument shall contain a string that identifies the event subscription that was removed successfully.
2. *string*: The protocol of the event subscription.
   - This argument shall contain the protocol from the `Protocol` property of the event subscription.
3. *string*: The destination URI of the event subscription.
   - This argument shall contain the destination URI from the `Destination` property of the event subscription.

2.8.5 EventSubscriptionResumed

Example: Event subscription 'contoso-listener' was resumed with protocol 'Redfish' and destination 'https://contoso.com/events'.

Indicates that an event subscription was resumed.

- This message shall indicate that an event subscription was resumed. Services may resume an event subscription due an administrator performing the `EventDestnation.ResumeSubscription` action.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Event subscription '`<1>`' was resumed with protocol '`<2>`' and destination '`<3>`'."

1. *string*: The identifier of the event subscription.
   - This argument shall contain a string that identifies the event subscription that was resumed.
2. *string*: The protocol of the event subscription.
   - This argument shall contain the protocol from the `Protocol` property of the event subscription.
3. *string*: The destination URI of the event subscription.
   - This argument shall contain the destination URI from the `Destination` property of the event subscription.

2.8.6 EventSubscriptionSuspended

Example: Event subscription 'contoso-listener' was suspended with protocol 'Redfish' and destination 'https://contoso.com/events'.

Indicates that an event subscription was suspended.

- This message shall indicate that an event subscription was suspended. Services may suspend an event subscription due to repeated delivery failures or an administrator performing the `EventDestnation.SuspendSubscription` action.

Version Added: v1.0

Severity: Warning

Resolution: Determine the reason for suspension. If due to delivery failures, check subscriber endpoint availability and check the subscription configuration. If due to an administrative action, resume the subscription when necessary.

Message and Arguments: "Event subscription ' `<1>` ' was suspended with protocol ' `<2>` ' and destination ' `<3>` '."

1. *string*: The identifier of the event subscription.
   - This argument shall contain a string that identifies the event subscription that was suspended.
2. *string*: The protocol of the event subscription.
   - This argument shall contain the protocol from the `Protocol` property of the event subscription.
3. *string*: The destination URI of the event subscription.
   - This argument shall contain the destination URI from the `Destination` property of the event subscription.

2.9 Fabric 1.1.0

Version	*v1.1*	*v1.0*
Release	2026.1	2020.4

This registry defines messages for generic fabrics.

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

Message	Severity	Description
CableRemoved	Warning	Indicates that a cable was removed from a switch's port.
ConnectionCreated	OK	Indicates that a connection was created.
ConnectionModified	OK	Indicates that a connection was modified.
ConnectionRemoved	OK	Indicates that a connection was removed.
DegradedDownstreamLinkEstablished	Warning	Indicates that a switch's downstream connection is established but is in a degraded state.
DegradedInterswitchLinkEstablished	Warning	Indicates that a switch's interswitch connection is established but is in a degraded state.
DegradedUpstreamLinkEstablished	Warning	Indicates that a switch's upstream connection is established but is 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 was created or discovered.
EndpointModified	OK	Indicates that an endpoint was modified.
EndpointRemoved	OK	Indicates that an endpoint was 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 was exceeded.
MediaControllerAdded	OK	Indicates that a media controller was added.
MediaControllerModified	OK	Indicates that a media controller was modified.
MediaControllerRemoved	OK	Indicates that a media controller was removed.
PortAutomaticallyDisabled	OK	Indicates that a switch's port was automatically disabled.
PortAutomaticallyEnabled	OK	Indicates that a switch's port was 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 was manually disabled.

Message	Severity	Description
PortManuallyEnabled	OK	Indicates that a switch's port was 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 was created.
ZoneModified	OK	Indicates that a zone was modified.
ZoneRemoved	OK	Indicates that a zone was removed.

2.9.1 AddressPoolCreated

Example: Address pool 'AddressPool1' was created in fabric 'Fabric1'.

Indicates that an address pool was created.

- This message shall indicate that a resource of type `AddressPool` was created.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Address pool ' `<1>` ' was created in fabric ' `<2>` '."

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

2.9.2 AddressPoolModified

Example: Address pool 'AddressPool1' in fabric 'Fabric1' was modified.

Indicates that an address pool was modified.

- This message shall indicate that a resource of type `AddressPool` was modified.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Address pool ' `<1>` ' in fabric ' `<2>` ' was modified."

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

2.9.3 AddressPoolRemoved

Example: Address pool 'AddressPool1' was removed from fabric 'Fabric1'.

Indicates that an address pool was removed.

- This message shall indicate that a resource of type `AddressPool` was removed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Address pool ' `<1>` ' was removed from fabric ' `<2>` '."

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

    - This argument shall contain the value of the `Id` property of the `Fabric` resource in which the address pool was removed.

2.9.4 CableFailed

Example: The cable in switch 'Switch1' port 'Port1' has failed.

Indicates that a cable has failed.

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "The cable in switch ' `<1>` ' port ' `<2>` ' has failed."

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

2.9.5 CableInserted

Example: A cable was inserted into switch 'Switch1' port 'Port1'.

Indicates that a cable was inserted into a switch's port.

- This message shall indicate that a cable was inserted into a port on a switch.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A cable was inserted into switch ' `<1>` ' port ' `<2>` '."

1. *string*: The `Id` of the switch.

    - This argument shall contain the value of the `Id` property of the `Switch` resource 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` resource whose cable was inserted.

2.9.6 CableOK

Example: The cable in switch 'Switch1' port 'Port1' has returned to working condition.

Indicates that a cable has returned to working condition.

- This message shall indicate that a cable was restored to working condition.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The cable in switch '`<1>`' port '`<2>`' 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` resource 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` resource with the cable that returned to a functional state.

2.9.7 CableRemoved

Example: A cable was removed from switch 'Switch1' port 'Port1'.

Indicates that a cable was removed from a switch's port.

- This message shall indicate that a cable was removed from a port on a switch.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "A cable was removed from switch '`<1>`' port '`<2>`'."

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

2.9.8 ConnectionCreated

Example: Connection 'Connection1' was created in fabric 'Fabric1'.

Indicates that a connection was created.

- This message shall indicate that a resource of type `Connection` was created.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Connection '`<1>`' was created in fabric '`<2>`'."

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

2.9.9 ConnectionModified

Example: Connection 'Connection1' in fabric 'Fabric1' was modified.

Indicates that a connection was modified.

- This message shall indicate that a resource of type `Connection` was modified.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Connection '`<1>`' in fabric '`<2>`' was modified."

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

2.9.10 ConnectionRemoved

Example: Connection 'Connection1' was removed from fabric 'Fabric1'.

Indicates that a connection was removed.

- This message shall indicate that a resource of type `Connection` was removed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Connection '`<1>`' was removed from fabric '`<2>`'."

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

2.9.11 DegradedDownstreamLinkEstablished

Example: Switch 'Switch1' downstream link is established on port 'Port1', but is running in a degraded state.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '`<1>`' downstream link is established on port '`<2>`', 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` resource.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that represents the downstream port.

2.9.12 DegradedInterswitchLinkEstablished

Example: Switch 'Switch1' interswitch link is established on port 'Port1', but is running in a degraded state.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '`<1>`' interswitch link is established on port '`<2>`', 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` resource.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that represents the interswitch port.

2.9.13 DegradedUpstreamLinkEstablished

Example: Switch 'Switch1' upstream link is established on port 'Port1', but is running in a degraded state.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch ' `<1>` ' upstream link is established on port ' `<2>` ', 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` resource.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that represents the upstream port.

2.9.14 DownstreamLinkDropped

Example: Switch 'Switch1' downstream link has gone down on port 'Port1'.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch ' `<1>` ' downstream link has gone down on port ' `<2>` '."

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

2.9.15 DownstreamLinkEstablished

Example: Switch 'Switch1' downstream link is established on port 'Port1'.

Indicates that a switch's downstream connection is established.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch '`<1>`' downstream link is established on port '`<2>`'."

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

2.9.16 DownstreamLinkFlapDetected

Example: Switch 'Switch1' downstream link on port 'Port1' was established and dropped 5 times in the last 10 minutes.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '`<1>`' downstream link on port '`<2>`' was established and dropped `<3>` times in the last `<4>` minutes."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that represents 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 was detected.

2.9.17 EndpointCreated

Example: Endpoint 'Endpoint1' was created in fabric 'Fabric1'.

Indicates that an endpoint was created or discovered.

- This message shall indicate that a resource of type `Endpoint` was created or discovered.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Endpoint '`<1>`' was created in fabric '`<2>`'."

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

2.9.18 EndpointModified

Example: Endpoint 'Endpoint1' in fabric 'Fabric1' was modified.

Indicates that an endpoint was modified.

- This message shall indicate that a resource of type `Endpoint` was modified.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Endpoint '`<1>`' in fabric '`<2>`' was modified."

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

    - This argument shall contain the value of the `Id` property of the `Fabric` resource in which the endpoint was modified.

2.9.19 EndpointRemoved

Example: Endpoint 'Endpoint1' was removed from fabric 'Fabric1'.

Indicates that an endpoint was removed.

- This message shall indicate that a resource of type `Endpoint` was removed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Endpoint '`<1>`' was removed from fabric '`<2>`'."

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

2.9.20 InterswitchLinkDropped

Example: Switch 'Switch1' interswitch link has gone down on port 'Port1'.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '`<1>`' interswitch link has gone down on port '`<2>`'."

1. *string*: The `Id` of the switch.
    - This argument shall contain the value of the `Id` property of the `Switch` resource.

2. *string*: The `Id` of the port.
    - This argument shall contain the value of the `Id` property of the `Port` resource that represents the interswitch port.

2.9.21 InterswitchLinkEstablished

Example: Switch 'Switch1' interswitch link is established on port 'Port1'.

Indicates that a switch's interswitch connection is established.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch '`<1>`' interswitch link is established on port '`<2>`'."

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

2.9.22 InterswitchLinkFlapDetected

Example: Switch 'Switch1' interswitch link on port 'Port1' was established and dropped 5 times in the last 10 minutes.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '`<1>`' interswitch link on port '`<2>`' was established and dropped `<3>` times in the last `<4>` minutes."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that represents 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 was detected.

2.9.23 MaxFrameSizeExceeded

Example: MTU size on switch 'Switch1' port 'Port1' is set to 1500. One or more packets with a larger size were dropped.

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

- This message shall indicate that packets were 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: v1.0

Severity: Warning

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

Message and Arguments: "MTU size on switch ' `<1>` ' port ' `<2>` ' is set to `<3>`. One or more packets with a larger size were dropped."

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

2.9.24 MediaControllerAdded

Example: Media controller 'MediaController1' was added to chassis 'Chassis1'.

Indicates that a media controller was added.

- This message shall indicate that a resource of type `MediaController` was added.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Media controller '`<1>`' was added to chassis '`<2>`'."

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

2.9.25 MediaControllerModified

Example: Media controller 'MediaController1' in chassis 'Chassis1' was modified.

Indicates that a media controller was modified.

- This message shall indicate that a resource of type `MediaController` was modified.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Media controller '`<1>`' in chassis '`<2>`' was modified."

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

2.9.26 MediaControllerRemoved

Example: Media controller 'MediaController1' was removed from chassis 'Chassis1'.

Indicates that a media controller was removed.

- This message shall indicate that a resource of type `MediaController` was removed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Media controller '`<1>`' was removed from chassis '`<2>`'."

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

2.9.27 PortAutomaticallyDisabled

Example: Switch 'Switch1' port 'Port1' was automatically disabled.

Indicates that a switch's port was automatically disabled.

- This message shall indicate that a switch's port was automatically disabled, such as by the service.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch '`<1>`' port '`<2>`' was automatically disabled."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource with the disabled port.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that was disabled.

2.9.28 PortAutomaticallyEnabled

Example: Switch 'Switch1' port 'Port1' was automatically enabled.

Indicates that a switch's port was automatically enabled.

- This message shall indicate that a switch's port was automatically enabled, such as by the service.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch ' `<1>` ' port ' `<2>` ' was automatically enabled."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource with the enabled port.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that was enabled.

2.9.29 PortDegraded

Example: Switch 'Switch1' port 'Port1' is in a degraded state.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch ' `<1>` ' port ' `<2>` ' 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` resource with the degraded port.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that has entered a degraded state.

2.9.30 PortFailed

Example: Switch 'Switch1' port 'Port1' has failed and is inoperative.

Indicates that a switch's port has become inoperative.

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

Version Added: v1.0

Severity: Critical

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch ' `<1>` ' port ' `<2>` ' 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` resource with the failed port.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that has failed.

2.9.31 PortManuallyDisabled

Example: Switch 'Switch1' port 'Port1' was manually disabled.

Indicates that a switch's port was manually disabled.

- This message shall indicate that a switch's port was manually disabled, such as by a network administrator.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch ' `<1>` ' port ' `<2>` ' was manually disabled."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource with the disabled port.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that was disabled.

2.9.32 PortManuallyEnabled

Example: Switch 'Switch1' port 'Port1' was manually enabled.

Indicates that a switch's port was manually enabled.

- This message shall indicate that a switch's port was manually enabled, such as by a network administrator.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch ' `<1>` ' port ' `<2>` ' was manually enabled."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource with the enabled port.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that was enabled.

2.9.33 PortOK

Example: Switch 'Switch1' port 'Port1' has returned to a functional state.

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

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch ' `<1>` ' port ' `<2>` ' 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` resource with the functional port.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that has returned to a functional state.

2.9.34 SwitchDegraded

Example: Switch 'Switch1' is in a degraded state.

Indicates that a switch is in a degraded state.

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch ' `<1>` ' 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` resource that has entered a degraded state.

2.9.35 SwitchFailed

Example: Switch 'Switch1' has failed and is inoperative.

Indicates that a switch has become inoperative.

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

Version Added: v1.0

Severity: Critical

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch ' `<1>` ' 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.

2.9.36 SwitchOK

Example: Switch 'Switch1' has returned to a functional state.

Indicates that a switch has returned to a functional state.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch '`<1>`' 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` resource that has returned to a functional state.

2.9.37 UpstreamLinkDropped

Example: Switch 'Switch1' upstream link has gone down on port 'Port1'.

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

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '`<1>`' upstream link has gone down on port '`<2>`'."

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

2.9.38 UpstreamLinkEstablished

Example: Switch 'Switch1' upstream link is established on port 'Port1'.

Indicates that a switch's upstream connection is established.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Switch '`<1>`' upstream link is established on port '`<2>`'."

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

2.9.39 UpstreamLinkFlapDetected

Example: Switch 'Switch1' upstream link on port 'Port1' was established and dropped 5 times in the last 10 minutes.

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

- This message shall indicate that a switch's upstream connection was repeatedly established and dropped. This message shall not be repeated more than once in any 24-hour period.

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "Switch '`<1>`' upstream link on port '`<2>`' was established and dropped `<3>` times in the last `<4>` minutes."

1. *string*: The `Id` of the switch.
   - This argument shall contain the value of the `Id` property of the `Switch` resource.
2. *string*: The `Id` of the port.
   - This argument shall contain the value of the `Id` property of the `Port` resource that represents 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 was detected.

2.9.40 ZoneCreated

Example: Zone 'Zone1' was created in fabric 'Fabric1'.

Indicates that a zone was created.

- This message shall indicate that a resource of type `Zone` was created.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Zone '`<1>`' was created in fabric '`<2>`'."

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

2.9.41 ZoneModified

Example: Zone 'Zone1' in fabric 'Fabric1' was modified.

Indicates that a zone was modified.

- This message shall indicate that a resource of type `Zone` was modified.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Zone '`<1>`' in fabric '`<2>`' was modified."

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

2.9.42 ZoneRemoved

Example: Zone 'Zone1' was removed from fabric 'Fabric1'.

Indicates that a zone was removed.

- This message shall indicate that a resource of type `Zone` was removed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Zone ' `<1>` ' was removed from fabric ' `<2>` '."

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

2.10 HeartbeatEvent 1.2.0

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.1	2024.4	2021.2

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 indicate that the Redfish service is functional.
RedfishServiceShuttingDown *(v1.1+)*	OK	Indicates that the Redfish service is shutting down and will no longer accept requests.
RedfishServiceStarted *(v1.1+)*	OK	Indicates that the Redfish service started and is ready to accept requests.

2.10.1 RedfishServiceFunctional

Example: Redfish service is functional.

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

- This message shall 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 `HeartbeatIntervalMinutes` property.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Redfish service is functional."

2.10.2 RedfishServiceShuttingDown

Example: Redfish service is shutting down.

Indicates that the Redfish service is shutting down and will no longer accept requests.

- This message shall indicate that the Redfish service is shutting down and is no longer able to service requests.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Redfish service is shutting down."

2.10.3 RedfishServiceStarted

Example: Redfish service started.

Indicates that the Redfish service started and is ready to accept requests.

- This message shall indicate that the Redfish service started and is fully functional.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Redfish service started."

2.11 JobEvent 1.2.1

Version	*v1.2*	*v1.1*	*v1.0*

Release	2026.1	2025.2	2020.3

This registry defines the messages for job related events.

Message	Severity	Description
JobCancelled	Warning	A job was 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 was removed.
JobResumed	OK	A job has resumed.
JobStarted	OK	A job has started.
JobSuspended	Warning	A job was suspended.
JobValidateException *(v1.1+)*	Warning	A job validation has completed with warnings or errors.
JobValidatedOK *(v1.1+)*	OK	A job validation has completed.
JobValidationBypassed *(v1.1+)*	OK	A job has been queued without validation.
JobValidationStarted *(v1.1+)*	OK	A job validation has been started.

2.11.1 JobCancelled

Example: The job with Id 'Job1' was cancelled.

A job was cancelled.

- This message shall indicate that a job was cancelled and the `JobState` property transitioned to `Cancelled`.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' was cancelled."

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

2.11.2 JobCompletedException

Example: The job with Id 'Job1' has completed with warnings or errors.

A job has completed with warnings or errors.

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

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' 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.

2.11.3 JobCompletedOK

Example: The job with Id 'Job1' has completed.

A job has completed.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' 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.

2.11.4 JobProgressChanged

Example: The job with Id 'Job1' has changed to progress 50 percent complete.

A job has changed progress.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' has changed to progress `<2>` 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.

2.11.5 JobRemoved

Example: The job with Id 'Job1' was removed.

A job was removed.

- This message shall indicate that a job was removed.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' was removed."

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

2.11.6 JobResumed

Example: The job with Id 'Job1' has resumed.

A job has resumed.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' 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.

2.11.7 JobStarted

Example: The job with Id 'Job1' has started.

A job has started.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' has started."

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

2.11.8 JobSuspended

Example: The job with Id 'Job1' was suspended.

A job was suspended.

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

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' was suspended."

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

2.11.9 JobValidateException

Example: The job with Id 'Job1' has completed validation with warnings or errors.

A job validation has completed with warnings or errors.

- This message shall indicate that a job self-validation has completed and the `JobState` property transitioned to `Invalid`.

Version Added: v1.1

Severity: Warning

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' has completed validation 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 self-validation with warnings or errors.

2.11.10 JobValidatedOK

Example: The job with Id 'Job1' has been validated.

A job validation has completed.

- This message shall indicate that a job self-validation has completed and the `JobState` property transitioned to

`Queued`.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' has been validated."

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

2.11.11 JobValidationBypassed

Example: The job with Id 'Job1' has been queued without validation.

A job has been queued without validation.

- This message shall indicate that a job is in the execution queue and the `JobState` property transitioned to `Queued` without validation.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' has been queued without validation."

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

2.11.12 JobValidationStarted

Example: The job with Id 'Job1' is under validation.

A job validation has been started.

- This message shall indicate that a job has started self-validation as the `JobState` property transitions to `Validation`.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The job with Id '`<1>`' is under validation."

1. *string*: The `Id` of the job that is being validated.
   - This argument shall contain the value of the `Id` property of the `Job` resource that is being validated.

2.12 License 1.1.0

Version	*v1.1*	*v1.0*
Release	2026.1	2021.3

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 was 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 was 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.

2.12.1 DaysBeforeExpiration

Example: The license 'License1' will expire in 30 days.

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

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The license '`<1>`' will expire in `<2>` 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.

2.12.2 Expired

Example: The license 'License1' has expired.

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

- This message shall indicate that a license has expired and its functionality was disabled.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The license '`<1>`' has expired."

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

2.12.3 GracePeriod

Example: The license 'License1' has expired, 7 day grace period before licensed functionality is disabled.

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

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

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The license '<1>' has expired, <2> 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.

2.12.4 InstallFailed

Example: Failed to install the license. Reason: Invalid license format.

Indicates that the service failed to install the license.

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

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "Failed to install the license. Reason: <1>."

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

2.12.5 InvalidLicense

Example: The content of the license was not recognized, is corrupted, or is invalid.

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

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

Version Added: v1.0

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."

2.12.6 LicenseInstalled

Example: The license 'License1' was installed.

Indicates that a license was installed.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The license '`<1>`' was installed."

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

2.12.7 NotApplicableToTarget

Example: The license is not applicable to the target.

Indicates that the license is not applicable to the target.

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

Version Added: v1.0

Severity: Critical

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

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

2.12.8 TargetsRequired

Example: The license requires targets to be specified.

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

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

Version Added: v1.0

Severity: Critical

Resolution: Add AuthorizedDevices to Links and resubmit the request.

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

2.13 LogService 1.2.1

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.1	2025.4	2020.3

This registry defines the messages for log service related events.

Message	Severity	Description
DiagnosticDataBlocked *(v1.2+)*	Warning	Indicates that the service attempted to collect diagnostic data automatically based on its own policies, such as in response to hardware errors, but restrictions are preventing it from starting. For example, a setting in a device that blocks sideband access to collecting diagnostic data.
DiagnosticDataCollected	OK	Indicates that diagnostic data was collected due to a client invoking the `CollectDiagnosticData` action.
DiagnosticDataCollectionCompleted *(v1.2+)*	OK	Indicates that the service completed automated diagnostic data collection, such as in response to hardware errors.
DiagnosticDataCollectionFailed *(v1.2+)*	Critical	Indicates that the service failed to collect diagnostic data automatically based on its own policies, such as in response to hardware errors.
DiagnosticDataCollectionStarted *(v1.2+)*	OK	Indicates that the service started collecting diagnostic data automatically based on its own policies, such as in response to hardware errors.
DiagnosticDataInvalid *(v1.2+)*	Critical	Indicates that the service completed collecting diagnostic data automatically based on its own policies, such as in response to hardware errors, but the data is not usable. Examples includes truncated diagnostic data and invalid checksums for the diagnostic data.
LogCleared *(v1.1+)*	OK	Indicates that the log was cleared.

2.13.1 DiagnosticDataBlocked

Example: 'CPER' diagnostic data collection was attempted for device 'CPU0', but was blocked.

Indicates that the service attempted to collect diagnostic data automatically based on its own policies, such as in response to hardware errors, but restrictions are preventing it from starting. For example, a setting in a device that blocks sideband access to collecting diagnostic data.

- This message shall indicate that the service attempted to collect diagnostic data automatically based on its own policies, such as in response to hardware errors, but restrictions are preventing it from starting.

Version Added: v1.2

Severity: Warning

Resolution: Check device settings or other policies that can prevent the collection of diagnostic data.

Message and Arguments: "'`<1>`' diagnostic data collection was attempted for device '`<2>`', but was blocked."

1. *string*: The type of diagnostic data.
   - This argument shall contain the type of diagnostic data the service is collecting. The value shall be one of the values defined by the `LogDiagnosticDataTypes` enumeration in the `LogEntry` schema or an OEM-defined value for `OEM` diagnostic data types.
2. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.

2.13.2 DiagnosticDataCollected

Example: 'Manager' diagnostic data collected.

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

- This message shall indicate that diagnostic data was collected due to a client invoking the `CollectDiagnosticData` action.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "'`<1>`' diagnostic data collected."

1. *string*: The value of the `DiagnosticDataType` parameter.
   - This argument shall contain the value of the `DiagnosticDataType` parameter provided by the client when the `CollectDiagnosticData` action was invoked.

2.13.3 DiagnosticDataCollectionCompleted

Example: 'CPER' diagnostic data collection completed for device 'CPU0'.

Indicates that the service completed automated diagnostic data collection, such as in response to hardware errors.

- This message shall indicate that the service completed automated diagnostic data collection based on its own policies.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "' `<1>` ' diagnostic data collection completed for device ' `<2>` '."

1. *string*: The type of diagnostic data.
   - This argument shall contain the type of diagnostic data the service is collecting. The value shall be one of the values defined by the `LogDiagnosticDataTypes` enumeration in the `LogEntry` schema or an OEM-defined value for `OEM` diagnostic data types.
2. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.

2.13.4 DiagnosticDataCollectionFailed

Example: 'CPER' diagnostic data collection failed for device 'CPU0'.

Indicates that the service failed to collect diagnostic data automatically based on its own policies, such as in response to hardware errors.

- This message shall indicate that the service failed to collect diagnostic data automatically based on its own policies.

Version Added: v1.2

Severity: Critical

Resolution: Check the service's storage availability. Check for additional conditions preventing the collection of diagnostic data.

Message and Arguments: "' `<1>` ' diagnostic data collection failed for device ' `<2>` '."

1. *string*: The type of diagnostic data.

    - This argument shall contain the type of diagnostic data the service is collecting. The value shall be one of the values defined by the `LogDiagnosticDataTypes` enumeration in the `LogEntry` schema or an OEM-defined value for `OEM` diagnostic data types.
2. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.13.5 DiagnosticDataCollectionStarted

Example: 'CPER' diagnostic data collection started for device 'CPU0'.

Indicates that the service started collecting diagnostic data automatically based on its own policies, such as in response to hardware errors.

- This message shall indicate that the service started collecting diagnostic data automatically based on its own policies.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "' `<1>` ' diagnostic data collection started for device ' `<2>` '."

1. *string*: The type of diagnostic data.
    - This argument shall contain the type of diagnostic data the service is collecting. The value shall be one of the values defined by the `LogDiagnosticDataTypes` enumeration in the `LogEntry` schema or an OEM-defined value for `OEM` diagnostic data types.
2. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.13.6 DiagnosticDataInvalid

Example: 'CPER' diagnostic data collection completed for device 'CPU0', but is not usable.

Indicates that the service completed collecting diagnostic data automatically based on its own policies, such as in response to hardware errors, but the data is not usable. Examples includes truncated diagnostic data and invalid checksums for the diagnostic data.

- This message shall indicate that the service completed collecting diagnostic data automatically based on its own policies, but the data is not usable.

Version Added: v1.2

Severity: Critical

Resolution: Check the service's storage availability. Check for additional conditions preventing the collection of diagnostic data.

Message and Arguments: "' `<1>` ' diagnostic data collection completed for device ' `<2>` ', but is not usable."

1. *string*: The type of diagnostic data.
   - This argument shall contain the type of diagnostic data the service is collecting. The value shall be one of the values defined by the `LogDiagnosticDataTypes` enumeration in the `LogEntry` schema or an OEM-defined value for `OEM` diagnostic data types.
2. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.

2.13.7 LogCleared

Example: Log 'SystemEventLog' was cleared.

Indicates that the log was cleared.

- This message shall indicate that the log was cleared.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Log ' `<1>` ' was cleared."

1. *string*: The name or identifier of the log.
   - This argument shall contain a string that identifies or describes the log.

2.14 ManagerEvent 1.2.0

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.2	2026.1	2025.4

This registry defines messages to be used by a service to indicate conditions or configuration changes related to a manager. The affected manager may provide the Redfish service.

Message	Severity	Description
ManagerGoingOfflineTemporarily *(v1.2+)*	OK	Indicates that a manager is going offline temporarily, such as due to a maintenance operation.
ManagerReset	OK	Indicates that a manager reset occurred.
ManagerResetWatchdogTimer	Warning	Indicates that a manager reset occurred due to a watchdog timer action.
ManagerSwitchedToBackupSoftwareImage	Warning	Indicates that a manager switched to a backup software image.
PeerInvalidState *(v1.2+)*	Warning	Indicates that the peer manager is not in a valid state for redundancy.
PeerMissing *(v1.2+)*	Critical	Indicates that the peer manager cannot be detected.
PeerNotProvisioned *(v1.2+)*	Warning	Indicates that the peer manager is not provisioned.
PeerUnavailable *(v1.2+)*	Critical	Indicates that the peer manager is present but not responding.
PeerVersionMismatch *(v1.2+)*	Warning	Indicates that the managers in the redundancy group are running mismatched firmware versions, preventing redundancy.
RedundancyDisabled *(v1.2+)*	Warning	Indicates that a manager redundancy group was manually disabled.
RedundancyGroupCritical *(v1.2+)*	Critical	Indicates that a manager redundancy group has a critical status.
RedundancyGroupNormal *(v1.2+)*	OK	Indicates that a manager redundancy group has returned to normal operations.
RedundancyGroupWarning *(v1.2+)*	Warning	Indicates that a manager redundancy group has a warning status.

2.14.1 ManagerGoingOfflineTemporarily

Example: Manager 'BMC' is going offline temporarily and should be available again in 120 seconds; clients should allow for up to 600 seconds before taking action.

Indicates that a manager is going offline temporarily, such as due to a maintenance operation.

- This message shall indicate that a manager is going offline temporarily.

Version Added: v1.2

Severity: OK

Resolution: Wait the specified amount of time before accessing the service again.

Message and Arguments: "Manager '`<1>`' is going offline temporarily and should be available again in `<2>` seconds; clients should allow for up to `<3>` seconds before taking action."

1. *string*: The name or identifier of the manager.

    - This argument shall contain a string that identifies the affected manager.
2. *number*: The duration the manager is expected to be offline in seconds.
    - This argument shall contain the duration the manager is expected to be offline in seconds.
3. *number*: The duration clients should allow for the manager to be offline in seconds.
    - This argument shall contain the duration clients should allow for the manager to be offline in seconds.

2.14.2 ManagerReset

Example: Manager 'Manager1' performed a 'ForceRestart' type of reset.

Indicates that a manager reset occurred.

- This message shall indicate that a manager reset occurred.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Manager '`<1>`' performed a '`<2>`' type of reset."

1. *string*: The name or identifier of the manager that was reset.
    - This argument shall contain a string that identifies the affected manager.
2. *string*: The type of reset that occurred.
    - This argument shall contain the type of reset that occurred. This argument shall contain one of the values of the `ResetType` enumeration from the `Resource` schema.

2.14.3 ManagerResetWatchdogTimer

Example: Manager 'Manager1' performed a 'ForceRestart' type of reset due to watchdog timer action.

Indicates that a manager reset occurred due to a watchdog timer action.

- This message shall indicate that a manager reset occurred due to a watchdog timer action.

Version Added: v1.0

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Manager '`<1>`' performed a '`<2>`' type of reset due to watchdog timer action."

1. *string*: The name or identifier of the manager that was reset.
    - This argument shall contain a string that identifies the affected manager.
2. *string*: The type of reset that occurred.
    - This argument shall contain the type of reset that occurred. This argument shall contain one of the values of the `ResetType` enumeration from the `Resource` schema.

2.14.4 ManagerSwitchedToBackupSoftwareImage

Example: Manager 'Manager1' switched to a backup software image.

Indicates that a manager switched to a backup software image.

- This message shall indicate that a manager is using a backup software image.

Version Added: v1.0

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Manager '`<1>`' switched to a backup software image."

1. *string*: The name or identifier of the manager.
    - This argument shall contain a string that identifies the affected manager.

2.14.5 PeerInvalidState

Example: The peer manager of 'Manager1' is not in a valid state for redundancy.

Indicates that the peer manager is not in a valid state for redundancy.

- This message shall indicate that the peer manager is not in a valid state to establish redundancy, and redundancy is therefore unavailable.

Version Added: v1.2

Severity: Warning

Resolution: Check the peer manager state and ensure it is configured correctly for redundancy.

Message and Arguments: "The peer manager of '`<1>`' is not in a valid state for redundancy."

1. *string*: The name or identifier of the manager reporting the condition.
    - This argument shall contain a string that identifies the manager that detected the invalid state on the peer.

2.14.6 PeerMissing

Example: The peer manager of 'Manager1' is missing.

Indicates that the peer manager cannot be detected.

- This message shall indicate that the peer manager cannot be detected and redundancy is therefore unavailable.

Version Added: v1.2

Severity: Critical

Resolution: Check that the peer manager is installed, powered, and connected.

Message and Arguments: "The peer manager of '`<1>`' is missing."

1. *string*: The name or identifier of the manager reporting the condition.
   - This argument shall contain a string that identifies the manager that detected the missing peer.

2.14.7 PeerNotProvisioned

Example: The peer manager of 'Manager1' is not provisioned.

Indicates that the peer manager is not provisioned.

- This message shall indicate that the peer manager is present but is not provisioned, and redundancy is therefore unavailable.

Version Added: v1.2

Severity: Warning

Resolution: Complete provisioning of the peer manager.

Message and Arguments: "The peer manager of '`<1>`' is not provisioned."

1. *string*: The name or identifier of the manager reporting the condition.
   - This argument shall contain a string that identifies the manager that detected the unprovisioned peer.

2.14.8 PeerUnavailable

Example: The peer manager of 'Manager1' is not responding.

Indicates that the peer manager is present but not responding.

- This message shall indicate that the peer manager is present but not responding, and redundancy is therefore unavailable.

Version Added: v1.2

Severity: Critical

Resolution: Check the peer manager for errors and verify its health status.

Message and Arguments: "The peer manager of '`<1>`' is not responding."

1. *string*: The name or identifier of the manager reporting the condition.
    - This argument shall contain a string that identifies the manager that detected the unresponsive peer.

2.14.9 PeerVersionMismatch

Example: Manager 'Manager1' and its peer are running mismatched firmware versions.

Indicates that the managers in the redundancy group are running mismatched firmware versions, preventing redundancy.

- This message shall indicate that the managers in the redundancy group are running mismatched firmware versions, and redundancy is therefore unavailable.

Version Added: v1.2

Severity: Warning

Resolution: Update both managers to the same firmware version.

Message and Arguments: "Manager '`<1>`' and its peer are running mismatched firmware versions."

1. *string*: The name or identifier of the manager reporting the condition.
    - This argument shall contain a string that identifies the manager that detected the firmware version mismatch.

2.14.10 RedundancyDisabled

Example: Redundancy on redundancy group 'BmcRedundancy' was manually disabled.

Indicates that a manager redundancy group was manually disabled.

- This message shall indicate that a manager redundancy group was manually disabled by a client.

Version Added: v1.2

Severity: Warning

Resolution: Re-enable redundancy if it was disabled unintentionally.

Message and Arguments: "Redundancy on redundancy group '`<1>`' was manually disabled."

1. *string*: The name or identifier of the redundancy group.
   - This argument shall contain a string that identifies the affected redundancy group.

2.14.11 RedundancyGroupCritical

Example: Manager redundancy group 'BmcRedundancy' is in a critical state.

Indicates that a manager redundancy group has a critical status.

- This message shall indicate that a manager redundancy group is operating in a critical state due to a fault or a change in configuration.

Version Added: v1.2

Severity: Critical

Resolution: None.

Message and Arguments: "Manager redundancy group '`<1>`' is in a critical state."

1. *string*: The name or identifier of the manager redundancy group.
   - This argument shall contain a string that identifies or describes the group of `Manager` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical

2.14.12 RedundancyGroupNormal

Example: Manager redundancy group 'BmcRedundancy' is operating normally.

Indicates that a manager redundancy group has returned to normal operations.

- This message shall indicate that a manager redundancy group has returned to a normal operating state.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "Manager redundancy group '`<1>`' is operating normally."

1. *string*: The name or identifier of the manager redundancy group.
   - This argument shall contain a string that identifies or describes the group of `Manager` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.14.13 RedundancyGroupWarning

Example: Manager redundancy group 'BmcRedundancy' is in a warning state.

Indicates that a manager redundancy group has a warning status.

- This message shall indicate that a manager redundancy group is operating with a warning status or in a non-redundant state due to a fault or a change in configuration.

Version Added: v1.2

Severity: Warning

Resolution: None.

Message and Arguments: "Manager redundancy group '`<1>`' is in a warning state."

1. *string*: The name or identifier of the manager redundancy group.
   - This argument shall contain a string that identifies or describes the group of `Manager` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.15 NetworkDevice 1.2.1

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.1	2024.1	2020.1

This registry defines the messages for networking devices.

Message	Severity	Description
CableInserted	OK	Indicates that a network cable was inserted.
CableRemoved	OK	Indicates that a network cable was removed.
ConnectionDropped	OK	Indicates that a network connection was dropped.
ConnectionEstablished	OK	Indicates that a network connection was established.
ConnectionSpeedLow *(v1.1+)*	Warning	Indicates that a network connection is operating at a speed below the specified threshold.
DegradedConnectionEstablished	Warning	Indicates that a network connection was established, but at an unexpectedly low link speed.
LinkFlapDetected	Warning	Indicates that a network connection is highly unstable.

2.15.1 CableInserted

Example: A network cable was inserted into network adapter 'NIC1' port 'Port1'.

Indicates that a network cable was inserted.

- This message shall indicate that a network cable was inserted. This message shall not be sent for a backplane connection.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A network cable was inserted into network adapter ' `<1>` ' port ' `<2>` '."

1. *string*: The name or identifier of the network adapter.
   - This argument shall contain a string that identifies or describes the location or physical context of the network adapter into which a cable was plugged.
2. *string*: The name or identifier of the network port.
   - This argument shall contain a string that identifies or describes the location or physical context of the network port into which a cable was plugged.

2.15.2 CableRemoved

Example: A cable was removed from network adapter 'NIC1' port 'Port1'.

Indicates that a network cable was removed.

- This message shall indicate that a network cable was removed. This message shall not be sent for a backplane connection.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A cable was removed from network adapter '`<1>`' port '`<2>`'."

1. *string*: The name or identifier of the network adapter.
   - This argument shall contain a string that identifies or describes the location or physical context of the network adapter from which a cable was unplugged.
2. *string*: The name or identifier of the network port.
   - This argument shall contain a string that identifies or describes the location or physical context of the network port from which a cable was unplugged.

2.15.3 ConnectionDropped

Example: The connection is no longer active for network adapter 'NIC1' port 'Port1' function 'Function1'.

Indicates that a network connection was dropped.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The connection is no longer active for network adapter '`<1>`' port '`<2>`' function '`<3>`'."

1. *string*: The name or identifier of the network adapter.
   - This argument shall contain a string that identifies or describes the location or physical context of the network adapter from which a connection was dropped.
2. *string*: The name or identifier of the network port.
   - This argument shall contain a string that identifies or describes the location or physical context of the network port from which a connection was dropped.
3. *string*: The name or identifier of the network function.
   - This argument shall contain a string that identifies or describes the location or physical context of the network device function from which a connection was dropped.

2.15.4 ConnectionEstablished

Example: A network connection was established for network adapter 'NIC1' port 'Port1' function 'Function1'.

Indicates that a network connection was established.

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

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A network connection was established for network adapter '`<1>`' port '`<2>`' function '`<3>`'."

1. *string*: The name or identifier of the network adapter.
   - This argument shall contain a string that identifies or describes the location or physical context of the network adapter to which a connection was established.
2. *string*: The name or identifier of the network port.
   - This argument shall contain a string that identifies or describes the location or physical context of the network port to which a connection was established.
3. *string*: The name or identifier of the network function.
   - This argument shall contain a string that identifies or describes the location or physical context of the network device function to which a connection was established.

2.15.5 ConnectionSpeedLow

Example: The network adapter 'NIC1' port 'Port1' function 'Function1' is operating at a link speed of 1 Gbps which is below the specified threshold of 10 Gbps.

Indicates that a network connection is operating at a speed below the specified threshold.

- This message shall indicate that a network connection is operating at a link speed lower than the specified threshold.

Version Added: v1.1

Severity: Warning

Resolution: Check the network equipment and cabling for this connection.

Message and Arguments: "The network adapter '<1>' port '<2>' function '<3>' is operating at a link speed of <4> Gbps which is below the specified threshold of <5> Gbps."

1. *string*: The name or identifier of the network adapter.
   - This argument shall contain a string that identifies or describes the location or physical context of the associated network adapter.
2. *string*: The name or identifier of the network port.
   - This argument shall contain a string that identifies or describes the location or physical context of the associated network port.
3. *string*: The name or identifier of the network function.
   - This argument shall contain a string that identifies or describes the location or physical context of the associated network device function.
4. *number*: The current link speed (Gbps).
   - This argument shall contain the current connection speed (Gbps).
5. *number*: The alert threshold (Gbps).
   - This argument shall contain the value of the threshold that triggered this message.

2.15.6 DegradedConnectionEstablished

Example: A degraded network connection was established for network adapter 'NIC1' port 'Port1' function 'Function1'.

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

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

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "A degraded network connection was established for network adapter '<1>' port '<2>' function '<3>'."

1. *string*: The name or identifier of the network adapter.
   - This argument shall contain a string that identifies or describes the location or physical context of the network adapter to which a connection was established.
2. *string*: The name or identifier of the network port.
   - This argument shall contain a string that identifies or describes the location or physical context of the network port to which a connection was established.
3. *string*: The name or identifier of the network function.

- This argument shall contain a string that identifies or describes the location or physical context of the network device function to which a connection was established.

2.15.7 LinkFlapDetected

Example: The network connection for network adapter 'NIC1' port 'Port1' function 'Function1' was established and dropped 10 times in the last 5 minutes.

Indicates that a network connection is highly unstable.

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

Version Added: v1.0

Severity: Warning

Resolution: Contact the network administrator for problem resolution.

Message and Arguments: "The network connection for network adapter ' `<1>` ' port ' `<2>` ' function ' `<3>` ' was established and dropped ' `<4>` ' times in the last ' `<5>` ' minutes."

1. *string*: The name or identifier of the network adapter.
   - This argument shall contain a string that identifies or describes the location or physical context of the associated network adapter.
2. *string*: The name or identifier of the network port.
   - This argument shall contain a string that identifies or describes the location or physical context of the associated network port.
3. *string*: The name or identifier of the network function.
   - This argument shall contain a string that identifies or describes the location or physical context of the associated network device function.
4. *number*: The number of times the link has flapped.
   - This argument shall contain be a number representing the count of link establishment/disconnection cycles.
5. *number*: The number of minutes over which the link has flapped.
   - This argument shall contain be a number representing the number of minutes over which link flapping activity was detected.

2.16 Platform 1.5.0

Version	*v1.5*	*v1.4*	*v1.3*	*v1.2*	*v1.1*	*v1.0*

Release	2026.2	2026.1	2025.2	2024.4	2024.1	2024.1

This registry defines messages for compute platforms, covering topics related to processor, memory, and I/O device connectivity.

Message	Severity	Description
CPUMismatch *(v1.5+)*	Warning	Indicates that a CPU mismatch was detected.
DeviceAssemblyDataInaccessible *(v1.3+, deprecated v1.5)*	Warning	Indicates that the assembly data, historically 'FRU data', for a device is inaccessible.
DeviceAssemblyDataMalformed *(v1.3+, deprecated v1.5)*	Warning	Indicates that the assembly data, historically 'FRU data', for a device is malformed, such as due to a checksum error.
DeviceBootTimeout *(v1.3+, deprecated v1.5)*	Critical	Indicates that a device encountered a boot timeout.
DeviceDisabled *(v1.3+, deprecated v1.5)*	Warning	Indicates that a device was disabled or halted.
DeviceInRecovery *(v1.3+, deprecated v1.5)*	Critical	Indicates that a device is not operating properly and entered recovery mode to restore firmware or critical data to a state of integrity.
DeviceManagementInterfaceError *(v1.3+, deprecated v1.5)*	Critical	Indicates that the management interface for a device encountered an error.
DeviceManagementInterfaceUnresponsive *(v1.3+, deprecated v1.5)*	Critical	Indicates that the management interface for a device is unresponsive.
DeviceMissing *(v1.3+, deprecated v1.5)*	Critical	Indicates that the device is absent.
DeviceSecureBootFailure *(v1.3+, deprecated v1.5)*	Critical	Indicates that a device encountered a secure boot failure.
DeviceWatchdogTimeout *(v1.3+, deprecated v1.5)*	Critical	Indicates that a device encountered a watchdog timeout.
MemoryContainedError *(v1.3+)*	Warning	Indicates that a memory device detected an uncorrectable error, but contained it.
MemoryMirroringDegraded *(v1.5+)*	Warning	Indicates that memory mirroring degraded for a memory device. The platform continues to operate on the remaining copy, but data redundancy was lost.
MemoryMirroringFailed *(v1.5+)*	Critical	Indicates that memory mirroring failed for a memory device. Both mirror copies are unavailable or the mirroring mechanism itself failed.
MemoryParityError *(v1.5+)*	Critical	Indicates that a memory device detected a parity error.
MemoryRowRemappingError *(v1.3+)*	Critical	Indicates that a memory device encountered a row remapping failure.
MemoryRowRemappingPending *(v1.3+)*	Warning	Indicates that row remapping is pending for a memory device.

Message	Severity	Description
MemorySparingActivated *(v1.5+)*	OK	Indicates that memory sparing was activated for a memory device. Sparing activation indicates that the primary memory region experienced sufficient errors to trigger automatic replacement.
MemorySparingFailed *(v1.5+)*	Critical	Indicates that memory sparing failed for a memory device. The spare bank could not substitute the failing memory region.
MemoryTrainingFailed *(v1.5+)*	Critical	Indicates that training failed for a memory device.
MemoryUncorrectableError *(v1.3+)*	Critical	Indicates that a memory device detected an uncorrectable error and the error was not contained.
NVRAMClearAsserted *(v1.1+)*	Warning	Indicates that the 'NVRAM Clear' jumper or switch is currently asserted.
OperatingSystemCrash	Critical	Indicates the operating system was halted due to a catastrophic error.
PCIeCompletionTimeoutDetected *(v1.5+)*	Critical	Indicates that a PCIe device encountered a completion timeout.
PCIeECRCErrorDetected *(v1.5+)*	Warning	Indicates that a PCIe device encountered an End-to-End Cyclic Redundancy Check (ECRC) error.
PCIeFatalErrorDetected *(v1.5+)*	Critical	Indicates that a PCIe device encountered a fatal error.
PCIeHeaderLogOverflow *(v1.5+)*	Warning	Indicates that a PCIe device encountered a header log overflow.
PCIeIncorrectLinkSpeed *(v1.3+)*	Critical	Indicates that a PCIe device did not train to its expected link speed.
PCIeIncorrectLinkWidth *(v1.3+)*	Critical	Indicates that a PCIe device did not train to its expected link width.
PCIeLinkDown *(v1.3+)*	Critical	Indicates that a PCIe device unexpectedly lost its link with the upstream interface. This is also known as a surprise link down.
PCIeLinkRestored *(v1.3+)*	Critical	Indicates that a PCIe device restored its link with the upstream interface.
PCIeLinkSpeedChange *(v1.3+)*	Critical	Indicates that a PCIe link speed changed at runtime.
PCIeLinkTrainingError *(v1.3+)*	Critical	Indicates that a PCIe device could not establish a link with the upstream interface.
PCIeLinkWidthChange *(v1.3+)*	Critical	Indicates that a PCIe link width changed at runtime.
PCIeManagedLinkDown *(v1.5+)*	OK	Indicates that a PCIe device lost its link with the upstream interface due to a controlled request.
PCIeReplayTimeoutDetected *(v1.5+)*	Warning	Indicates that a PCIe device encountered a replay timeout.
PCIeUncorrectableError *(v1.3+)*	Critical	Indicates that a PCIe device encountered an uncorrectable error.
PlatformError	Warning	Indicates that a platform error occurred.
PlatformErrorAtLocation	Warning	Indicates that a platform error occurred and device or other location information is available.

Message	Severity	Description
PlatformReset *(v1.2+)*	OK	Indicates that the platform reset occurred.
PlatformResetWatchdogTimer *(v1.2+)*	Warning	Indicates that the platform reset occurred due to a watchdog timer action.
PowerBrake *(v1.3+)*	Warning	Indicates that a power brake is applied to a component or equipment.
PowerSubsystemDegraded *(v1.5+)*	Warning	Indicates that the platform power subsystem was degraded.
PowerSubsystemRestored *(v1.5+)*	OK	Indicates that the platform power subsystem was restored to a healthy state.
SecurityBypassAsserted *(v1.1+)*	Warning	Indicates that the 'Security Bypass' jumper or switch is currently asserted.
ThermalSubsystemDegraded *(v1.5+)*	Warning	Indicates that the platform thermal subsystem was degraded.
ThermalSubsystemRestored *(v1.5+)*	OK	Indicates that the platform thermal subsystem was restored to a healthy state.
ThermalTrip *(v1.5+)*	Critical	Indicates that a platform-level thermal trip occurred.
Throttled *(v1.5+)*	Warning	Indicates that a platform-level throttling occurred.
UnhandledExceptionDetectedAfterReset	Critical	Indicates that an unhandled exception caused the platform to reset.
VoltageRegulatorFault *(v1.3+)*	Critical	Indicates that a voltage regulator has faulted.

2.16.1 CPUMismatch

Example: CPU mismatch detected on 'System4'.

Indicates that a CPU mismatch was detected.

- This message shall indicate that a mismatch was detected between processors in a multi-socket system.

Version Added: v1.5

Severity: Warning

Resolution: Check the CPU configuration of the platform.

Message and Arguments: "CPU mismatch detected on '`<1>`'."

1. *string*: The name or identifier of the platform.
   - This argument shall contain a string that identifies the name or identifier of the platform for which this condition was detected.

2.16.2 DeviceAssemblyDataInaccessible

Example: Unable to access the assembly data for device 'GPU1'.

Indicates that the assembly data, historically 'FRU data', for a device is inaccessible.

- This message shall indicate that the assembly data for a device is inaccessible.

Deprecated in v1.5 and later. This message has been deprecated in favor of `AssemblyDataInaccessible` *in the Device Message Registry.*

Version Added: v1.3

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Unable to access the assembly data for device '`<1>`'."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.16.3 DeviceAssemblyDataMalformed

Example: Malformed assembly data detected for device 'GPU1'.

Indicates that the assembly data, historically 'FRU data', for a device is malformed, such as due to a checksum error.

- This message shall indicate that the assembly data for a device is malformed, such as due to a checksum error.

Deprecated in v1.5 and later. This message has been deprecated in favor of `AssemblyDataMalformed` *in the Device Message Registry.*

Version Added: v1.3

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Malformed assembly data detected for device '`<1>`'."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.16.4 DeviceBootTimeout

Example: Device 'GPU1' encountered a boot timeout.

Indicates that a device encountered a boot timeout.

- This message shall indicate that a device encountered a boot timeout.

Deprecated in v1.5 and later. This message has been deprecated in favor of `BootTimeout` *in the Device Message Registry.*

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' encountered a boot timeout."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.16.5 DeviceDisabled

Example: Device 'GPU1' was disabled or halted.

Indicates that a device was disabled or halted.

- This message shall indicate that a device was disabled or halted.

Deprecated in v1.5 and later. This message has been deprecated in favor of `Disabled` *in the Device Message Registry.*

Version Added: v1.3

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' was disabled or halted."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.16.6 DeviceInRecovery

Example: Device 'GPU1' entered recovery mode due to: Firmware corruption.

Indicates that a device is not operating properly and entered recovery mode to restore firmware or critical data to a state of integrity.

- This message shall indicate that a device is not operating properly and entered recovery mode, such as defined by the OCP Secure Firmware Recovery specification, to restore firmware or critical data to a state of integrity.

Deprecated in v1.5 and later. This message has been deprecated in favor of `InRecovery` *in the Device Message Registry.*

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' entered recovery mode due to: `<2>`."

1. *string*: The name or identifier of the device.
   - This argument shall contain a string that identifies or describes the device.
2. *string*: The reason the device entered recovery mode.
   - This argument shall contain a string that describes the reason the device entered recovery mode.

2.16.7 DeviceManagementInterfaceError

Example: The management interface for device 'GPU1' encountered the error: I2C communication failure.

Indicates that the management interface for a device encountered an error.

- This message shall indicate that the management interface for a device encountered an error.

Deprecated in v1.5 and later. This message has been deprecated in favor of `ManagementInterfaceError` *in the Device Message Registry.*

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "The management interface for device '`<1>`' encountered the error: `<2>`."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.
2. *string*: The error encountered on the management interface of the device.
    - This argument shall contain a string that describes error encountered on the management interface of the device.

2.16.8 DeviceManagementInterfaceUnresponsive

Example: The management interface for device 'GPU1' is unresponsive.

Indicates that the management interface for a device is unresponsive.

- This message shall indicate that the management interface for a device is unresponsive.

Deprecated in v1.5 and later. This message has been deprecated in favor of `ManagementInterfaceUnresponsive` *in the Device Message Registry.*

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "The management interface for device '`<1>`' is unresponsive."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.16.9 DeviceMissing

Example: Device 'GPU1' state changed to absent.

Indicates that the device is absent.

- This message shall be used to indicate the device is absent.

Deprecated in v1.5 and later. This message has been deprecated in favor of `Missing` *in the Device Message Registry.*

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' state changed to absent."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.16.10 DeviceSecureBootFailure

Example: Device 'GPU1' encountered a secure boot failure.

Indicates that a device encountered a secure boot failure.

- This message shall indicate that a device encountered a secure boot failure.

Deprecated in v1.5 and later. This message has been deprecated in favor of `SecureBootFailure` in the Device Message Registry.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' encountered a secure boot failure."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.16.11 DeviceWatchdogTimeout

Example: Device 'GPU1' encountered a watchdog timeout.

Indicates that a device encountered a watchdog timeout.

- This message shall indicate that a device encountered a watchdog timeout.

Deprecated in v1.5 and later. This message has been deprecated in favor of `WatchdogTimeout` in the Device Message Registry.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Device '`<1>`' encountered a watchdog timeout."

1. *string*: The name or identifier of the device.
    - This argument shall contain a string that identifies or describes the device.

2.16.12 MemoryContainedError

Example: Memory device 'DIMM1' contained an uncorrectable error.

Indicates that a memory device detected an uncorrectable error, but contained it.

- This message shall indicate that a memory device detected an uncorrectable error, but contained it.

Version Added: v1.3

Severity: Warning

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Memory device '`<1>`' contained an uncorrectable error."

1. *string*: The name or identifier of the memory device.
    - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.13 MemoryMirroringDegraded

Example: Memory mirroring degraded for memory device 'DIMM1'.

Indicates that memory mirroring degraded for a memory device. The platform continues to operate on the remaining copy, but data redundancy was lost.

- This message shall indicate that memory mirroring degraded for a memory device.

Version Added: v1.5

Severity: Warning

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Memory mirroring degraded for memory device '`<1>`'."

1. *string*: The name or identifier of the memory device.
    - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.14 MemoryMirroringFailed

Example: Memory mirroring failed for memory device 'DIMM1'.

Indicates that memory mirroring failed for a memory device. Both mirror copies are unavailable or the mirroring mechanism itself failed.

- This message shall indicate that memory mirroring failed for a memory device.

Version Added: v1.5

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Memory mirroring failed for memory device '`<1>`'."

1. *string*: The name or identifier of the memory device.
   - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.15 MemoryParityError

Example: Memory device 'DIMM1' detected a parity error.

Indicates that a memory device detected a parity error.

- This message shall indicate that a memory device detected a parity error.

Version Added: v1.5

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Memory device '`<1>`' detected a parity error."

1. *string*: The name or identifier of the memory device.
   - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.16 MemoryRowRemappingError

Example: Memory device 'DIMM1' encountered a row remapping failure.

Indicates that a memory device encountered a row remapping failure.

- This message shall indicate that a memory device encountered a row remapping failure.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Memory device '`<1>`' encountered a row remapping failure."

1. *string*: The name or identifier of the memory device.
   - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.17 MemoryRowRemappingPending

Example: Row remapping is pending for memory device 'DIMM1'.

Indicates that row remapping is pending for a memory device.

- This message shall indicate that row remapping is pending for a memory device.

Version Added: v1.3

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Row remapping is pending for memory device '`<1>`'."

1. *string*: The name or identifier of the memory device.
   - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.18 MemorySparingActivated

Example: Memory sparing was activated for memory device 'DIMM1'.

Indicates that memory sparing was activated for a memory device. Sparing activation indicates that the primary memory region experienced sufficient errors to trigger automatic replacement.

- This message shall indicate that memory sparing was activated for a memory device.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "Memory sparing was activated for memory device '`<1>`'."

1. *string*: The name or identifier of the memory device.
   - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.19 MemorySparingFailed

Example: Memory sparing failed for memory device 'DIMM1'.

Indicates that memory sparing failed for a memory device. The spare bank could not substitute the failing memory region.

- This message shall indicate that memory sparing failed for a memory device.

Version Added: v1.5

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Memory sparing failed for memory device '`<1>`'."

1. *string*: The name or identifier of the memory device.
   - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.20 MemoryTrainingFailed

Example: Training failed for memory device 'DIMM1'.

Indicates that training failed for a memory device.

- This message shall indicate that training failed for a memory device.

Version Added: v1.5

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Training failed for memory device '`<1>`'."

1. *string*: The name or identifier of the memory device.
   - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.21 MemoryUncorrectableError

Example: Memory device 'DIMM1' detected an uncorrectable error.

Indicates that a memory device detected an uncorrectable error and the error was not contained.

- This message shall indicate that a memory device detected an uncorrectable error and the error was not contained.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Memory device '`<1>`' detected an uncorrectable error."

1. *string*: The name or identifier of the memory device.
   - This argument shall contain a string that identifies or describes the `Memory` resource.

2.16.22 NVRAMClearAsserted

Example: 'NVRAM Clear' is currently on.

Indicates that the 'NVRAM Clear' jumper or switch is currently asserted.

- This message shall indicate that the 'NVRAM Clear' jumper or switch is currently asserted.

Version Added: v1.1

Severity: Warning

Resolution: Turn off the 'NVRAM Clear' jumper or switch.

Message and Arguments: "'NVRAM Clear' is currently on."

2.16.23 OperatingSystemCrash

Example: An operating system crash occurred.

Indicates the operating system was halted due to a catastrophic error.

- This message shall indicate that an unhandled exception caused a compute platform to crash or otherwise halt.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "An operating system crash occurred."

2.16.24 PCIeCompletionTimeoutDetected

Example: PCIe device 'GPU1' encountered a completion timeout.

Indicates that a PCIe device encountered a completion timeout.

- This message shall indicate that a PCIe device encountered a completion timeout.

Version Added: v1.5

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' encountered a completion timeout."

1. *string*: The name or identifier of the PCIe device.
   - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.25 PCIeECRCErrorDetected

Example: PCIe device 'GPU1' encountered an ECRC error.

Indicates that a PCIe device encountered an End-to-End Cyclic Redundancy Check (ECRC) error.

- This message shall indicate that a PCIe device encountered an End-to-End Cyclic Redundancy Check (ECRC) error.

Version Added: v1.5

Severity: Warning

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' encountered an ECRC error."

1. *string*: The name or identifier of the PCIe device.

    - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.26 PCIeFatalErrorDetected

Example: PCIe device 'GPU1' encountered a fatal error.

Indicates that a PCIe device encountered a fatal error.

- This message shall indicate that a PCIe device encountered a fatal error.

Version Added: v1.5

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' encountered a fatal error."

1. *string*: The name or identifier of the PCIe device.
    - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.27 PCIeHeaderLogOverflow

Example: PCIe device 'GPU1' encountered a header log overflow.

Indicates that a PCIe device encountered a header log overflow.

- This message shall indicate that a PCIe device encountered a header log overflow.

Version Added: v1.5

Severity: Warning

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' encountered a header log overflow."

1. *string*: The name or identifier of the PCIe device.
    - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.28 PCIeIncorrectLinkSpeed

Example: PCIe device 'GPU1' did not train to its expected link speed.

Indicates that a PCIe device did not train to its expected link speed.

- This message shall indicate that a PCIe device did not train to its expected link speed.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' did not train to its expected link speed."

1. *string*: The name or identifier of the PCIe device.
   - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.29 PCIeIncorrectLinkWidth

Example: PCIe device 'GPU1' did not train to its expected link width.

Indicates that a PCIe device did not train to its expected link width.

- This message shall indicate that a PCIe device did not train to its expected link width.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' did not train to its expected link width."

1. *string*: The name or identifier of the PCIe device.
   - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.30 PCIeLinkDown

Example: PCIe device 'GPU1' unexpectedly lost its link.

Indicates that a PCIe device unexpectedly lost its link with the upstream interface. This is also known as a surprise link down.

- This message shall indicate that a PCIe device unexpectedly lost its link with the upstream interface.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' unexpectedly lost its link."

1. *string*: The name or identifier of the PCIe device.
    - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.31 PCIeLinkRestored

Example: PCIe device 'GPU1' restored its link.

Indicates that a PCIe device restored its link with the upstream interface.

- This message shall indicate that a PCIe device restored its link with the upstream interface.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' restored its link."

1. *string*: The name or identifier of the PCIe device.
    - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.32 PCIeLinkSpeedChange

Example: PCIe device 'GPU1' detected a runtime speed change to 8 Gbps.

Indicates that a PCIe link speed changed at runtime.

- This message shall indicate that a PCIe link speed changed at runtime.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' detected a runtime speed change to `<2>` Gbps."

1. *string*: The name or identifier of the PCIe device.
    - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.
2. *number*: The new link speed (Gbps).

    - This argument shall contain the new link speed (Gbps).

2.16.33 PCIeLinkTrainingError

Example: PCIe device 'GPU1' could not train its link.

Indicates that a PCIe device could not establish a link with the upstream interface.

- This message shall indicate that a PCIe device could not establish a link with the upstream interface.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' could not train its link."

1. *string*: The name or identifier of the PCIe device.
    - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.34 PCIeLinkWidthChange

Example: PCIe device 'GPU1' detected a runtime link width change to 8 lanes.

Indicates that a PCIe link width changed at runtime.

- This message shall indicate that a PCIe link width changed at runtime.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' detected a runtime link width change to `<2>` lanes."

1. *string*: The name or identifier of the PCIe device.
    - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.
2. *number*: The new link width.
    - This argument shall contain the new link width.

2.16.35 PCIeManagedLinkDown

Example: PCIe device 'GPU1' lost its link due to a controlled request.

Indicates that a PCIe device lost its link with the upstream interface due to a controlled request.

- This message shall indicate that a PCIe device lost its link with the upstream interface due to a controlled request.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "PCIe device '`<1>`' lost its link due to a controlled request."

1. *string*: The name or identifier of the PCIe device.
   - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.36 PCIeReplayTimeoutDetected

Example: PCIe device 'GPU1' encountered a replay timeout.

Indicates that a PCIe device encountered a replay timeout.

- This message shall indicate that a PCIe device encountered a replay timeout.

Version Added: v1.5

Severity: Warning

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' encountered a replay timeout."

1. *string*: The name or identifier of the PCIe device.
   - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.37 PCIeUncorrectableError

Example: PCIe device 'GPU1' encountered an uncorrectable error.

Indicates that a PCIe device encountered an uncorrectable error.

- This message shall indicate that a PCIe device encountered an uncorrectable error.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "PCIe device '`<1>`' encountered an uncorrectable error."

1. *string*: The name or identifier of the PCIe device.
   - This argument shall contain a string that identifies or describes the `PCIeDevice` resource.

2.16.38 PlatformError

Example: A platform error occurred.

Indicates that a platform error occurred.

- This message shall indicate that a platform error occurred.

Version Added: v1.0

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "A platform error occurred."

2.16.39 PlatformErrorAtLocation

Example: A platform error occurred at location 'CPU1'.

Indicates that a platform error occurred and device or other location information is available.

- This message shall indicate that a platform error occurred with specific device identification information available.

Version Added: v1.0

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "A platform error occurred at location '`<1>`'."

1. *string*: The device description or location where the error occurred.
    - This argument shall contain a description of the device or location where the error occurred.

2.16.40 PlatformReset

Example: A platform performed a 'ForceRestart' type of reset.

Indicates that the platform reset occurred.

- This message shall indicate that a platform reset occurred.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "A platform performed a '`<1>`' type of reset."

1. *string*: The type of reset that occurred.
    - This argument shall contain the type of reset that occurred. This argument shall contain one of the values of the `ResetType` enumeration from the `Resource` schema.

2.16.41 PlatformResetWatchdogTimer

Example: A platform performed a 'ForceRestart' type of reset due to watchdog timer action.

Indicates that the platform reset occurred due to a watchdog timer action.

- This message shall indicate that a platform reset occurred due to a watchdog timer action.

Version Added: v1.2

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "A platform performed a '`<1>`' type of reset due to watchdog timer action."

1. *string*: The type of reset that occurred.
    - This argument shall contain the type of reset that occurred. This argument shall contain one of the values of the `ResetType` enumeration from the `Resource` schema.

2.16.42 PowerBrake

Example: A power brake is applied to 'CPU1'.

Indicates that a power brake is applied to a component or equipment.

- This message shall indicate that a power brake is applied to a component or equipment.

Version Added: v1.3

Severity: Warning

Resolution: Check additional diagnostic data if available.

Message and Arguments: "A power brake is applied to '`<1>`'."

1. *string*: The name or identifier of the component or equipment.
   - This argument shall contain a string that identifies or describes the component or equipment.

2.16.43 PowerSubsystemDegraded

Example: The power subsystem is degraded for 'System4'.

Indicates that the platform power subsystem was degraded.

- This message shall indicate that the platform power subsystem entered a degraded state. Degradation may result from, but not limited to, a power supply failure, loss of power redundancy, or a reduction in power capacity that places the power margin at risk.

Version Added: v1.5

Severity: Warning

Resolution: Check power sensors, power supply health, and power device operations.

Message and Arguments: "The power subsystem is degraded for '`<1>`'."

1. *string*: The name or identifier of the platform.
   - This argument shall contain a string that identifies the name or identifier of the platform for which this condition was detected.

2.16.44 PowerSubsystemRestored

Example: The power subsystem is restored for 'System4'.

Indicates that the platform power subsystem was restored to a healthy state.

- This message shall indicate that the platform power subsystem returned to a healthy operational state after a previous degradation event.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "The power subsystem is restored for '`<1>`'."

1. *string*: The name or identifier of the platform.
   - This argument shall contain a string that identifies the name or identifier of the platform for which this condition was detected.

2.16.45 SecurityBypassAsserted

Example: 'Security Bypass' is currently on.

Indicates that the 'Security Bypass' jumper or switch is currently asserted.

- This message shall indicate that the 'Security Bypass' or 'Password Bypass' jumper or switch is currently asserted.

Version Added: v1.1

Severity: Warning

Resolution: Turn off the 'Security Bypass' jumper or switch.

Message and Arguments: "'Security Bypass' is currently on."

2.16.46 ThermalSubsystemDegraded

Example: The thermal subsystem is degraded for 'System4'.

Indicates that the platform thermal subsystem was degraded.

- This message shall indicate that the platform thermal subsystem entered a degraded state. Degradation may result from, but not limited to, a cooling device failure, a reduction in cooling capacity that places the thermal margin at risk, or a critical thermal condition.

Version Added: v1.5

Severity: Warning

Resolution: Check thermal sensors, fan health, and cooling device operations.

Message and Arguments: "The thermal subsystem is degraded for '`<1>`'."

1. *string*: The name or identifier of the platform.
    - This argument shall contain a string that identifies the name or identifier of the platform for which this condition was detected.

2.16.47 ThermalSubsystemRestored

Example: The thermal subsystem is restored for 'System4'.

Indicates that the platform thermal subsystem was restored to a healthy state.

- This message shall indicate that the platform thermal subsystem returned to a healthy operational state after a previous degradation event.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "The thermal subsystem is restored for '`<1>`'."

1. *string*: The name or identifier of the platform.
    - This argument shall contain a string that identifies the name or identifier of the platform for which this condition was detected.

2.16.48 ThermalTrip

Example: A platform thermal trip occurred on 'System4'.

Indicates that a platform-level thermal trip occurred.

- This message shall indicate that the platform experienced a platform-level thermal trip condition, resulting in the platform shutdown.

Version Added: v1.5

Severity: Critical

Resolution: Check additional diagnostic data if available. Check thermal sensors, fan health, and cooling device operations.

Message and Arguments: "A platform thermal trip occurred on '`<1>`'."

1. *string*: The name or identifier of the platform.
    - This argument shall contain a string that identifies the name or identifier of the platform for which this condition was detected.

2.16.49 Throttled

Example: Platform throttling occurred on 'System4'.

Indicates that a platform-level throttling occurred.

- This message shall indicate that the platform experienced a platform-level throttling condition, resulting in degraded performance.

Version Added: v1.5

Severity: Warning

Resolution: Check additional diagnostic data if available. Check thermal sensors, fan health, and cooling device operations.

Message and Arguments: "Platform throttling occurred on '`<1>`'."

1. *string*: The name or identifier of the platform.
    - This argument shall contain a string that identifies the name or identifier of the platform for which this condition was detected.

2.16.50 UnhandledExceptionDetectedAfterReset

Example: An unhandled exception caused a platform reset.

Indicates that an unhandled exception caused the platform to reset.

- This message shall indicate that an unhandled exception caused a compute platform to reset. The exception was detected after the reset occurred.

Version Added: v1.0

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "An unhandled exception caused a platform reset."

2.16.51 VoltageRegulatorFault

Example: Voltage regulator 'VR1' has faulted.

Indicates that a voltage regulator has faulted.

- This message shall indicate that a voltage regulator has faulted.

Version Added: v1.3

Severity: Critical

Resolution: Check additional diagnostic data if available. Check the platform and connectors for any faults.

Message and Arguments: "Voltage regulator '`<1>`' has faulted."

1. *string*: The name or identifier of the voltage regulator.
   - This argument shall contain a string that identifies or describes the voltage regulator.

2.17 Power 1.4.0

Version	*v1.4*	*v1.3*	*v1.2*	*v1.1*	*v1.0*
Release	2026.2	2026.1	2025.4	2025.3	2024.1

This registry defines messages related to electrical measurements and power distribution equipment.

Message	Severity	Description
BreakerFault	Critical	Indicates that a circuit breaker has an internal fault.
BreakerReset	OK	Indicates that a circuit breaker reset.
BreakerTripped	Critical	Indicates that a circuit breaker tripped.
ChassisPowerLimitExceeded *(v1.3+)*	Warning	Indicates that a chassis exceeded its power limit.
ChassisPowerLimitRestored *(v1.3+)*	OK	Indicates that a chassis returned below its power limit.
ChassisPoweredOff *(v1.1+)*	OK	Indicates that the power state of a chassis changed to off.
ChassisPoweredOn *(v1.1+)*	OK	Indicates that the power state of a chassis changed to on.

Message	Severity	Description
CircuitPoweredOff	OK	Indicates that a circuit was powered off.
CircuitPoweredOn	OK	Indicates that a circuit was powered on.
CurrentAboveLowerCriticalThreshold	Warning	Indicates that a current reading is no longer below the lower critical threshold but is still outside of normal operating range.
CurrentAboveLowerFatalThreshold	Critical	Indicates that a current reading is no longer below the lower fatal threshold but is still outside of normal operating range.
CurrentAboveUpperCautionThreshold	Warning	Indicates that a current reading is above the upper caution threshold.
CurrentAboveUpperCriticalThreshold	Critical	Indicates that a current reading is above the upper critical threshold.
CurrentAboveUpperFatalThreshold	Critical	Indicates that a current reading is above the upper fatal threshold.
CurrentBelowLowerCautionThreshold	Warning	Indicates that a current reading is below the lower caution threshold.
CurrentBelowLowerCriticalThreshold	Critical	Indicates that a current reading is below the lower critical threshold.
CurrentBelowLowerFatalThreshold	Critical	Indicates that a current reading is below the lower fatal threshold.
CurrentBelowUpperCriticalThreshold	Warning	Indicates that a current reading is no longer above the upper critical threshold but is still outside of normal operating range.
CurrentBelowUpperFatalThreshold	Critical	Indicates that a current reading is no longer above the upper fatal threshold but is still outside of normal operating range.
CurrentCritical	Critical	Indicates that a current reading exceeds an internal critical level.
CurrentNoLongerCritical	Warning	Indicates that a current reading no longer exceeds an internal critical level but still exceeds an internal warning level.
CurrentNormal	OK	Indicates that a current reading is now within normal operating range.
CurrentWarning	Warning	Indicates that a current reading exceeds an internal warning level.
FrequencyAboveLowerCriticalThreshold	Warning	Indicates that a frequency reading is no longer below the lower critical threshold but is still outside of normal operating range.
FrequencyAboveUpperCautionThreshold	Warning	Indicates that a frequency reading is above the upper caution threshold.
FrequencyAboveUpperCriticalThreshold	Critical	Indicates that a frequency reading is above the upper critical threshold.
FrequencyBelowLowerCautionThreshold	Warning	Indicates that a frequency reading is below the lower caution threshold.
FrequencyBelowLowerCriticalThreshold	Critical	Indicates that a frequency reading is below the lower critical threshold.
FrequencyBelowUpperCriticalThreshold	Warning	Indicates that a frequency reading is no longer above the upper critical threshold but is still outside of normal operating range.
FrequencyCritical	Critical	Indicates that a frequency reading exceeds an internal critical level.

Message	Severity	Description
FrequencyNoLongerCritical	Warning	Indicates that a frequency reading no longer exceeds an internal critical level but still exceeds an internal warning level.
FrequencyNormal	OK	Indicates that a frequency reading is now within normal operating range.
FrequencyWarning	Warning	Indicates that a frequency reading exceeds an internal warning level.
LineInputPowerFault	Warning	Indicates a fault on an electrical power input.
LineInputPowerRestored	OK	Indicates that an electrical power input was restored to normal operation.
LossOfInputPower	Warning	Indicates a loss of power at an electrical input.
OutletPoweredOff	OK	Indicates that an outlet was powered off.
OutletPoweredOn	OK	Indicates that an outlet was powered on.
PowerAboveLowerCriticalThreshold	Warning	Indicates that a power reading is no longer below the lower critical threshold but is still outside of normal operating range.
PowerAboveLowerFatalThreshold	Critical	Indicates that a power reading is no longer below the lower fatal threshold but is still outside of normal operating range.
PowerAboveUpperCautionThreshold	Warning	Indicates that a power reading is above the upper caution threshold.
PowerAboveUpperCriticalThreshold	Critical	Indicates that a power reading is above the upper critical threshold.
PowerAboveUpperFatalThreshold	Critical	Indicates that a power reading is above the upper fatal threshold.
PowerBelowLowerCautionThreshold	Warning	Indicates that a power reading is below the lower caution threshold.
PowerBelowLowerCriticalThreshold	Critical	Indicates that a power reading is below the lower critical threshold.
PowerBelowLowerFatalThreshold	Critical	Indicates that a power reading is below the lower fatal threshold.
PowerBelowUpperCriticalThreshold	Warning	Indicates that a power reading is no longer above the upper critical threshold but is still outside of normal operating range.
PowerBelowUpperFatalThreshold	Critical	Indicates that a power reading is no longer above the upper fatal threshold but is still outside of normal operating range.
PowerBrakeApplied *(v1.4+)*	Warning	Indicates the power brake was applied to the equipment.
PowerBrakeReleased *(v1.4+)*	OK	Indicates the power brake was released from the equipment.
PowerCritical	Critical	Indicates that a power reading exceeds an internal critical level.
PowerDistributionGroupCritical *(v1.2+)*	Critical	Indicates that a power distribution group has a critical status.
PowerDistributionGroupNormal *(v1.2+)*	OK	Indicates that a power distribution group has returned to normal operations.
PowerDistributionGroupWarning *(v1.2+)*	Warning	Indicates that a power distribution group has a warning status.

Message	Severity	Description
PowerNoLongerCritical	Warning	Indicates that a power reading no longer exceeds an internal critical level but still exceeds an internal warning level.
PowerNormal	OK	Indicates that a power reading is now within normal operating range.
PowerSupplyFailed	Critical	Indicates that a power supply has failed.
PowerSupplyGroupCritical	Critical	Indicates that a power supply group has a critical status.
PowerSupplyGroupNormal	OK	Indicates that a power supply group has returned to normal operations.
PowerSupplyGroupWarning	Warning	Indicates that a power supply group has a warning status.
PowerSupplyInserted	OK	Indicates that a power supply was inserted or installed.
PowerSupplyPredictiveFailure	Warning	Indicates that the power supply predicted a future failure condition.
PowerSupplyRemoved	OK	Indicates that a power supply was removed.
PowerSupplyRestored	OK	Indicates that a power supply was repaired or restored to normal operation.
PowerSupplyWarning	Warning	Indicates that a power supply has a warning condition.
PowerWarning	Warning	Indicates that a power reading exceeds an internal warning level.
VoltageAboveLowerCriticalThreshold	Warning	Indicates that a voltage reading is no longer below the lower critical threshold but is still outside of normal operating range.
VoltageAboveLowerFatalThreshold	Critical	Indicates that a voltage reading is no longer below the lower fatal threshold but is still outside of normal operating range.
VoltageAboveUpperCautionThreshold	Warning	Indicates that a voltage reading is above the upper caution threshold.
VoltageAboveUpperCriticalThreshold	Critical	Indicates that a voltage reading is above the upper critical threshold.
VoltageAboveUpperFatalThreshold	Critical	Indicates that a voltage reading is above the upper fatal threshold.
VoltageBelowLowerCautionThreshold	Warning	Indicates that a voltage reading is below the lower caution threshold.
VoltageBelowLowerCriticalThreshold	Critical	Indicates that a voltage reading is below the lower critical threshold.
VoltageBelowLowerFatalThreshold	Critical	Indicates that a voltage reading is below the lower fatal threshold.
VoltageBelowUpperCriticalThreshold	Warning	Indicates that a voltage reading is no longer above the upper critical threshold but is still outside of normal operating range.
VoltageBelowUpperFatalThreshold	Critical	Indicates that a voltage reading is no longer above the upper fatal threshold but is still outside of normal operating range.
VoltageCritical	Critical	Indicates that a voltage reading exceeds an internal critical level.

Message	Severity	Description
VoltageNoLongerCritical	Warning	Indicates that a voltage reading no longer exceeds an internal critical level but still exceeds an internal warning level.
VoltageNormal	OK	Indicates that a voltage reading is now within normal operating range.
VoltageWarning	Warning	Indicates that a voltage reading exceeds an internal warning level.

2.17.1 BreakerFault

Example: Fault detected in breaker '14'.

Indicates that a circuit breaker has an internal fault.

- This message shall indicate that a fault was detected within a circuit breaker. The power state of the circuit may not be known.

Version Added: v1.0

Severity: Critical

Resolution: Check the breaker hardware and replace any faulty components.

Message and Arguments: "Fault detected in breaker '`<1>`'."

1. *string*: The name or identifier of the circuit.
   - This argument shall contain a string that identifies or describes the `Circuit` resource or the location of the circuit breaker.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical

2.17.2 BreakerReset

Example: Breaker '7' reset.

Indicates that a circuit breaker reset.

- This message shall indicate that a circuit breaker reset. The power state may depend on any policy applicable to the circuit or resource.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Breaker '`<1>`' reset."

1. *string*: The name or identifier of the circuit.
   - This argument shall contain a string that identifies or describes the `Circuit` resource or the location of the circuit breaker.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.17.3 BreakerTripped

Example: Breaker '7' has tripped.

Indicates that a circuit breaker tripped.

- This message shall indicate that a circuit breaker tripped due to an over-current condition or an electrical fault.

Version Added: v1.0

Severity: Critical

Resolution: Check the circuit and connected devices, and disconnect or replace any faulty devices.

Message and Arguments: "Breaker '`<1>`' has tripped."

1. *string*: The name or identifier of the circuit.
   - This argument shall contain a string that identifies or describes the `Circuit` resource or the location of the electrical fault.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical, ResourceEvent.ResourcePoweredOff

2.17.4 ChassisPowerLimitExceeded

Example: Power limit exceeded on Rack 'C8'.

Indicates that a chassis exceeded its power limit.

- This message shall indicate that a chassis exceeded its power limit. This condition may cause the chassis to throttle components or initiate a protective shutdown.

Version Added: v1.3

Severity: Warning

Resolution: Check additional diagnostic data if available. Review power limit configuration and reduce system load or increase the power budget if permitted.

Message and Arguments: "Power limit exceeded on `<1>` '`<2>`'."

1. *string*: The type or product category of the chassis.
   - This argument shall contain a string that describes the chassis type or the product category of the `Chassis` resource. For some resources, the value of `ChassisType`, such as `Rack`, may provide an appropriate description, while other resources, such as servers with `ChassisType` values of `RackMount`, `StandAlone`, `Blade`, or `Sled` could all be better described as a `Server`.
2. *string*: The name or identifier for the chassis.
   - This argument shall contain a string that identifies or describes the `Chassis` resource.

2.17.5 ChassisPowerLimitRestored

Example: Power limit restored on Rack 'C8'.

Indicates that a chassis returned below its power limit.

- This message shall indicate that a chassis returned below its power limit.

Version Added: v1.3

Severity: OK

Resolution: None.

Message and Arguments: "Power limit restored on `<1>` '`<2>`'."

1. *string*: The type or product category of the chassis.
   - This argument shall contain a string that describes the chassis type or the product category of the `Chassis` resource. For some resources, the value of `ChassisType`, such as `Rack`, may provide an appropriate description, while other resources, such as servers with `ChassisType` values of `RackMount`, `StandAlone`, `Blade`, or `Sled` could all be better described as a `Server`.
2. *string*: The name or identifier for the chassis.
   - This argument shall contain a string that identifies or describes the `Chassis` resource.

2.17.6 ChassisPoweredOff

Example: Rack 'A7' powered off.

Indicates that the power state of a chassis changed to off.

- This message shall indicate that the value of the `PowerState` property of a `Chassis` resource changed to `Off`.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "`<1>` '`<2>`' powered off."

1. *string*: The type or product category of the chassis.
   - This argument shall contain a string that describes the chassis type or the product category of the `Chassis` resource. For some resources, the value of `ChassisType`, such as `Rack`, may provide an appropriate description, while other resources, such as servers with `ChassisType` values of `RackMount`, `StandAlone`, `Blade`, or `Sled` could all be better described as a `Server`.
2. *string*: The name or identifier for the chassis.
   - This argument shall contain a string that identifies or describes the `Chassis` resource.

> **Note:** This message is a more specific version of the following messages: ResourceEvent.ResourcePoweredOff

2.17.7 ChassisPoweredOn

Example: Rack 'C8' powered on.

Indicates that the power state of a chassis changed to on.

- This message shall indicate that the value of the `PowerState` property of a `Chassis` resource changed to `On`.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "`<1>` '`<2>`' powered on."

1. *string*: The type or product category of the chassis.
   - This argument shall contain a string that describes the chassis type or the product category of the `Chassis` resource. For some resources, the value of `ChassisType`, such as `Rack`, may provide an appropriate description, while other resources, such as servers with `ChassisType` values of `RackMount`, `StandAlone`, `Blade`, or `Sled` could all be better described as a `Server`.
2. *string*: The name or identifier for the chassis.
   - This argument shall contain a string that identifies or describes the `Chassis` resource.

> **Note:** This message is a more specific version of the following messages: ResourceEvent.ResourcePoweredOn

2.17.8 CircuitPoweredOff

Example: Circuit 'Branch 4' powered off.

Indicates that a circuit was powered off.

- This message shall indicate that the value of the `PowerState` property of a `Circuit` resource changed to `Off`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Circuit '`<1>`' powered off."

1. *string*: The name or identifier of the circuit.
   - This argument shall contain a string that identifies or describes the `Circuit` resource.

> **Note:** This message is a more specific version of the following messages: ResourceEvent.ResourcePoweredOff

2.17.9 CircuitPoweredOn

Example: Circuit 'Branch 4' powered on.

Indicates that a circuit was powered on.

- This message shall indicate that the value of the `PowerState` property of a `Circuit` resource changed to `On`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Circuit '`<1>`' powered on."

1. *string*: The name or identifier of the circuit.
   - This argument shall contain a string that identifies or describes the `Circuit` resource.

> **Note:** This message is a more specific version of the following messages: ResourceEvent.ResourcePoweredOn

2.17.10 CurrentAboveLowerCriticalThreshold

Example: Current 'Branch-C' reading of 2.1 amperes is now above the 2 lower critical threshold but remains outside of normal range.

Indicates that a current reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a current reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the current violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerCriticalThreshold

2.17.11 CurrentAboveLowerFatalThreshold

Example: Current 'Branch-C' reading of 1.57 amperes is now above the 0.5 lower fatal threshold but remains outside of normal range.

Indicates that a current reading is no longer below the lower fatal threshold but is still outside of normal operating range.

- This message shall indicate that a current reading, previously below the fatal threshold, no longer violates the lower fatal threshold. However, the current violates at least one lower threshold and is not within the normal

operating range.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is now above the `<3>` lower fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerFatalThreshold

2.17.12 CurrentAboveUpperCautionThreshold

Example: Current 'Branch-C' reading of 25.6 amperes is above the 24.0 upper caution threshold.

Indicates that a current reading is above the upper caution threshold.

- This message shall indicate that a current reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.

2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.17.13 CurrentAboveUpperCriticalThreshold

Example: Current 'Branch-C' reading of 28.42 amperes is above the 28.0 upper critical threshold.

Indicates that a current reading is above the upper critical threshold.

- This message shall indicate that a current reading violates an upper critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.17.14 CurrentAboveUpperFatalThreshold

Example: Current 'Branch-C' reading of 30.28 amperes is above the 30 upper fatal threshold.

Indicates that a current reading is above the upper fatal threshold.

- This message shall indicate that a current reading violates an upper fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperFatal` property.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is above the `<3>` upper fatal threshold."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperFatalThreshold

2.17.15 CurrentBelowLowerCautionThreshold

Example: Current 'Branch-C' reading of 2.15 amperes is below the 3.0 lower caution threshold.

Indicates that a current reading is below the lower caution threshold.

- This message shall indicate that a current reading violates a lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of

the current reading.

2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.17.16 CurrentBelowLowerCriticalThreshold

Example: Current 'Branch-C' reading of 1.3 amperes is below the 2.0 lower critical threshold.

Indicates that a current reading is below the lower critical threshold.

- This message shall indicate that a current reading violates a lower critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.17.17 CurrentBelowLowerFatalThreshold

Example: Current 'Branch-C' reading of 0.29 amperes is below the 0.5 lower fatal threshold.

Indicates that a current reading is below the lower fatal threshold.

- This message shall indicate that a current reading violates a lower fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerFatal` property.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is below the `<3>` lower fatal threshold."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerFatalThreshold

2.17.18 CurrentBelowUpperCriticalThreshold

Example: Current 'Branch-C' reading of 27.3 amperes is now below the 28.0 upper critical threshold but remains outside of normal range.

Indicates that a current reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a current reading, previously above the critical threshold, no longer violates the upper critical threshold. However, the current violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.17.19 CurrentBelowUpperFatalThreshold

Example: Current 'Branch-C' reading of 28.4 amperes is now below the 30 upper fatal threshold but remains outside of normal range.

Indicates that a current reading is no longer above the upper fatal threshold but is still outside of normal operating range.

- This message shall indicate that a current reading, previously above the fatal threshold, no longer violates the upper fatal threshold. However, the current violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is now below the `<3>` upper fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperFatalThreshold

2.17.20 CurrentCritical

Example: Current 'Branch-C' reading of 28.3 amperes exceeds the critical level.

Indicates that a current reading exceeds an internal critical level.

- This message shall indicate that a current reading violates an internal critical level. This message shall be used for current-related properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The current-related properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes exceeds the critical level."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.17.21 CurrentNoLongerCritical

Example: Current 'Branch-C' reading of 27.2 amperes no longer exceeds the critical level.

Indicates that a current reading no longer exceeds an internal critical level but still exceeds an internal warning level.

- This message shall indicate that a current reading no longer violates an internal critical level but still exceeds an internal warning level. This message shall be used for current-related properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The current-related properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes no longer exceeds the critical level."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.17.22 CurrentNormal

Example: Current 'Branch-C' reading of 13.7 amperes is within normal operating range.

Indicates that a current reading is now within normal operating range.

- This message shall indicate that a current reading has returned to its normal operating range.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes is within normal operating range."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in ampere units.
   - This argument shall contain the current in ampere units.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.SensorReadingNormalRange

2.17.23 CurrentWarning

Example: Current 'Branch-C' reading of 25.7 amperes exceeds the warning level.

Indicates that a current reading exceeds an internal warning level.

- This message shall indicate that a current reading violates an internal warning level. This message shall be used for current-related properties or `Sensor` resources that implement boolean trip points for a caution limit, such as from hardware mechanisms with no threshold value. The current-related properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Current '`<1>`' reading of `<2>` amperes exceeds the warning level."

1. *string*: The name or identifier of the current reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the current reading.
2. *number*: The current in amperes.
   - This argument shall contain the current in amperes.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.17.24 FrequencyAboveLowerCriticalThreshold

Example: Frequency 'Mains' reading of 59.78 hertz is now above the 59.7 lower critical threshold but remains outside of normal range.

Indicates that a frequency reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a frequency reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the frequency violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency '`<1>`' reading of `<2>` hertz is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the frequency reading.

    - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz.
    - This argument shall contain the frequency in hertz.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerCriticalThreshold

2.17.25 FrequencyAboveUpperCautionThreshold

Example: Frequency 'Mains' reading of 60.18 hertz is above the 60.15 upper caution threshold.

Indicates that a frequency reading is above the upper caution threshold.

- This message shall indicate that a frequency reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency ' `<1>` ' reading of `<2>` hertz is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the frequency reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz units.
    - This argument shall contain the frequency in hertz units.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.17.26 FrequencyAboveUpperCriticalThreshold

Example: Frequency 'Mains' reading of 60.32 hertz is above the 60.30 upper critical threshold.

Indicates that a frequency reading is above the upper critical threshold.

- This message shall indicate that a frequency reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency ' `<1>` ' reading of `<2>` hertz is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the frequency reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz units.
   - This argument shall contain the frequency in hertz units.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.17.27 FrequencyBelowLowerCautionThreshold

Example: Frequency 'Mains' reading of 59.81 hertz is below the 59.85 lower caution threshold.

Indicates that a frequency reading is below the lower caution threshold.

- This message shall indicate that a frequency reading violates an lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency '`<1>`' reading of `<2>` hertz is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the frequency reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz units.
   - This argument shall contain the frequency in hertz units.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.17.28 FrequencyBelowLowerCriticalThreshold

Example: Frequency 'Mains' reading of 59.64 hertz is below the 59.70 lower critical threshold.

Indicates that a frequency reading is below the lower critical threshold.

- This message shall indicate that a frequency reading violates a lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency '`<1>`' reading of `<2>` hertz is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the frequency reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz.
   - This argument shall contain the frequency in hertz units.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.17.29 FrequencyBelowUpperCriticalThreshold

Example: Frequency 'Mains' reading of 60.18 hertz is now below the 60.30 upper critical threshold but remains outside of normal range.

Indicates that a frequency reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a frequency reading, previously above the critical threshold, no longer violates the upper critical threshold. However, the frequency violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency ' `<1>` ' reading of `<2>` hertz is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the frequency reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz.
   - This argument shall contain the frequency in hertz units.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.17.30 FrequencyCritical

Example: Frequency 'Mains' reading of 60.34 hertz exceeds the critical level.

Indicates that a frequency reading exceeds an internal critical level.

- This message shall indicate that a frequency reading violates an internal critical level. This message shall be used for frequency properties or `Sensor` resources that implement boolean trip points for a critical limit, such as

from hardware mechanisms with no threshold value. The frequency properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency ' `<1>` ' reading of `<2>` hertz exceeds the critical level."

1. *string*: The name or identifier of the frequency reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz.
   - This argument shall contain the frequency in hertz.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.17.31 FrequencyNoLongerCritical

Example: Frequency 'Mains' reading of 59.81 hertz no longer exceeds the critical level.

Indicates that a frequency reading no longer exceeds an internal critical level but still exceeds an internal warning level.

- This message shall indicate that a frequency reading no longer violates an internal critical level but still exceeds an internal warning level. This message shall be used for frequency properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The frequency properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency ' `<1>` ' reading of `<2>` hertz no longer exceeds the critical level."

1. *string*: The name or identifier of the frequency reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz.
   - This argument shall contain the frequency in hertz.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.17.32 FrequencyNormal

Example: Frequency 'Mains' reading of 60.07 hertz is within normal operating range.

Indicates that a frequency reading is now within normal operating range.

- This message shall indicate that a frequency reading has returned to its normal operating range.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Frequency '`<1>`' reading of `<2>` hertz is within normal operating range."

1. *string*: The name or identifier of the frequency reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The hertz frequency.
   - This argument shall contain the frequency in hertz units.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.SensorReadingNormalRange

2.17.33 FrequencyWarning

Example: Frequency 'Mains' reading of 60.19 hertz exceeds the warning level.

Indicates that a frequency reading exceeds an internal warning level.

- This message shall indicate that a frequency reading violates an internal warning level. This message shall be used for frequency properties or `Sensor` resources that implement boolean trip points for a warning limit, such as from hardware mechanisms with no threshold value. The frequency properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Frequency '`<1>`' reading of `<2>` hertz exceeds the warning level."

1. *string*: The name or identifier of the frequency reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the frequency reading.
2. *number*: The frequency in hertz.
   - This argument shall contain the frequency in hertz.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.17.34 LineInputPowerFault

Example: Line input power fault at 'AC Input B'.

Indicates a fault on an electrical power input.

- This message shall indicate that a power fault was detected at an electrical input. The equipment may not be able to produce output power.

Version Added: v1.0

Severity: Warning

Resolution: Check the electrical power input and connections.

Message and Arguments: "Line input power fault at '`<1>`'."

1. *string*: The name or identifier of the electrical input or equipment.
   - This argument shall contain a string that identifies or describes a `PowerSupply`, `Circuit`, `PowerDistribution`, or `Chassis` resource.

> **Note:** This message is a more specific version of the following messages: ResourceEvent.ResourceStatusChangedWarning

2.17.35 LineInputPowerRestored

Example: Line input power at 'AC Input B' was restored.

Indicates that an electrical power input was restored to normal operation.

- This message shall indicate that an electrical power input was repaired, reconnected, or otherwise returned to normal operation.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Line input power at '`<1>`' was restored."

1. *string*: The name or identifier of the electrical input or equipment.
   - This argument shall contain a string that identifies or describes a `PowerSupply`, `Circuit`, `PowerDistribution`, or `Chassis` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.17.36 LossOfInputPower

Example: Loss of input power at 'AC Input B'.

Indicates a loss of power at an electrical input.

- This message shall indicate that a loss of power was detected at an electrical input.

Version Added: v1.0

Severity: Warning

Resolution: Check the electrical power input and connections.

Message and Arguments: "Loss of input power at '`<1>`'."

1. *string*: The name or identifier of the electrical input or equipment.
   - This argument shall contain a string that identifies or describes a `PowerSupply`, `Circuit`, `PowerDistribution`, or `Chassis` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.17.37 OutletPoweredOff

Example: Outlet 'C6' powered off.

Indicates that an outlet was powered off.

- This message shall indicate that the value of the `PowerState` property of an `Outlet` resource changed to `Off`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Outlet '`<1>`' powered off."

1. *string*: The name or identifier of the outlet.
   - This argument shall contain a string that identifies or describes the `Outlet` resource.

> **Note:** This message is a more specific version of the following messages: ResourceEvent.ResourcePoweredOff

2.17.38 OutletPoweredOn

Example: Outlet 'C1' powered on.

Indicates that an outlet was powered on.

- This message shall indicate that the value of the `PowerState` property of an `Outlet` resource changed to `On`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Outlet '`<1>`' powered on."

1. *string*: The name or identifier of the outlet.
   - This argument shall contain a string that identifies or describes the `Outlet` resource.

> **Note:** This message is a more specific version of the following messages: ResourceEvent.ResourcePoweredOn

2.17.39 PowerAboveLowerCriticalThreshold

Example: Power 'Rack9' reading of 310.8 watts is now above the 300 lower critical threshold but remains outside of normal range.

Indicates that a power reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a power reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the power violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power ' `<1>` ' reading of `<2>` watts is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
   - This argument shall contain the power in watts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerCriticalThreshold

2.17.40 PowerAboveLowerFatalThreshold

Example: Power 'Rack9' reading of 227 watts is now above the 200 lower fatal threshold but remains outside of normal range.

Indicates that a power reading is no longer below the lower fatal threshold but is still outside of normal operating range.

- This message shall indicate that a power reading, previously below the fatal threshold, no longer violates the lower fatal threshold. However, the power violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power ' `<1>` ' reading of `<2>` watts is now above the `<3>` lower fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.

    - This argument shall contain the power in watts.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerFatalThreshold

2.17.41 PowerAboveUpperCautionThreshold

Example: Power 'Rack8' reading of 28192 watts is above the 28000 upper caution threshold.

Indicates that a power reading is above the upper caution threshold.

- This message shall indicate that a power reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power ' `<1>` ' reading of `<2>` watts is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the power reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
    - This argument shall contain the power in watts.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.17.42 PowerAboveUpperCriticalThreshold

Example: Power 'Rack8' reading of 31533 watts is above the 30000 upper critical threshold.

Indicates that a power reading is above the upper critical threshold.

- This message shall indicate that a power reading violates an upper critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
   - This argument shall contain the power in watts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.17.43 PowerAboveUpperFatalThreshold

Example: Power 'Rack8' reading of 32862 watts is above the 32000 upper fatal threshold.

Indicates that a power reading is above the upper fatal threshold.

- This message shall indicate that a power reading violates an upper fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperFatal` property.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts is above the `<3>` upper fatal threshold."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of

the power reading.

2. *number*: The power in watts.
   - This argument shall contain the power in watts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperFatalThreshold

2.17.44 PowerBelowLowerCautionThreshold

Example: Power 'Rack9' reading of 462.6 watts is below the 500 lower caution threshold.

Indicates that a power reading is below the lower caution threshold.

- This message shall indicate that a power reading violates a lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power ' `<1>` ' reading of `<2>` watts is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
   - This argument shall contain the power in watts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.17.45 PowerBelowLowerCriticalThreshold

Example: Power 'Rack9' reading of 179.8 watts is below the 200 lower critical threshold.

Indicates that a power reading is below the lower critical threshold.

- This message shall indicate that a power reading violates a lower critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
   - This argument shall contain the power in watts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.17.46 PowerBelowLowerFatalThreshold

Example: Power 'Rack9' reading of 162.3 watts is below the 200.0 lower fatal threshold.

Indicates that a power reading is below the lower fatal threshold.

- This message shall indicate that a power reading violates a lower fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerFatal` property.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts is below the `<3>` lower fatal threshold."

1. *string*: The name or identifier of the power reading.

    - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
    - This argument shall contain the power in watts.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerFatalThreshold

2.17.47 PowerBelowUpperCriticalThreshold

Example: Power 'Rack9' reading of 26812 watts is now below the 30000 upper critical threshold but remains outside of normal range.

Indicates that a power reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a power reading, previously above the critical threshold, no longer violates the upper critical threshold. However, the power violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power ' `<1>` ' reading of `<2>` watts is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the power reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
    - This argument shall contain the power in watts.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.17.48 PowerBelowUpperFatalThreshold

Example: Power 'Rack9' reading of 31419 watts is now below the 32000 upper fatal threshold but remains outside of normal range.

Indicates that a power reading is no longer above the upper fatal threshold but is still outside of normal operating range.

- This message shall indicate that a power reading, previously above the fatal threshold, no longer violates the upper fatal threshold. However, the power violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts is now below the `<3>` upper fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
   - This argument shall contain the power in watts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperFatalThreshold

2.17.49 PowerBrakeApplied

Example: Power brake applied due to 'LoadPercent'.

Indicates the power brake was applied to the equipment.

- This message shall indicate that the power brake feature was applied to the equipment. Power consumption has been reduced while the power brake is applied.

Version Added: v1.4

Severity: Warning

Resolution: Check the power load and temperature of the equipment.

Message and Arguments: "Power brake applied due to `<1>'."

1. *string*: The `PowerBrakeReason` that describes why the power brake was applied.
   - This argument shall contain the `PowerBrakeReason` value that identifies the reason the power brake was applied. The `PowerBrakeReason` values are defined in the `PowerSubsystem` schema.

2.17.50 PowerBrakeReleased

Example: Power brake released.

Indicates the power brake was released from the equipment.

- This message shall indicate that the power brake feature was released from the equipment. Power consumption is no longer restricted by the power brake.

Version Added: v1.4

Severity: OK

Resolution: None.

Message and Arguments: "Power brake released."

2.17.51 PowerCritical

Example: Power 'GPU#3' reading of 531.2 watts exceeds the critical level.

Indicates that a power reading exceeds an internal critical level.

- This message shall indicate that a power reading violates an internal critical level. This message shall be used for power properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The power properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts exceeds the critical level."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
   - This argument shall contain the power in watts.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.17.52 PowerDistributionGroupCritical

Example: Power distribution group 'Row 5' is in a critical state.

Indicates that a power distribution group has a critical status.

- This message shall indicate that a power distribution group is operating in a critical state.

Version Added: v1.2

Severity: Critical

Resolution: Check the electrical connections and power equipment related to this power distribution group.

Message and Arguments: "Power distribution group '`<1>`' is in a critical state."

1. *string*: The name or identifier of the power distribution group.
   - This argument shall contain a string that identifies or describes the group of `PowerDistribution` resources.

> **Note:** This message is a more specific version of the following messages: ResourceEvent.ResourceStatusChangedCritical

2.17.53 PowerDistributionGroupNormal

Example: Power distribution group 'Row 5' is operating normally.

Indicates that a power distribution group has returned to normal operations.

- This message shall indicate that a power distribution group has returned to a normal operating state.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "Power distribution group '`<1>`' is operating normally."

1. *string*: The name or identifier of the power distribution group.
   - This argument shall contain a string that identifies or describes the group of `PowerDistribution` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.17.54 PowerDistributionGroupWarning

Example: Power distribution group 'Row 5' is in a warning state.

Indicates that a power distribution group has a warning status.

- This message shall indicate that a power distribution group is operating with a warning status or in a non-redundant state due to a fault or a change in configuration.

Version Added: v1.2

Severity: Warning

Resolution: Check the electrical connections and power equipment related to this power distribution group.

Message and Arguments: "Power distribution group '`<1>`' is in a warning state."

1. *string*: The name or identifier of the power distribution group.
   - This argument shall contain a string that identifies or describes the group of `PowerDistribution` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.17.55 PowerNoLongerCritical

Example: Power 'GPU#3' reading of 371.5 watts no longer exceeds the critical level.

Indicates that a power reading no longer exceeds an internal critical level but still exceeds an internal warning level.

- This message shall indicate that a power reading no longer violates an internal critical level but still exceeds an internal warning level. This message shall be used for power properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The power properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts no longer exceeds the critical level."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
   - This argument shall contain the power in watts.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.17.56 PowerNormal

Example: Power 'GPU#3' reading of 285.2 watts is within normal operating range.

Indicates that a power reading is now within normal operating range.

- This message shall indicate that a power reading has returned to its normal operating range.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts is within normal operating range."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.
2. *number*: The power in watts.
   - This argument shall contain the power in watts.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.SensorReadingNormalRange

2.17.57 PowerSupplyFailed

Example: Power supply 'B2' has failed.

Indicates that a power supply has failed.

- This message shall indicate that a power supply has failed.

Version Added: v1.0

Severity: Critical

Resolution: Check the power supply hardware and replace any faulty component.

Message and Arguments: "Power supply '`<1>`' has failed."

1. *string*: The name or identifier of the power supply.
   - This argument shall contain a string that identifies or describes the `PowerSupply` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical

2.17.58 PowerSupplyGroupCritical

Example: Power supply group 'B Side' is in a critical state.

Indicates that a power supply group has a critical status.

- This message shall indicate that a power supply group is operating in a critical state.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "Power supply group '`<1>`' is in a critical state."

1. *string*: The name or identifier of the power supply group.
   - This argument shall contain a string that identifies or describes the group of `PowerSupply` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedCritical

2.17.59 PowerSupplyGroupNormal

Example: Power supply group 'B Side' is operating normally.

Indicates that a power supply group has returned to normal operations.

- This message shall indicate that a power supply group has returned to a normal operating state.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Power supply group '`<1>`' is operating normally."

1. *string*: The name or identifier of the power supply group.
   - This argument shall contain a string that identifies or describes the group of `PowerSupply` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.17.60 PowerSupplyGroupWarning

Example: Power supply group 'B Side' is in a warning state.

Indicates that a power supply group has a warning status.

- This message shall indicate that a power supply group is operating with a warning status or in a non-redundant state due to a fault or a change in configuration.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "Power supply group '`<1>`' is in a warning state."

1. *string*: The name or identifier of the power supply group.
   - This argument shall contain a string that identifies or describes the group of `PowerSupply` resources.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.17.61 PowerSupplyInserted

Example: Power supply 'B4' was inserted.

Indicates that a power supply was inserted or installed.

- This message shall indicate that a power supply was inserted or installed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Power supply '`<1>`' was inserted."

1. *string*: The name or identifier of the power supply.
   - This argument shall contain a string that identifies or describes the `PowerSupply` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStateChanged

2.17.62 PowerSupplyPredictiveFailure

Example: Power supply 'A5' has a predicted failure condition.

Indicates that the power supply predicted a future failure condition.

- This message shall indicate a power supply predicted a future failure condition, but the power supply remains functional.

Version Added: v1.0

Severity: Warning

Resolution: Check the power supply hardware and replace any faulty component.

Message and Arguments: "Power supply '`<1>`' has a predicted failure condition."

1. *string*: The name or identifier of the power supply.
   - This argument shall contain a string that identifies or describes the `PowerSupply` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.17.63 PowerSupplyRemoved

Example: Power supply 'B4' was removed.

Indicates that a power supply was removed.

- This message shall indicate that a power supply was removed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Power supply '`<1>`' was removed."

1. *string*: The name or identifier of the power supply.
   - This argument shall contain a string that identifies or describes the `PowerSupply` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStateChanged

2.17.64 PowerSupplyRestored

Example: Power supply 'B2' was restored.

Indicates that a power supply was repaired or restored to normal operation.

- This message shall indicate that a power supply was repaired, replaced, or otherwise returned to normal operation.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Power supply '`<1>`' was restored."

1. *string*: The name or identifier of the power supply.
   - This argument shall contain a string that identifies or describes the `PowerSupply` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedOK

2.17.65 PowerSupplyWarning

Example: Power supply 'B2' has a warning condition.

Indicates that a power supply has a warning condition.

- This message shall indicate that a power supply has a warning condition but continues to supply output power.

Version Added: v1.0

Severity: Warning

Resolution: Check the power supply hardware and replace any faulty component.

Message and Arguments: "Power supply '`<1>`' has a warning condition."

1. *string*: The name or identifier of the power supply.
   - This argument shall contain a string that identifies or describes the `PowerSupply` resource.

> **Note:** This message is a more specific version of the following messages:
> ResourceEvent.ResourceStatusChangedWarning

2.17.66 PowerWarning

Example: Power 'GPU#3' reading of 468.7 watts exceeds the warning level.

Indicates that a power reading exceeds an internal warning level.

- This message shall indicate that a power reading violates an internal warning level. This message shall be used for power properties or `Sensor` resources that implement boolean trip points for a warning limit, such as from hardware mechanisms with no threshold value. The power properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Power '`<1>`' reading of `<2>` watts exceeds the warning level."

1. *string*: The name or identifier of the power reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the power reading.

2. *number*: The power in watts.
    - This argument shall contain the power in watts.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.17.67 VoltageAboveLowerCriticalThreshold

Example: Voltage 'Mains-B' reading of 219.8 volts is now above the 216 lower critical threshold but remains outside of normal range.

Indicates that a voltage reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall indicate that a voltage reading, previously below the lower critical threshold, no longer violates the lower critical threshold. However, the voltage violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage ' `<1>` ' reading of `<2>` volts is now above the `<3>` lower critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the voltage reading.
    - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
    - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
    - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingAboveLowerCriticalThreshold

2.17.68 VoltageAboveLowerFatalThreshold

Example: Voltage 'Mains-B' reading of 209.5 volts is now above the 204 lower fatal threshold but remains outside of normal range.

Indicates that a voltage reading is no longer below the lower fatal threshold but is still outside of normal operating range.

- This message shall indicate that a voltage reading, previously below the fatal threshold, no longer violates the lower fatal threshold. However, the voltage violates at least one lower threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage '`<1>`' reading of `<2>` volts is now above the `<3>` lower fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveLowerFatalThreshold

2.17.69 VoltageAboveUpperCautionThreshold

Example: Voltage 'Mains-B' reading of 255.9 volts is above the 252 upper caution threshold.

Indicates that a voltage reading is above the upper caution threshold.

- This message shall indicate that a voltage reading violates an upper caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage '`<1>`' reading of `<2>` volts is above the `<3>` upper caution threshold."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCaution` or `UpperCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCautionThreshold

2.17.70 VoltageAboveUpperCriticalThreshold

Example: Voltage 'Mains-B' reading of 266.1 volts is above the 264 upper critical threshold.

Indicates that a voltage reading is above the upper critical threshold.

- This message shall indicate that a voltage reading violates an upper critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage ' `<1>` ' reading of `<2>` volts is above the `<3>` upper critical threshold."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperCriticalThreshold

2.17.71 VoltageAboveUpperFatalThreshold

Example: Voltage 'Mains-B' reading of 287.6 volts is above the 280 upper fatal threshold.

Indicates that a voltage reading is above the upper fatal threshold.

- This message shall indicate that a voltage reading violates an upper fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is above the value of the `Reading` property within the `UpperFatal` property.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage ' `<1>` ' reading of `<2>` volts is above the `<3>` upper fatal threshold."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingAboveUpperFatalThreshold

2.17.72 VoltageBelowLowerCautionThreshold

Example: Voltage 'Mains-B' reading of 223.6 volts is below the 228 lower caution threshold.

Indicates that a voltage reading is below the lower caution threshold.

- This message shall indicate that a voltage reading violates a lower caution threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage '`<1>`' reading of `<2>` volts is below the `<3>` lower caution threshold."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCaution` or `LowerCautionUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCautionThreshold

2.17.73 VoltageBelowLowerCriticalThreshold

Example: Voltage 'Mains-B' reading of 213.7 volts is below the 216 lower critical threshold.

Indicates that a voltage reading is below the lower critical threshold.

- This message shall indicate that a voltage reading violates a lower critical threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage '`<1>`' reading of `<2>` volts is below the `<3>` lower critical threshold."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerCritical` or `LowerCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerCriticalThreshold

2.17.74 VoltageBelowLowerFatalThreshold

Example: Voltage 'Mains-B' reading of 198.2 volts is below the 204 lower fatal threshold.

Indicates that a voltage reading is below the lower fatal threshold.

- This message shall indicate that a voltage reading violates a lower fatal threshold. If the reading originates from a `Sensor` resource, the message shall indicate that the value of the `Reading` property is below the value of the `Reading` property within the `LowerFatal` property.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage ' `<1>` ' reading of `<2>` volts is below the `<3>` lower fatal threshold."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `LowerFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowLowerFatalThreshold

2.17.75 VoltageBelowUpperCriticalThreshold

Example: Voltage 'Mains-B' reading of 261.4 volts is now below the 264 upper critical threshold but remains outside of normal range.

Indicates that a voltage reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall indicate that a voltage reading, previously above the critical threshold, no longer violates the upper critical threshold. However, the voltage violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage '`<1>`' reading of `<2>` volts is now below the `<3>` upper critical threshold but remains outside of normal range."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperCritical` or `UpperCriticalUser` properties of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingBelowUpperCriticalThreshold

2.17.76 VoltageBelowUpperFatalThreshold

Example: Voltage 'Mains-B' reading of 266.8 volts is now below the 280 upper fatal threshold but remains outside of normal range.

Indicates that a voltage reading is no longer above the upper fatal threshold but is still outside of normal operating range.

- This message shall indicate that a voltage reading, previously above the fatal threshold, no longer violates the upper fatal threshold. However, the voltage violates at least one upper threshold and is not within the normal operating range.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage '`<1>`' reading of `<2>` volts is now below the `<3>` upper fatal threshold but remains outside of normal range."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of

the voltage reading.

2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.
3. *number*: The threshold value.
   - This argument shall contain the value of the `Reading` property within the `UpperFatal` property of the `Sensor` resource.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingBelowUpperFatalThreshold

2.17.77 VoltageCritical

Example: Voltage 'Mains-B' reading of 267.1 volts exceeds the critical level.

Indicates that a voltage reading exceeds an internal critical level.

- This message shall indicate that a voltage reading violates an internal critical level. This message shall be used for voltage properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The voltage properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage ' `<1>` ' reading of `<2>` volts exceeds the critical level."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingCritical

2.17.78 VoltageNoLongerCritical

Example: Voltage 'Mains-B' reading of 257.3 volts no longer exceeds the critical level.

Indicates that a voltage reading no longer exceeds an internal critical level but still exceeds an internal warning level.

- This message shall indicate that a voltage reading no longer violates an internal critical level but still violates an

internal warning level. This message shall be used for voltage properties or `Sensor` resources that implement boolean trip points for a critical limit, such as from hardware mechanisms with no threshold value. The voltage properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage '`<1>`' reading of `<2>` volts no longer exceeds the critical level."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.ReadingNoLongerCritical

2.17.79 VoltageNormal

Example: Voltage 'Mains-B' reading of 244.8 volts is within normal operating range.

Indicates that a voltage reading is now within normal operating range.

- This message shall indicate that a voltage reading has returned to its normal operating range.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Voltage '`<1>`' reading of `<2>` volts is within normal operating range."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.

> **Note:** This message is a more specific version of the following messages:
> SensorEvent.SensorReadingNormalRange

2.17.80 VoltageWarning

Example: Voltage 'Mains-B' reading of 254.1 volts exceeds the warning level.

Indicates that a voltage reading exceeds an internal warning level.

- This message shall indicate that a voltage reading violates an internal warning level. This message shall be used for voltage properties or `Sensor` resources that implement boolean trip points for a caution limit, such as from hardware mechanisms with no threshold value. The voltage properties or `Sensor` resources may contain readable thresholds that coexist with this unspecified trip point.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the resource listed in OriginOfCondition.

Message and Arguments: "Voltage ' `<1>` ' reading of `<2>` volts exceeds the warning level."

1. *string*: The name or identifier of the voltage reading.
   - This argument shall contain a string that identifies or describes the location or physical context of the voltage reading.
2. *number*: The voltage in volts.
   - This argument shall contain the voltage in volts.

> **Note:** This message is a more specific version of the following messages: SensorEvent.ReadingWarning

2.18 ResourceEvent 1.5.0

Version	*v1.5*	*v1.4*	*v1.3*	*v1.2*	*v1.1*	*v1.0*
Release	2026.1	2024.1	2024.1	2021.4	2021.3	2018.2

This registry defines the messages to use for resource events.

Message	Severity	Description
AggregationSourceDiscovered *(v1.2+)*	OK	Indicates that a new aggregation source was discovered.
LicenseAdded *(deprecated v1.1)*	OK	Indicates that a license was added.

Message	Severity	Description
LicenseChanged *(deprecated v1.1)*	Warning	Indicates that a license has changed.
LicenseExpired *(deprecated v1.1)*	Warning	Indicates that a license has expired.
PropertyValueModifiedByClient *(v1.4+)*	OK	Indicates that the value of a property was modified by a client.
ResourceAvailable *(v1.5+)*	OK	Indicates that a resource that was previously unavailable due to errors is now available.
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 when only the state has changed.
ResourceCreated	OK	Indicates that all conditions of a successful creation operation were met.
ResourceErrorThresholdCleared	OK	Indicates that a specified resource property has cleared its error threshold. Examples of this are drive I/O errors or network link errors.
ResourceErrorThresholdExceeded	Critical	Indicates that a specified resource property has exceeded its error threshold. Examples of this are drive I/O errors or network link errors.
ResourceErrorsCorrected	OK	Indicates that a specified resource property has corrected errors. Examples of this are drive I/O errors or network link errors.
ResourceErrorsDetected	Warning	Indicates that a specified resource property has detected errors. Examples of this are drive I/O errors or network link errors.
ResourcePaused *(v1.1+)*	OK	Indicates that the power state of a resource has changed to paused.
ResourcePoweredOff *(v1.1+)*	OK	Indicates that the power state of a resource has changed to powered off.
ResourcePoweredOn *(v1.1+)*	OK	Indicates that the power state of a resource has changed to powered on.
ResourcePoweringOff *(v1.1+)*	OK	Indicates that the power state of a resource has changed to powering off.
ResourcePoweringOn *(v1.1+)*	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 were 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 *(v1.1+)*	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. This might also be referred to as the health being restored.

Message	Severity	Description
ResourceStatusChangedWarning	Warning	Indicates that the health of a resource has changed to Warning. This might also be referred to as the health being degraded.
ResourceUnavailable *(v1.5+)*	Warning	Indicates that a resource encountered errors and is now unavailable.
ResourceVersionIncompatible	Warning	Indicates that an incompatible version of software was detected. Examples of this are after a component or system-level software update.
ResourceWarningThresholdCleared	OK	Indicates that a specified resource property has cleared its warning threshold. Examples of this are 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 of this are drive I/O errors or network link errors. Suggested resolution may be provided as OEM data.
TestMessage	OK	A test message used to validate event delivery mechanisms.
URIForResourceChanged	OK	Indicates that the URI for a resource has changed. Examples of this are physical component replacement or redistribution.

2.18.1 AggregationSourceDiscovered

Example: An aggregation source of connection method 'HTTPS' located at 'https://Contoso.com/redfish/v1' was discovered.

Indicates that a new aggregation source was discovered.

- This message shall indicate an aggregation source was discovered in the network or host. The `OriginOfCondition` property shall contain a link to the matching `ConnectionMethod` resource, if known.

Version Added: v1.2

Severity: OK

Resolution: The aggregation source is available to the service and can be identified using the identified connection method.

Message and Arguments: "An aggregation source of connection method ' `<1>` ' located at ' `<2>` ' was 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.

2.18.2 LicenseAdded

Example: A license for 'Feature1' was added. The following message was returned: 'The license for Feature1 is valid until 2025-01-01.'

Indicates that a license was added.

- This message shall indicate that a license was added.

Deprecated in v1.1 and later. This message was deprecated in favor of `LicenseAdded` *in the License Message Registry.*

Version Added: v1.0

Severity: OK

Resolution: See vendor specific instructions for specific actions.

Message and Arguments: "A license for '`<1>`' was added. The following message was returned: '`<2>`'."

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.

2.18.3 LicenseChanged

Example: A license for 'Feature1' has changed. The following message was returned: 'The license for Feature1 was renewed and is now valid until 2025-01-01.'

Indicates that a license has changed.

- This message shall indicate that a license has changed.

Deprecated in v1.1 and later. This message was deprecated in favor of `ResourceChanged`*.*

Version Added: v1.0

Severity: Warning

Resolution: See vendor specific instructions for specific actions.

Message and Arguments: "A license for '`<1>`' has changed. The following message was returned: '`<2>`'."

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.

2.18.4 LicenseExpired

Example: A license for 'Feature1' has expired. The following message was returned: 'The license for Feature1 expired on 2024-01-01.'

Indicates that a license has expired.

- This message shall indicate that a license has expired.

Deprecated in v1.1 and later. This message was deprecated in favor of `LicenseExpired` in the License Message Registry.

Version Added: v1.0

Severity: Warning

Resolution: See vendor specific instructions for specific actions.

Message and Arguments: "A license for '`<1>`' has expired. The following message was returned: '`<2>`'."

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.

2.18.5 PropertyValueModifiedByClient

Example: The property 'DHCPEnabled' was assigned the value 'true' due to modification by a client.

Indicates that the value of a property was modified by a client.

- This message shall indicate that the value of a property was modified by a client. For example, DHCPEnabled property was assigned the value `true` by a client.

Version Added: v1.4

Severity: OK

Resolution: No resolution is required.

Message and Arguments: "The property '`<1>`' was assigned the value '`<2>`' due to modification by a client."

1. *string*: The name of the property.
   - This argument shall contain the name of the property. The value should follow RFC6901-defined JSON pointer notation rules.
2. *string*: The value assigned for property.
   - This argument shall contain the value assigned for property.

2.18.6 ResourceAvailable

Example: Resource 'Drive1' is available.

Indicates that a resource that was previously unavailable due to errors is now available.

- This message shall indicate that a resource that was previously unavailable due to errors is now available and the service is able to provide responses for the resource.

Version Added: v1.5

Severity: OK

Resolution: None

Message and Arguments: "Resource '`<1>`' is available."

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

2.18.7 ResourceChanged

Example: One or more resource properties have changed.

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 when only the state has changed.

- This message shall indicate that one or more resource properties have changed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "One or more resource properties have changed."

2.18.8 ResourceCreated

Example: The resource was created successfully.

Indicates that all conditions of a successful creation operation were met.

- This message shall indicate that all conditions of a successful creation operation were met.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource was created successfully."

2.18.9 ResourceErrorThresholdCleared

Example: The resource property 'UncorrectableIOWriteErrorCount' has cleared the error threshold of value 5.

Indicates that a specified resource property has cleared its error threshold. Examples of this are drive I/O errors or network link errors.

- This message shall indicate that a specified resource property has cleared its error threshold.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource property '`<1>`' has cleared the error threshold of value `<2>`."

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.

2.18.10 ResourceErrorThresholdExceeded

Example: The resource property 'UncorrectableIOWriteErrorCount' has exceeded error threshold of value 5.

Indicates that a specified resource property has exceeded its error threshold. Examples of this are drive I/O errors or network link errors.

- This message shall indicate that a specified resource property has exceeded its error threshold.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "The resource property '`<1>`' has exceeded error threshold of value `<2>`."

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.

2.18.11 ResourceErrorsCorrected

Example: The resource property 'UncorrectableIOWriteErrorCount' has corrected errors of type 'I/O Error'.

Indicates that a specified resource property has corrected errors. Examples of this are drive I/O errors or network link errors.

- This message shall indicate that a specified resource property has corrected errors.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource property '`<1>`' has corrected errors of type '`<2>`'."

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.

2.18.12 ResourceErrorsDetected

Example: The resource property 'UncorrectableIOWriteErrorCount' has detected errors of type 'I/O Error'.

Indicates that a specified resource property has detected errors. Examples of this are drive I/O errors or network link errors.

- This message shall indicate that a specified resource property has detected errors.

Version Added: v1.0

Severity: Warning

Resolution: Resolution is dependent upon error type.

Message and Arguments: "The resource property '`<1>`' has detected errors of type '`<2>`'."

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.

2.18.13 ResourcePaused

Example: The resource 'Drive1' was paused.

Indicates that the power state of a resource has changed to paused.

- This message shall indicate that the value of the PowerState property of a resource has changed to `Paused`.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The resource '`<1>`' was paused."

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

2.18.14 ResourcePoweredOff

Example: The resource 'Drive1' has powered off.

Indicates that the power state of a resource has changed to powered off.

- This message shall indicate that the value of the PowerState property of a resource has changed to `Off`.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The resource '`<1>`' has powered off."

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

2.18.15 ResourcePoweredOn

Example: The resource 'Drive1' has powered on.

Indicates that the power state of a resource has changed to powered on.

- This message shall indicate that the value of the PowerState property of a resource has changed to `On`.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The resource '`<1>`' has powered on."

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

2.18.16 ResourcePoweringOff

Example: The resource 'Drive1' is powering off.

Indicates that the power state of a resource has changed to powering off.

- This message shall 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: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The resource ' `<1>` ' is powering off."

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

2.18.17 ResourcePoweringOn

Example: The resource 'Drive1' is powering on.

Indicates that the power state of a resource has changed to powering on.

- This message shall 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: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The resource ' `<1>` ' is powering on."

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

2.18.18 ResourceRemoved

Example: The resource was removed successfully.

Indicates that all conditions of a successful remove operation were met.

- This message shall indicate that all conditions of a successful remove operation were met.

Version Added: v1.0

Severity: OK

Resolution: None.

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

2.18.19 ResourceSelfTestCompleted

Example: A self-test has completed.

Indicates that a self-test has completed.

- This message shall indicate that a self-test has completed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A self-test has completed."

2.18.20 ResourceSelfTestFailed

Example: A self-test has failed. The following message was returned: 'Drive1 failed the self-test.'

Indicates that a self-test has failed. Suggested resolution may be provided as OEM data.

- This message shall indicate that a self-test has failed.

Version Added: v1.0

Severity: Critical

Resolution: See vendor specific instructions for specific actions.

Message and Arguments: "A self-test has failed. The following message was returned: '`<1>`'."

1. *string*: The self-test error message.
    - This argument shall contain the error message received as a result from the self-test.

2.18.21 ResourceStateChanged

Example: The state of resource 'Drive1' has changed to Enabled.

Indicates that the state of a resource has changed.

- This message shall indicate that the value of the State property of a resource has changed.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The state of resource '`<1>`' has changed to `<2>`."

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.

2.18.22 ResourceStatusChangedCritical

Example: The health of resource 'Drive1' has changed to Critical.

Indicates that the health of a resource has changed to Critical.

- This message shall indicate that the health of a resource has changed to Critical.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "The health of resource '`<1>`' has changed to `<2>`."

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.

2.18.23 ResourceStatusChangedOK

Example: The health of resource 'Drive1' has changed to OK.

Indicates that the health of a resource has changed to OK. This might also be referred to as the health being restored.

- This message shall indicate that the health of a resource has changed to OK.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The health of resource '`<1>`' has changed to `<2>`."

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.

2.18.24 ResourceStatusChangedWarning

Example: The health of resource 'Drive1' has changed to Warning.

Indicates that the health of a resource has changed to Warning. This might also be referred to as the health being degraded.

- This message shall indicate that the health of a resource has changed to Warning.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The health of resource '`<1>`' has changed to `<2>`."

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.

2.18.25 ResourceUnavailable

Example: Resource 'Drive1' is unavailable due to errors.

Indicates that a resource encountered errors and is now unavailable.

- This message shall indicate that a resource encountered errors and the service is unable to provide responses for the resource.

Version Added: v1.5

Severity: Warning

Resolution: Check additional diagnostic data if available

Message and Arguments: "Resource '`<1>`' is unavailable due to errors."

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

2.18.26 ResourceVersionIncompatible

Example: An incompatible version of software 'Firmware1.0' was detected.

Indicates that an incompatible version of software was detected. Examples of this are after a component or system-level software update.

- This message shall indicate that an incompatible version of software was detected.

Version Added: v1.0

Severity: Warning

Resolution: Compare the version of the resource with the compatible version of the software.

Message and Arguments: "An incompatible version of software '`<1>`' was 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.

2.18.27 ResourceWarningThresholdCleared

Example: The resource property 'UncorrectableIOWriteErrorCount' has cleared the warning threshold of value 5.

Indicates that a specified resource property has cleared its warning threshold. Examples of this are drive I/O errors or network link errors. Suggested resolution may be provided as OEM data.

- This message shall indicate that a specified resource property has cleared its warning threshold.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The resource property ' `<1>` ' has cleared the warning threshold of value `<2>` ."

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.

2.18.28 ResourceWarningThresholdExceeded

Example: The resource property 'UncorrectableIOWriteErrorCount' has exceeded its warning threshold of value 5.

Indicates that a specified resource property has exceeded its warning threshold. Examples of this are drive I/O errors or network link errors. Suggested resolution may be provided as OEM data.

- This message shall indicate that a specified resource property has exceeded its warning threshold.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The resource property ' `<1>` ' has exceeded its warning threshold of value `<2>` ."

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.

2.18.29 TestMessage

Example: Test message.

A test message used to validate event delivery mechanisms.

- This message shall indicate a test of an event delivery mechanism, such as the creation of a LogEntry resource or an event transmission.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Test message."

2.18.30 URIForResourceChanged

Example: The URI for the resource has changed.

Indicates that the URI for a resource has changed. Examples of this are physical component replacement or redistribution.

- This message shall indicate that the URI for a resource has changed.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The URI for the resource has changed."

2.19 SensorEvent 1.2.0

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.1	2025.3	2022.2

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.
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.

Message	Severity	Description
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 still exceeds an internal warning level.
ReadingWarning	Warning	Indicates that a sensor reading exceeds an internal warning level.
SensorConnected *(v1.1+)*	OK	Indicates that an external sensor was connected.
SensorDisconnected *(v1.1+)*	Warning	Indicates that an external sensor was disconnected.
SensorFailure	Warning	Indicates that the service cannot communicate with a sensor or detected a failure.
SensorReadingNormalRange	OK	Indicates that a sensor reading is now within normal operating range.
SensorRestored	OK	Indicates that a sensor was repaired or communications were restored. It may also indicate that the service is receiving valid data from a sensor.

2.19.1 InvalidSensorReading

Example: Invalid reading received from sensor 'CPU1 Temperature'.

Indicates that the service received an invalid reading from a sensor.

- This message shall indicate that the service received invalid data while attempting to update the value of the `Reading` property in a `Sensor` resource.

Version Added: v1.0

Severity: Warning

Resolution: Check the sensor hardware or connection.

Message and Arguments: "Invalid reading received from sensor '`<1>`'."

1. *string*: The name or identifier of the sensor.
    - This argument shall contain a string that identifies or describes the `Sensor` resource.

2.19.2 ReadingAboveLowerCriticalThreshold

Example: Sensor 'CPU1 Temperature' reading of 15 (Degrees Celsius) is now above the 10 lower critical threshold but remains outside of normal range.

Indicates that a sensor reading is no longer below the lower critical threshold but is still outside of normal operating range.

- This message shall 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: v1.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor ' `<1>` ' reading of `<2>` (`<3>`) is now above the `<4>` 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.

2.19.3 ReadingAboveLowerFatalThreshold

Example: Sensor 'CPU1 Temperature' reading of 15 (Degrees Celsius) is now above the 10 lower fatal threshold but remains outside of normal range.

Indicates that a sensor reading is no longer below the lower fatal threshold but is still outside of normal operating range.

- This message shall 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: v1.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) is now above the `<4>` 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.

2.19.4 ReadingAboveUpperCautionThreshold

Example: Sensor 'CPU1 Temperature' reading of 85 (Degrees Celsius) is above the 80 upper caution threshold.

Indicates that a sensor reading is above the upper caution threshold.

- This message shall 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: v1.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<1>' reading of <2> (<3>) is above the <4> 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.

2.19.5 ReadingAboveUpperCriticalThreshold

Example: Sensor 'CPU1 Temperature' reading of 95 (Degrees Celsius) is above the 90 upper critical threshold.

Indicates that a sensor reading is above the upper critical threshold.

- This message shall 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: v1.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '<1>' reading of <2> (<3>) is above the <4> 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.

2.19.6 ReadingAboveUpperFatalThreshold

Example: Sensor 'CPU1 Temperature' reading of 100 (Degrees Celsius) is above the 95 upper fatal threshold.

Indicates that a sensor reading is above the upper fatal threshold.

- This message shall 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: v1.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor ' `<1>` ' reading of `<2>` (`<3>`) is above the `<4>` 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.

2.19.7 ReadingBelowLowerCautionThreshold

Example: Sensor 'CPU1 Temperature' reading of 15 (Degrees Celsius) is below the 20 lower caution threshold.

Indicates that a sensor reading is below the lower caution threshold.

- This message shall 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: v1.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) is below the `<4>` 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.

2.19.8 ReadingBelowLowerCriticalThreshold

Example: Sensor 'CPU1 Temperature' reading of 5 (Degrees Celsius) is below the 10 lower critical threshold.

Indicates that a sensor reading is below the lower critical threshold.

- This message shall 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: v1.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) is below the `<4>` 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.

2.19.9 ReadingBelowLowerFatalThreshold

Example: Sensor 'CPU1 Temperature' reading of 0 (Degrees Celsius) is below the 5 lower fatal threshold.

Indicates that a sensor reading is below the lower fatal threshold.

- This message shall 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: v1.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor ' `<1>` ' reading of `<2>` (`<3>`) is below the `<4>` 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.

2.19.10 ReadingBelowUpperCriticalThreshold

Example: Sensor 'CPU1 Temperature' reading of 85 (Degrees Celsius) is now below the 90 upper critical threshold but remains outside of normal range.

Indicates that a sensor reading is no longer above the upper critical threshold but is still outside of normal operating range.

- This message shall 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: v1.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) is now below the `<4>` 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.

2.19.11 ReadingBelowUpperFatalThreshold

Example: Sensor 'CPU1 Temperature' reading of 90 (Degrees Celsius) is now below the 95 upper fatal threshold but remains outside of normal range.

Indicates that a sensor reading is no longer above the upper fatal threshold but is still outside of normal operating range.

- This message shall 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: v1.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) is now below the `<4>` 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.

2.19.12 ReadingCritical

Example: Sensor 'CPU1 Temperature' reading of 95 (Degrees Celsius) exceeds the 90 upper critical threshold.

Indicates that a sensor reading exceeds an internal critical level.

- This message shall 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: v1.0

Severity: Critical

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) 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`.

2.19.13 ReadingNoLongerCritical

Example: Sensor 'CPU1 Temperature' reading of 85 (Degrees Celsius) is now below the 90 upper critical threshold but remains outside of normal range.

Indicates that a sensor reading no longer exceeds an internal critical level but still exceeds an internal warning level.

- This message shall 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: v1.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) 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`.

2.19.14 ReadingWarning

Example: Sensor 'CPU1 Temperature' reading of 85 (Degrees Celsius) exceeds the 80 upper caution threshold.

Indicates that a sensor reading exceeds an internal warning level.

- This message shall 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: v1.0

Severity: Warning

Resolution: Check the condition of the resources listed in RelatedItem.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) 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`.

2.19.15 SensorConnected

Example: Sensor 'CPU1 Temperature' was connected.

Indicates that an external sensor was connected.

- This message shall indicate that an external sensor was connected.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Sensor '`<1>`' was connected."

1. *string*: The name or identifier of the sensor.
    - This argument shall contain a string that identifies or describes the `Sensor` resource.

2.19.16 SensorDisconnected

Example: Sensor 'CPU1 Temperature' was disconnected.

Indicates that an external sensor was disconnected.

- This message shall indicate that an external sensor was disconnected.

Version Added: v1.1

Severity: Warning

Resolution: Check the sensor connection or re-connect the sensor.

Message and Arguments: "Sensor '`<1>`' was disconnected."

1. *string*: The name or identifier of the sensor.
    - This argument shall contain a string that identifies or describes the `Sensor` resource.

2.19.17 SensorFailure

Example: Sensor 'CPU1 Temperature' failed.

Indicates that the service cannot communicate with a sensor or detected a failure.

- This message shall indicate that the service is unable to communicate with a sensor or otherwise detected a fault condition with the sensor.

Version Added: v1.0

Severity: Warning

Resolution: Check the sensor hardware or connection.

Message and Arguments: "Sensor '`<1>`' failed."

1. *string*: The name or identifier of the sensor.
   - This argument shall contain a string that identifies or describes the `Sensor` resource.

2.19.18 SensorReadingNormalRange

Example: Sensor 'CPU1 Temperature' reading of 75 (Degrees Celsius) is now within the normal operating range.

Indicates that a sensor reading is now within normal operating range.

- This message shall indicate that the value of the `Reading` property within a `Sensor` resource has returned to its normal operating range.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Sensor '`<1>`' reading of `<2>` (`<3>`) 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`.

2.19.19 SensorRestored

Example: Sensor 'CPU1 Temperature' was restored.

Indicates that a sensor was repaired or communications were restored. It may also indicate that the service is receiving valid data from a sensor.

- This message shall indicate that the service is able to communicate with a sensor that had previously failed, or the sensor was repaired or replaced.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Sensor '`<1>`' was restored."

1. *string*: The name or identifier of the sensor.
    - This argument shall contain a string that identifies or describes the `Sensor` resource.

2.20 ServiceCommunications 1.2.0

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.2	2026.1	2024.4

This registry defines messages to be used by a service to indicate communication problems with clients or other services.

Message	Severity	Description
CertificateVerificationFailure *(v1.2+)*	Critical	Indicates that the certificate of a remote service could not be verified, such as if the certificate is signed by an unknown certificate authority.
ConnectionRefused	Critical	Indicates that a connection to the remote service was refused.
ConnectionTimeout	Critical	Indicates that a connection to the remote service timed out.
HTTPFailure	Critical	Indicates that an HTTP failure code was returned.

Message	Severity	Description
NameResolutionFailure	Critical	Indicates that DNS or other name resolution failed on the indicated name.
SecureCommunicationFailure *(v1.2+)*	Critical	Indicates that a secure communication connection could not be established, such as during TLS handshaking.

2.20.1 CertificateVerificationFailure

Example: The certificate from 'https://contoso.com' could not be verified and is untrusted.

Indicates that the certificate of a remote service could not be verified, such as if the certificate is signed by an unknown certificate authority.

- This message shall indicate that the certificate of a remote service could not be verified.

Version Added: v1.2

Severity: Critical

Resolution: Check the certificate settings of the service and the remote service.

Message and Arguments: "The certificate from '`<1>`' could not be verified and is untrusted."

1. *string*: The network name of the service that provided a certificate that could not be verified.
   - This argument shall contain the network name of the remote service that provided a certificate that could not be verified.

2.20.2 ConnectionRefused

Example: The connection to 'Contoso.com:443' was refused.

Indicates that a connection to the remote service was refused.

- This message shall indicate that a connection to the remote service was refused.

Version Added: v1.0

Severity: Critical

Resolution: Check the settings on the remote service or the port settings.

Message and Arguments: "The connection to '`<1>`:`<2>`' was refused."

1. *string*: The network name of the service that refused the connection.
    - This argument shall contain the network name of the remote service that refused the connection.
2. *number*: The port number that was refused.
    - This argument shall contain the port number that was refused.

2.20.3 ConnectionTimeout

Example: The connection to 'Contoso.com:443' timed out.

Indicates that a connection to the remote service timed out.

- This message shall indicate that a connection to the remote service timed out.

Version Added: v1.0

Severity: Critical

Resolution: Check the settings on the remote service or the port settings.

Message and Arguments: "The connection to '`<1>`:`<2>`' timed out."

1. *string*: The network name of the service that timed out.
    - This argument shall contain the network name of the remote service that timed out.
2. *number*: The port number that timed out.
    - This argument shall contain the port number that timed out.

2.20.4 HTTPFailure

Example: Received HTTP status code 500 from 'Contoso.com'.

Indicates that an HTTP failure code was returned.

- This message shall indicate that an HTTP failure code was returned.

Version Added: v1.0

Severity: Critical

Resolution: Check the payload settings and the remote service for errors.

Message and Arguments: "Received HTTP status code `<1>` from '`<2>`'."

1. *number*: The HTTP status code that was returned.
    - This argument shall contain the HTTP status code indicating an error that was returned. This

argument should contain a value in the `4XX` or `5XX` range.

2. *string*: The network name of the service that returned the status code.
   - This argument shall contain the network name of the remote service that returned the status code.

2.20.5 NameResolutionFailure

Example: There was an error resolving the name 'Contoso.com'.

Indicates that DNS or other name resolution failed on the indicated name.

- This message shall indicate that DNS or other name resolution failed on the indicated name.

Version Added: v1.0

Severity: Critical

Resolution: Check your name server settings or the name provided.

Message and Arguments: "There was an error resolving the name '`<1>`'."

1. *string*: The network name that could not be resolved.
   - This argument shall contain the network name of the remote service that could not be resolved.

2.20.6 SecureCommunicationFailure

Example: Unable to establish a secure communication connection with 'https://contoso.com'.

Indicates that a secure communication connection could not be established, such as during TLS handshaking.

- This message shall indicate that a secure communication connection could not be established. Services shall use the `CertificateVerificationFailure` message if the failure was due to certificate verification.

Version Added: v1.2

Severity: Critical

Resolution: Check the configuration settings of the service and the remote service.

Message and Arguments: "Unable to establish a secure communication connection with '`<1>`'."

1. *string*: The network name of the service that could not establish a secure communication connection.
   - This argument shall contain the network name of the remote service that could not establish a secure communication connection.

2.21 StorageDevice 1.7.0

Version	*v1.7*	*v1.6*	*v1.5*	*v1.4*	*v1.3*	*v1.2*	*v1.1*	*v1.0*
Release	2026.2	2026.1	2025.2	2024.4	2024.3	2023.2	2021.4	2020.1

This registry defines the messages for storage devices.

Message	Severity	Description
BatteryCharging *(v1.1+)*	Warning	A battery charging condition was detected.
BatteryFailure *(v1.1+)*	Critical	A battery failure condition was detected.
BatteryMissing *(v1.1+)*	Critical	A battery missing condition was detected.
BatteryOK *(v1.1+)*	OK	The health of a battery has changed to OK.
BlockSecurityIDFailed *(v1.4+)*	Warning	The TCG-defined command to block establishment of a security ID failed.
BlockSecurityIDIssued *(v1.4+)*	OK	The TCG-defined command to block establishment of a security ID was issued.
BlockSecurityIDRemoved *(v1.4+)*	OK	TCG-defined commands to block establishment of a security ID are no longer being sent to a drive.
ControllerDegraded *(v1.1+)*	Warning	A storage controller degraded condition was detected.
ControllerFailure	Critical	A storage controller failure was detected.
ControllerOK	OK	The storage controller health has changed to OK.
ControllerPasswordAccepted *(v1.1+)*	OK	The storage controller password was entered.
ControllerPasswordRequired *(v1.1+)*	Critical	The storage controller requires a password.
ControllerPortDegraded *(v1.1+)*	Warning	A controller port degraded condition was detected.
ControllerPortFailure *(v1.1+)*	Critical	A controller port failure condition was detected.
ControllerPortOK *(v1.1+)*	OK	The health of a controller port has changed to OK.
ControllerPreviousError *(v1.1+)*	Warning	A storage controller error was detected prior to reboot.
DriveFailure	Critical	A drive failure condition was detected.
DriveFailureCleared	OK	A previously detected failure condition on a drive was cleared.

Message	Severity	Description
DriveInserted	OK	A drive was inserted.
DriveMissing *(v1.1+)*	Critical	A drive missing condition was detected.
DriveMissingCleared *(v1.1+)*	OK	A previous drive missing condition was cleared.
DriveOK	OK	The health of a drive has changed to OK.
DriveOffline *(v1.1+)*	Critical	A drive offline condition was detected.
DriveOfflineCleared *(v1.1+)*	OK	A drive offline condition was cleared.
DrivePredictedMediaLifeLow *(v1.3+)*	Warning	The drive media predicted life left is low.
DrivePredictiveFailure	Warning	A predictive drive failure condition was detected.
DrivePredictiveFailureCleared	OK	A previously detected predictive failure condition on a drive was cleared.
DriveRemoved	Critical	A drive was removed.
DriveSecureEraseCompleted *(v1.7+)*	OK	A secure drive erase operation was completed.
DriveSecureEraseFailed *(v1.7+)*	Critical	A secure drive erase operation failed.
DriveSecureEraseStarted *(v1.7+)*	OK	A secure drive erase operation was initiated.
VolumeCreated *(v1.7+)*	OK	A new volume was created under the storage controller.
VolumeDegraded	Warning	The storage controller has detected a degraded volume condition.
VolumeDeleted *(v1.7+)*	OK	A volume under the storage controller was deleted.
VolumeFailure	Critical	The storage controller has detected a failed volume condition.
VolumeOK	OK	A volume health has changed to OK.
VolumeOffine *(v1.1+, deprecated v1.2)*	Critical	The storage controller has detected an offline volume condition.
VolumeOffline *(v1.2+)*	Critical	The storage controller has detected an offline volume condition.
VolumeOfflineCleared *(v1.1+)*	OK	The storage controller has detected an online volume condition.
VolumeRebuilding *(v1.5+)*	Warning	The storage controller is in the process of rebuilding the volume as part of a repair process. Performance might be degraded while the rebuild is in progress.
VolumeReconfiguring *(v1.5+)*	Warning	The storage controller is in the process of reconfiguring the volume as part of adding new drives or changing RAID levels. Performance might be degraded while the reconfiguration is in progress.

Message	Severity	Description
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.

2.21.1 BatteryCharging

Example: A charging condition for the battery located in 'Chassis 1' was detected.

A battery charging condition was detected.

- This message shall indicate a battery is below the required capacity and is charging.

Version Added: v1.1

Severity: Warning

Resolution: None. There may be reduced performance and features while the battery is charging.

Message and Arguments: "A charging condition for the battery located in '`<1>`' was detected."

1. *string*: The location of the battery.
   - This argument shall contain the battery location.

2.21.2 BatteryFailure

Example: A failure condition for the battery located in 'Chassis 1' was detected.

A battery failure condition was detected.

- This message shall indicate a battery failure condition was detected.

Version Added: v1.1

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 '`<1>`' was detected."

1. *string*: The location of the battery.
    - This argument shall contain the battery location.

2.21.3 BatteryMissing

Example: The battery located in 'Chassis 1' is missing.

A battery missing condition was detected.

- This message shall indicate a battery that is required by the configuration is missing.

Version Added: v1.1

Severity: Critical

Resolution: Attach the battery. Ensure all cables are properly and securely connected.

Message and Arguments: "The battery located in '`<1>`' is missing."

1. *string*: The location of the battery.
    - This argument shall contain the battery location.

2.21.4 BatteryOK

Example: The health of the battery located in 'Chassis 1' has changed to OK.

The health of a battery has changed to OK.

- This message shall indicate a battery health has changed to OK.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The health of the battery located in '`<1>`' has changed to OK."

1. *string*: The location of the battery.
    - This argument shall contain the battery location.

2.21.5 BlockSecurityIDFailed

Example: The TCG-defined command to block establishment of a security ID failed when issued to the drive located in 'Bay 1'.

The TCG-defined command to block establishment of a security ID failed.

- This message shall indicate that the TCG-defined command to block establishment of a security ID failed when issued to a drive.

Version Added: v1.4

Severity: Warning

Resolution: Update the drive firmware or replace the drive.

Message and Arguments: "The TCG-defined command to block establishment of a security ID failed when issued to the drive located in '`<1>`'."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.

2.21.6 BlockSecurityIDIssued

Example: The TCG-defined command to block establishment of a security ID was issued to the drive located in 'Bay 1'.

The TCG-defined command to block establishment of a security ID was issued.

- This message shall indicate that the TCG-defined command to block establishment of a security ID was issued to a drive.

Version Added: v1.4

Severity: OK

Resolution: None.

Message and Arguments: "The TCG-defined command to block establishment of a security ID was issued to the drive located in '`<1>`'."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.

2.21.7 BlockSecurityIDRemoved

Example: The TCG-defined commands to block establishment of a security ID are no longer sent to the drive located in 'Bay 1'.

TCG-defined commands to block establishment of a security ID are no longer being sent to a drive.

- This message shall indicate that the TCG-defined command to block establishment of a security ID are no longer sent to a drive.

Version Added: v1.4

Severity: OK

Resolution: None.

Message and Arguments: "The TCG-defined commands to block establishment of a security ID are no longer sent to the drive located in '`<1>`'."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.

2.21.8 ControllerDegraded

Example: A degraded condition for the storage controller located in 'PCIe Slot 1' was detected due to reason 'Magic smoke escaping'.

A storage controller degraded condition was detected.

- This message shall indicate that a storage controller degraded condition was detected.

Version Added: v1.1

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 '`<1>`' was detected due to reason '`<2>`'."

1. *string*: The location of the storage controller.
    - This argument shall contain the storage controller location.
2. *string*: The reason for the degraded controller.
    - This argument shall contain the reason for the degraded controller.

2.21.9 ControllerFailure

Example: A failure condition for the storage controller located in 'PCIe Slot 1' was detected.

A storage controller failure was detected.

- This message shall indicate that a storage controller failure condition was detected due to a hardware, communication, or firmware failure.

Version Added: v1.0

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 '`<1>`' was detected."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.

2.21.10 ControllerOK

Example: The health of the storage controller located in 'PCIe Slot 1' has changed to OK.

The storage controller health has changed to OK.

- This message shall indicate that the storage controller health has changed to OK.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The health of the storage controller located in '`<1>`' has changed to OK."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.

2.21.11 ControllerPasswordAccepted

Example: A password was entered for the storage controller located in 'PCIe Slot 1'.

The storage controller password was entered.

- This message shall 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: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "A password was entered for the storage controller located in '<1>'."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.

2.21.12 ControllerPasswordRequired

Example: The storage controller located in 'PCIe Slot 1' requires a password.

The storage controller requires a password.

- This message shall 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: v1.1

Severity: Critical

Resolution: Enter the controller password.

Message and Arguments: "The storage controller located in '<1>' requires a password."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.

2.21.13 ControllerPortDegraded

Example: A degraded condition for the controller port located in 'PCIe Slot 1' was detected due to reason 'Magic smoke escaping'.

A controller port degraded condition was detected.

- This message shall indicate a controller port degraded condition was detected.

Version Added: v1.1

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 '<1>' was detected due to reason '<2>'."

1. *string*: The location of the controller port.
   - This argument shall contain the controller port location.
2. *string*: The reason for the degraded controller port.
   - This argument shall contain the reason for the degraded controller port.

2.21.14 ControllerPortFailure

Example: A failure condition for the controller port located in 'PCIe Slot 1' was detected.

A controller port failure condition was detected.

- This message shall indicate a controller port failure condition was detected.

Version Added: v1.1

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 '<1>' was detected."

1. *string*: The location of the controller port.
   - This argument shall contain the controller port location.

2.21.15 ControllerPortOK

Example: The health of the controller port located in 'PCIe Slot 1' has changed to OK.

The health of a controller port has changed to OK.

- This message shall indicate a controller port health has changed to OK.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The health of the controller port located in '<1>' has changed to OK."

1. *string*: The location of the controller port.
   - This argument shall contain the controller port location.

2.21.16 ControllerPreviousError

Example: A previous error condition for the storage controller located in 'PCIe Slot 1' was detected due to 'Magic smoke escaping'.

A storage controller error was detected prior to reboot.

- This message shall indicate that an unreported storage controller error condition was detected prior to reboot.

Version Added: v1.1

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 '`<1>`' was detected due to '`<2>`'."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.
2. *string*: The reason for error.
   - This argument shall contain the reason for failure.

2.21.17 DriveFailure

Example: A failure condition for the drive located in 'Bay 1' was detected.

A drive failure condition was detected.

- This message shall indicate a drive failure condition was detected due to a drive not installed correctly or failed.

Version Added: v1.0

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 '`<1>`' was detected."

1. *string*: The location of the drive.
   - This argument shall contain the drive location.

2.21.18 DriveFailureCleared

Example: A failure condition for the drive located in 'Bay 1' was cleared.

A previously detected failure condition on a drive was cleared.

- This message shall indicate a failure condition on a drive was cleared, but any predictive failure or other conditions remain.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A failure condition for the drive located in '`<1>`' was cleared."

1. *string*: The location of the drive.
   - This argument shall contain the drive location.

2.21.19 DriveInserted

Example: The drive located in 'Bay 1' was inserted.

A drive was inserted.

- This message shall indicate a drive was inserted.

Version Added: v1.0

Severity: OK

Resolution: If the drive is not properly displayed, attempt to refresh the cached data.

Message and Arguments: "The drive located in '`<1>`' was inserted."

1. *string*: The location of the drive.
   - This argument shall contain the drive location.

2.21.20 DriveMissing

Example: A missing condition for the drive located in 'Bay 1' was detected.

A drive missing condition was detected.

- This message shall 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: v1.1

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 it is no longer needed.

Message and Arguments: "A missing condition for the drive located in '`<1>`' was detected."

1. *string*: The location of the drive.
   - This argument shall contain the drive location.

2.21.21 DriveMissingCleared

Example: A missing condition for the drive located in 'Bay 1' was cleared.

A previous drive missing condition was cleared.

- This message shall indicate a previous drive missing condition was cleared.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "A missing condition for the drive located in '`<1>`' was cleared."

1. *string*: The location of the drive.
   - This argument shall contain the drive location.

2.21.22 DriveOK

Example: The health of the drive located in 'PCIe Slot 1' has changed to OK.

The health of a drive has changed to OK.

- This message shall indicate a drive health has changed to OK.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The health of the drive located in '<1>' has changed to OK."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.

2.21.23 DriveOffline

Example: An offline condition for the drive located in 'Bay 1' was detected.

A drive offline condition was detected.

- This message shall indicate a drive offline condition was detected. This may occur when a drive is unconfigured, encrypted by a foreign controller, offline during a background operation, or unsupported.

Version Added: v1.1

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 drive located in '<1>' was detected."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.

2.21.24 DriveOfflineCleared

Example: An offline condition for the drive located in 'Bay 1' was cleared.

A drive offline condition was cleared.

- This message shall indicate a drive offline condition was cleared.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "An offline condition for the drive located in '<1>' was cleared."

1. *string*: The location of the drive.

    - This argument shall contain the drive location.

2.21.25 DrivePredictedMediaLifeLow

Example: The drive located at 'Bay 1' has a predicted media life left of 10 percent and is below the specified threhsold of 20 percent.

The drive media predicted life left is low.

- This message shall indicate a drive media predicted life left level is below a user or vendor-defined threshold.

Version Added: v1.3

Severity: Warning

Resolution: Replace the drive before the predicted media life left reaches 0%.

Message and Arguments: "The drive located at `<1>` has a predicted media life left of `<2>` percent and is below the specified threhsold of `<3>` percent."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.
2. *number*: The percent of predicted media life left.
    - This argument shall contain the value of PredictedMediaLifeLeft for this drive.
3. *number*: The alert threshold.
    - This argument shall contain the value of the threshold that triggered this message.

2.21.26 DrivePredictiveFailure

Example: A predictive failure condition for the drive located in 'Bay 1' was detected.

A predictive drive failure condition was detected.

- This message shall indicate a drive predictive failure condition was detected.

Version Added: v1.0

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 ' `<1>` ' was detected."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.

2.21.27 DrivePredictiveFailureCleared

Example: A predictive failure condition for the drive located in 'Bay 1' was cleared.

A previously detected predictive failure condition on a drive was cleared.

- This message shall indicate a predictive failure condition on a drive was cleared, but any other failure conditions remain.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "A predictive failure condition for the drive located in '`<1>`' was cleared."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.

2.21.28 DriveRemoved

Example: The drive located in 'Bay 1' was removed.

A drive was removed.

- This message shall indicate a drive was removed.

Version Added: v1.0

Severity: Critical

Resolution: If the drive is still displayed, attempt to refresh the cached data.

Message and Arguments: "The drive located in '`<1>`' was removed."

1. *string*: The location of the drive.
    - This argument shall contain the drive location.

2.21.29 DriveSecureEraseCompleted

Example: A secure drive erase operation of type 'CryptographicErase' for the drive located in 'Bay 1' was completed.

A secure drive erase operation was completed.

- This message shall indicate a secure drive erase operation was completed.

Version Added: v1.7

Severity: OK

Resolution: None.

Message and Arguments: "A secure drive erase operation of type '`<1>`' for the drive located in '`<2>`' was completed."

1. *string*: The `DataSanitizationType` for the secure erase operation.
   - This argument shall contain the `DataSanitizationType` for the secure erase operation. If no `DataSanitizationType` was specified, or if the secure erase was not initiated through the `SecureErase` action, this argument may contain `Default`.
2. *string*: The location of the drive.
   - This argument shall contain the drive location.

2.21.30 DriveSecureEraseFailed

Example: A secure drive erase operation of type 'CryptographicErase' for the drive located in 'Bay 1' failed.

A secure drive erase operation failed.

- This message shall indicate a secure drive erase operation failed.

Version Added: v1.7

Severity: Critical

Resolution: Check the drive and storage controller for reported errors or health issues and take corrective action before retrying the secure erase operation.

Message and Arguments: "A secure drive erase operation of type '`<1>`' for the drive located in '`<2>`' failed."

1. *string*: The `DataSanitizationType` for the secure erase operation.
   - This argument shall contain the `DataSanitizationType` for the secure erase operation. If no `DataSanitizationType` was specified, or if the secure erase was not initiated through the `SecureErase` action, this argument may contain `Default`.
2. *string*: The location of the drive.
   - This argument shall contain the drive location.

2.21.31 DriveSecureEraseStarted

Example: A secure drive erase operation of type 'CryptographicErase' for the drive located in 'Bay 1' has started.

A secure drive erase operation was initiated.

- This message shall indicate a secure drive erase operation was initiated.

Version Added: v1.7

Severity: OK

Resolution: None.

Message and Arguments: "A secure drive erase operation of type ' `<1>` ' for the drive located in ' `<2>` ' has started."

1. *string*: The `DataSanitizationType` for the secure erase operation.
   - This argument shall contain the `DataSanitizationType` for the secure erase operation. If no `DataSanitizationType` was specified, or if the secure erase was not initiated through the `SecureErase` action, this argument may contain `Default`.
2. *string*: The location of the drive.
   - This argument shall contain the drive location.

2.21.32 VolumeCreated

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' was created.

A new volume was created under the storage controller.

- This message shall indicate a new volume was created under the storage controller.

Version Added: v1.7

Severity: OK

Resolution: None.

Message and Arguments: "The volume ' `<1>` ' attached to the storage controller located in ' `<2>` ' was created."

1. *string*: The identifier of the volume.
   - This argument shall contain 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.

    - This argument shall contain the storage controller location.

2.21.33 VolumeDegraded

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' is degraded.

The storage controller has detected a degraded volume condition.

- This message shall indicate a volume degraded condition was detected due to one or more drives not installed correctly, missing, or failed.

Version Added: v1.0

Severity: Warning

Resolution: Ensure all cables are properly and securely connected. Replace failed drives.

Message and Arguments: "The volume ' `<1>` ' attached to the storage controller located in ' `<2>` ' is degraded."

1. *string*: The identifier of the volume.
    - This argument shall contain 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.
    - This argument shall contain the storage controller location.

2.21.34 VolumeDeleted

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' was deleted.

A volume under the storage controller was deleted.

- This message shall indicate a volume under the storage controller was deleted.

Version Added: v1.7

Severity: OK

Resolution: None.

Message and Arguments: "The volume ' `<1>` ' attached to the storage controller located in ' `<2>` ' was deleted."

1. *string*: The identifier of the volume.
    - This argument shall contain 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.
    - This argument shall contain the storage controller location.

2.21.35 VolumeFailure

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' has failed.

The storage controller has detected a failed volume condition.

- This message shall indicate a volume failure condition was detected due to one or more drives not installed correctly, missing, or failed.

Version Added: v1.0

Severity: Critical

Resolution: Ensure all cables are properly and securely connected. Ensure all drives are fully seated and operational.

Message and Arguments: "The volume ' `<1>` ' attached to the storage controller located in ' `<2>` ' has failed."

1. *string*: The identifier of the volume.
    - This argument shall contain 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.
    - This argument shall contain the storage controller location.

2.21.36 VolumeOK

Example: The health of volume 'Volume1' that is attached to the storage controller located in 'PCIe Slot 1' has changed to OK.

A volume health has changed to OK.

- This message shall indicate a volume health has changed to OK.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The health of volume ' `<1>` ' that is attached to the storage controller located in ' `<2>` ' has changed to OK."

1. *string*: The identifier of the volume.
   - This argument shall contain 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.
   - This argument shall contain the storage controller location.

2.21.37 VolumeOffine

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' is offline.

The storage controller has detected an offline volume condition.

- This message shall indicate a volume offline condition was detected. This may occur if the volume configuration is not supported or needs to be imported.

Deprecated in v1.2 and later. This message was deprecated in favor of `VolumeOffline` due to a typographic error in the `MessageId`.

Version Added: v1.1

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 '`<1>`' attached to the storage controller located in '`<2>`' is offline."

1. *string*: The identifier of the volume.
   - This argument shall contain 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.
   - This argument shall contain the storage controller location.

2.21.38 VolumeOffline

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' is offline.

The storage controller has detected an offline volume condition.

- This message shall indicate a volume offline condition was detected. This may occur if the volume configuration is not supported or needs to be imported.

Version Added: v1.2

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 '`<1>`' attached to the storage controller located in '`<2>`' is offline."

1. *string*: The identifier of the volume.
   - This argument shall contain 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.
   - This argument shall contain the storage controller location.

2.21.39 VolumeOfflineCleared

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' is online.

The storage controller has detected an online volume condition.

- This message shall indicate a volume online condition was detected.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "The volume '`<1>`' attached to the storage controller located in '`<2>`' is online."

1. *string*: The identifier of the volume.
   - This argument shall contain 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.
   - This argument shall contain the storage controller location.

2.21.40 VolumeRebuilding

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' is being rebuilt.

The storage controller is in the process of rebuilding the volume as part of a repair process. Performance might be degraded while the rebuild is in progress.

- This message shall indicate that the storage controller is in the process of rebuilding the volume as part of a repair process. Performance might be degraded while the rebuild is in progress.

Version Added: v1.5

Severity: Warning

Resolution: Wait for the volume rebuild to complete.

Message and Arguments: "The volume ' `<1>` ' attached to the storage controller located in ' `<2>` ' is being rebuilt."

1. *string*: The identifier of the volume.
   - This argument shall contain 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.
   - This argument shall contain the storage controller location.

2.21.41 VolumeReconfiguring

Example: The volume 'Volume1' attached to the storage controller located in 'PCIe Slot 1' is being reconfigured.

The storage controller is in the process of reconfiguring the volume as part of adding new drives or changing RAID levels. Performance might be degraded while the reconfiguration is in progress.

- This message shall indicate that the storage controller is in the process of reconfiguring the volume as part of adding new drives or changing RAID levels. Performance might be degraded while the reconfiguration is in progress.

Version Added: v1.5

Severity: Warning

Resolution: Wait for the volume reconfiguration to complete.

Message and Arguments: "The volume ' `<1>` ' attached to the storage controller located in ' `<2>` ' is being reconfigured."

1. *string*: The identifier of the volume.
   - This argument shall contain 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.
   - This argument shall contain the storage controller location.

2.21.42 WriteCacheDataLoss

Example: The write cache on the storage controller located in 'PCIe Slot 1' has data loss.

The write cache is reporting loss of data in posted-writes memory.

- This message shall 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: v1.0

Severity: Critical

Resolution: Check the controller resource properties to determine the cause of the write cache data loss.

Message and Arguments: "The write cache on the storage controller located in '`<1>`' has data loss."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.

2.21.43 WriteCacheDegraded

Example: The write cache state on the storage controller located in 'PCIe Slot 1' is degraded.

The write cache state is degraded.

- This message shall indicate the write cache state degraded due to a missing battery or hardware failure.

Version Added: v1.0

Severity: Critical

Resolution: Check the controller to determine the cause of write cache degraded state, such as a missing battery or hardware failure. This condition typically requires manual intervention to resolve.

Message and Arguments: "The write cache state on the storage controller located in '`<1>`' is degraded."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.

2.21.44 WriteCacheProtected

Example: The write cache state on the storage controller located in 'PCIe Slot 1' is in protected mode.

A storage controller write cache state is in protected mode.

- This message shall indicate a controller write cache state is in protected mode.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The write cache state on the storage controller located in '`<1>`' is in protected mode."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.

2.21.45 WriteCacheTemporarilyDegraded

Example: The write cache state on the storage controller located in 'PCIe Slot 1' is temporarily degraded.

The write cache state is temporarily degraded.

- This message shall indicate write cache state temporarily degraded due to battery charging or data recovery rebuild operation pending.

Version Added: v1.0

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. Wait for the condition causing the temporarily degraded state to resolve itself.

Message and Arguments: "The write cache state on the storage controller located in '`<1>`' is temporarily degraded."

1. *string*: The location of the storage controller.
   - This argument shall contain the storage controller location.

2.22 TaskEvent 1.1.0

Version	*v1.1*	*v1.0*
Release	2026.1	2018.2

This registry defines the messages for task related events.

Message	Severity	Description
TaskAborted	Critical	A task has completed with errors.

Message	Severity	Description
TaskCancelled	Warning	A task was cancelled.
TaskCompletedOK	OK	A task has completed.
TaskCompletedWarning	Warning	A task has completed with warnings.
TaskPaused	Warning	A task was paused.
TaskProgressChanged	OK	A task has changed progress.
TaskRemoved	Warning	A task was removed.
TaskResumed	OK	A task was resumed.
TaskStarted	OK	A task has started.

2.22.1 TaskAborted

Example: The task with Id 'Task1' has completed with errors.

A task has completed with errors.

- This message shall indicate that a task has completed with errors and the `TaskState` property transitioned to `Exception`.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "The task with Id '`<1>`' 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.

2.22.2 TaskCancelled

Example: Work on the task with Id 'Task1' was halted prior to completion due to an explicit request.

A task was cancelled.

- This message shall indicate that work on a task has halted prior to completion due to an explicit request. The

`TaskState` property transitioned to `Cancelled`.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "Work on the task with Id '`<1>`' was 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 was halted.

2.22.3 TaskCompletedOK

Example: The task with Id 'Task1' has completed.

A task has completed.

- This message shall indicate that a task has completed and the `TaskState` property transitioned to `Completed`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The task with Id '`<1>`' 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.

2.22.4 TaskCompletedWarning

Example: The task with Id 'Task1' has completed with warnings.

A task has completed with warnings.

- This message shall indicate that a task has completed with warnings and the `TaskState` property transitioned to `Completed`.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The task with Id ' `<1>` ' 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.

2.22.5 TaskPaused

Example: The task with Id 'Task1' was paused.

A task was paused.

- This message shall indicate that a task was paused and the `TaskState` property transitioned to `Suspended` or `Interrupted`.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The task with Id ' `<1>` ' was paused."

1. *string*: The `Id` of the task.
    - This argument shall contain the value of the `Id` property of the `Task` resource that was paused.

2.22.6 TaskProgressChanged

Example: The task with Id 'Task1' has changed to progress 50 percent complete.

A task has changed progress.

- This message shall indicate that a task has changed progress.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The task with Id ' `<1>` ' has changed to progress `<2>` 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.

2.22.7 TaskRemoved

Example: The task with Id 'Task1' was removed.

A task was removed.

- This message shall indicate that a task was removed.

Version Added: v1.0

Severity: Warning

Resolution: None.

Message and Arguments: "The task with Id ' `<1>` ' was removed."

1. *string*: The `Id` of the task.
    - This argument shall contain the value of the `Id` property of the `Task` resource that was removed.

2.22.8 TaskResumed

Example: The task with Id 'Task1' was resumed.

A task was resumed.

- This message shall indicate that a task has resumed and the `TaskState` property transitioned to `Running` from `Suspended` or `Interrupted`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The task with Id ' `<1>` ' was resumed."

1. *string*: The `Id` of the task.
    - This argument shall contain the value of the `Id` property of the `Task` resource that was

resumed.

2.22.9 TaskStarted

Example: The task with Id 'Task1' has started.

A task has started.

- This message shall indicate that a task has started as the `TaskState` property transitions to `Starting` or `Running`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The task with Id '`<1>`' 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.

2.23 Telemetry 1.2.0

Version	*v1.2*	*v1.1*	*v1.0*
Release	2026.1	2025.2	2023.3

This registry defines the messages for telemetry related events.

Message	Severity	Description
TelemetryDataCreated *(v1.1+)*	OK	Indicates that a telemetry data resource was created.
TriggerDiscreteConditionMet	OK	Indicates that a discrete trigger condition is met.
TriggerNumericAboveLowerCritical	Warning	Indicates that a numeric metric reading is no longer below the lower critical trigger threshold, but is still outside of normal operating range.
TriggerNumericAboveUpperCritical	Critical	Indicates that a numeric metric reading is above the upper critical trigger threshold.
TriggerNumericAboveUpperWarning	Warning	Indicates that a numeric metric reading is above the upper warning trigger threshold.
TriggerNumericBelowLowerCritical	Critical	Indicates that a numeric metric reading is below the lower critical trigger threshold.

Message	Severity	Description
TriggerNumericBelowLowerWarning	Warning	Indicates that a numeric metric reading is below the lower warning trigger threshold.
TriggerNumericBelowUpperCritical	Warning	Indicates that a numeric metric reading is no longer above the upper critical trigger threshold, but is still outside of normal operating range.
TriggerNumericReadingNormal	OK	Indicates that a numeric metric reading is now within normal operating range.

2.23.1 TelemetryDataCreated

Example: Telemetry data 'CPU Usage', containing type 'ContosoCPUDebug', was created.

Indicates that a telemetry data resource was created.

- This message shall indicate that a telemetry data resource was created.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "Telemetry data '`<1>`', containing type '`<2>`', was created."

1. *string*: The name of the telemetry data.
   - This argument shall contain a string that identifies `TelemetryData` resource.
2. *string*: The type of telemetry data
   - This argument shall contain the value of the `TelemetryDataType` property or `OEMTelemetryDataType` property from the `TelemetryData` resource.

2.23.2 TriggerDiscreteConditionMet

Example: Metric 'Power Supply Status' has the value 'Absent', which meets the discrete condition of trigger 'Power Supply Absent'

Indicates that a discrete trigger condition is met.

- This message shall indicate that the `Value` of the `DiscreteTriggers` property in a `Triggers` resource is equal to the monitored discrete metric value. The severity of the event shall be the same as `Severity` property in the related `Triggers` resource.

Version Added: v1.0

Severity: OK

Resolution: Check the condition of the metric that reported the trigger.

Message and Arguments: "Metric '`<1>`' has the value '`<2>`', which meets the discrete condition of trigger '`<3>`'"

1. *string*: The metric name.
   - This argument shall contain a string that identifies or describes the metric.
2. *string*: The discrete value of the metric.
   - This argument shall contain a number that equals the discrete value of the metric reading.
3. *string*: The telemetry trigger name.
   - This argument shall contain a string that identifies or describes the `Triggers` resource.

2.23.3 TriggerNumericAboveLowerCritical

Example: Metric 'CPU Usage' value of 15 is now above the 10 lower critical threshold of trigger 'Low CPU Usage' but remains outside of normal range

Indicates that a numeric metric reading is no longer below the lower critical trigger threshold, but is still outside of normal operating range.

- This message shall indicate that a numeric value of a monitored metric, previously below the threshold, is now above the `Reading` property within the `LowerCritical` property in a `Triggers` resource`. However, the value of the metric is still below the` Reading `property within the` LowerWarning` property and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the metric that reported the trigger.

Message and Arguments: "Metric '`<1>`' value of `<2>` is now above the `<3>` lower critical threshold of trigger '`<4>`' but remains outside of normal range"

1. *string*: The metric name.
   - This argument shall contain a string that identifies or describes the metric.
2. *number*: The value of the metric.
   - This argument shall contain a number that equals the value of the metric reading.
3. *number*: The threshold value.
   - This argument shall contain a number that equals the value of the `Reading` property within the `LowerCritical` property of the `Triggers` resource.
4. *string*: The telemetry trigger name.
   - This argument shall contain a string that identifies or describes the `Triggers` resource.

2.23.4 TriggerNumericAboveUpperCritical

Example: Metric 'CPU Usage' value of 95 is above the 90 upper critical threshold of trigger 'High CPU Usage'

Indicates that a numeric metric reading is above the upper critical trigger threshold.

- This message shall indicate that a numeric value of a monitored metric is above the `Reading` property within the `UpperCritical` property in a `Triggers` resource`.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the metric that reported the trigger.

Message and Arguments: "Metric '`<1>`' value of `<2>` is above the `<3>` upper critical threshold of trigger '`<4>`'"

1. *string*: The metric name.
   - This argument shall contain a string that identifies or describes the metric.
2. *number*: The value of the metric.
   - This argument shall contain a number that equals the value of the metric reading.
3. *number*: The threshold value.
   - This argument shall contain a number that equals the value of the `Reading` property within the `UpperCritical` property of the `Triggers` resource.
4. *string*: The telemetry trigger name.
   - This argument shall contain a string that identifies or describes the `Triggers` resource.

2.23.5 TriggerNumericAboveUpperWarning

Example: Metric 'CPU Usage' value of 85 is above the 80 upper warning threshold of trigger 'High CPU Usage'

Indicates that a numeric metric reading is above the upper warning trigger threshold.

- This message shall indicate that a numeric value of a monitored metric is above the `Reading` property within the `UpperWarning` property in a `Triggers` resource`.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the metric that reported the trigger.

Message and Arguments: "Metric '`<1>`' value of `<2>` is above the `<3>` upper warning threshold of trigger '`<4>`'"

1. *string*: The metric name.
   - This argument shall contain a string that identifies or describes the metric.
2. *number*: The value of the metric.
   - This argument shall contain a number that equals the value of the metric reading.
3. *number*: The threshold value.
   - This argument shall contain a number that equals the value of the `Reading` property within the `UpperWarning` property of the `Triggers` resource.
4. *string*: The telemetry trigger name.
   - This argument shall contain a string that identifies or describes the `Triggers` resource.

2.23.6 TriggerNumericBelowLowerCritical

Example: Metric 'CPU Usage' value of 5 is below the 10 lower critical threshold of trigger 'Low CPU Usage'

Indicates that a numeric metric reading is below the lower critical trigger threshold.

- This message shall indicate that a numeric value of a monitored metric is below the `Reading` property within the `LowerCritical` property in a `Triggers` resource`.

Version Added: v1.0

Severity: Critical

Resolution: Check the condition of the metric that reported the trigger.

Message and Arguments: "Metric ' `<1>` ' value of `<2>` is below the `<3>` lower critical threshold of trigger ' `<4>` '"

1. *string*: The metric name.
   - This argument shall contain a string that identifies or describes the metric.
2. *number*: The value of the metric.
   - This argument shall contain a number that equals the value of the metric reading.
3. *number*: The threshold value.
   - This argument shall contain a number that equals the value of the `Reading` property within the `LowerCritical` property of the `Triggers` resource.
4. *string*: The telemetry trigger name.
   - This argument shall contain a string that identifies or describes the `Triggers` resource.

2.23.7 TriggerNumericBelowLowerWarning

Example: Metric 'CPU Usage' value of 15 is below the 20 lower warning threshold of trigger 'Low CPU Usage'

Indicates that a numeric metric reading is below the lower warning trigger threshold.

- This message shall indicate that a numeric value of a monitored metric is below the `Reading` property within the `LowerWarning` property in a `Triggers` resource`.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the metric that reported the trigger.

Message and Arguments: "Metric '`<1>`' value of `<2>` is below the `<3>` lower warning threshold of trigger '`<4>`'"

1. *string*: The metric name.
   - This argument shall contain a string that identifies or describes the metric.
2. *number*: The value of the metric.
   - This argument shall contain a number that equals the value of the metric reading.
3. *number*: The threshold value.
   - This argument shall contain a number that equals the value of the `Reading` property within the `LowerWarning` property of the `Triggers` resource.
4. *string*: The telemetry trigger name.
   - This argument shall contain a string that identifies or describes the `Triggers` resource.

2.23.8 TriggerNumericBelowUpperCritical

Example: Metric 'CPU Usage' value of 85 is now below the 90 upper critical threshold of trigger 'High CPU Usage' but remains outside of normal range

Indicates that a numeric metric reading is no longer above the upper critical trigger threshold, but is still outside of normal operating range.

- This message shall indicate that a numeric value of a monitored metric, previously above the threshold, is now below the `Reading` property within the `UpperCritical` property in a `Triggers` resource`. However, the value of the metric is still above the` Reading `property within the` UpperWarning` property and is not within the normal operating range.

Version Added: v1.0

Severity: Warning

Resolution: Check the condition of the metric that reported the trigger.

Message and Arguments: "Metric '`<1>`' value of `<2>` is now below the `<3>` upper critical threshold of trigger '`<4>`' but remains outside of normal range"

1. *string*: The metric name.

    - This argument shall contain a string that identifies or describes the metric.
2. *number*: The value of the metric.
    - This argument shall contain a number that equals the value of the metric reading.
3. *number*: The threshold value.
    - This argument shall contain a number that equals the value of the `Reading` property within the `UpperCritical` property of the `Triggers` resource.
4. *string*: The telemetry trigger name.
    - This argument shall contain a string that identifies or describes the `Triggers` resource.

2.23.9 TriggerNumericReadingNormal

Example: Metric 'CPU Usage' value of 50 is within normal operating range of trigger 'High CPU Usage'

Indicates that a numeric metric reading is now within normal operating range.

- This message shall indicate that a numeric value of a monitored metric in a `Triggers` resource has returned to its normal operating range.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Metric '`<1>`' value of `<2>` is within normal operating range of trigger '`<3>`'"

1. *string*: The metric name.
    - This argument shall contain a string that identifies or describes the metric.
2. *number*: The value of the metric.
    - This argument shall contain a number that equals the value of the metric reading.
3. *string*: The telemetry trigger name.
    - This argument shall contain a string that identifies or describes the `Triggers` resource.

2.24 Update 1.5.0

Version	*v1.5*	*v1.4*	*v1.3*	*v1.2*	*v1.1*	*v1.0*
Release	2026.2	2026.1	TBD	2025.2	2024.1	2021.1

This registry defines the update status and error messages.

Message	Severity	Description
ActivateFailed	Critical	Indicates that the component failed to activate the image.
ActivatingOnComponent *(v1.4+)*	OK	Indicates that a component is activating an image.
AllTargetsDetermined	OK	Indicates that all target resources or devices for an update operation were determined by the service.
ApplyFailed *(deprecated v1.4)*	Critical	Indicates that the component failed to apply an image.
ApplyingOnComponent *(deprecated v1.4)*	OK	Indicates that a component is applying an image.
AutomaticRecoveryFailed *(v1.5+)*	Critical	Indicates that automatic recovery failed.
AutomaticRecoveryInitiated *(v1.5+)*	OK	Indicates that an automatic recovery procedure was initiated.
AutomaticRecoverySuccessful *(v1.5+)*	OK	Indicates that automatic recovery completed successfully.
AwaitToActivate	OK	Indicates that the resource or device is waiting for an action to proceed with activating an image.
AwaitToUpdate	OK	Indicates that the resource or device is waiting for an action to proceed with installing an image.
CleanupFailed *(v1.4+)*	Critical	Indicates that the component failed to perform post-update cleanup of the image.
CleanupOnComponent *(v1.4+)*	OK	Indicates that a component is performing post-update cleanup of an image.
InstallingOnComponent	OK	Indicates that a component is installing an image.
NoTargetsDetermined *(v1.1+)*	OK	Indicates that no target resource or device for an image was determined for update.
OperationTransitionedToJob	OK	Indicates that the update operation transitioned to a job for managing the progress of the operation.
StageFailed *(v1.4+)*	Critical	Indicates that the component failed to stage the image.
StagingOnComponent *(v1.4+)*	OK	Indicates that a component is staging an image.
TargetDetermined	OK	Indicates that a target resource or device for an image was determined for update.
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.
UpdateNotApplicable *(v1.2+)*	Warning	Indicates that the update was not applicable to the resource or device.

Message	Severity	Description
UpdatePending *(v1.5+)*	OK	Indicates that an update operation is pending.
UpdateSkipped *(v1.2+)*	OK	Indicates that an image was applicable to a resource or device, but the update was skipped due to policies in the service.
UpdateSkippedSameVersion *(v1.2+)*	OK	Indicates that a resource or device update was skipped because the same version is already installed.
UpdateSkippedVersionTooLow *(v1.3+)*	OK	Indicates that a resource or device update was skipped because the version of the provided image was lower than the minimum required version.
UpdateSuccessful	OK	Indicates that a resource or device was updated.
VerificationFailed	Critical	Indicates that the service or component failed to verify an image.
VerifyingAtComponent	OK	Indicates that a component is verifying an image.

2.24.1 ActivateFailed

Example: Activation of image 'FWUpdate1' on 'NIC1' failed.

Indicates that the component failed to activate the image.

- This message shall indicate that the component failed to activate the image.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "Activation of image '`<1>`' on '`<2>`' 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.

2.24.2 ActivatingOnComponent

Example: Image 'FWUpdate1' is being activated on 'NIC1'.

Indicates that a component is activating an image.

- This message shall indicate that a component is activating an image.

Version Added: v1.4

Severity: OK

Resolution: None.

Message and Arguments: "Image '`<1>`' is being activated on '`<2>`'."

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.

2.24.3 AllTargetsDetermined

Example: All the target devices to be updated were determined.

Indicates that all target resources or devices for an update operation were determined by the service.

- This message shall indicate that all target resources or devices for an update operation were determined by the service.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "All the target devices to be updated were determined."

2.24.4 ApplyFailed

Example: Installation of image 'FWUpdate1' to 'NIC1' failed.

Indicates that the component failed to apply an image.

- This message shall indicate that the component failed to apply an image.

Deprecated in v1.4 and later. This message was deprecated in favor of `ActivateFailed`.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "Installation of image '`<1>`' to '`<2>`' 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.

2.24.5 ApplyingOnComponent

Example: Image 'FWUpdate1' is being applied on 'NIC1'.

Indicates that a component is applying an image.

- This message shall indicate that a component is applying an image.

Deprecated in v1.4 and later. This message was deprecated in favor of `ActivatingOnComponent`.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Image '`<1>`' is being applied on '`<2>`'."

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.

2.24.6 AutomaticRecoveryFailed

Example: Automatic recovery failed for 'NIC1' with image 'FWUpdate1'.

Indicates that automatic recovery failed.

- This message shall indicate that an automatic recovery attempt for the component failed.

Version Added: v1.5

Severity: Critical

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Automatic recovery failed for '`<1>`' with image '`<2>`'."

1. *string*: The URI or name of the target device.
   - This argument shall contain the URI of the resource or a unique name of the device.
2. *string*: The identifier for the image.
   - This argument shall contain the identifier for the image.

2.24.7 AutomaticRecoveryInitiated

Example: Automatic recovery initiated for 'NIC1' with image 'FWUpdate1'.

Indicates that an automatic recovery procedure was initiated.

- This message shall indicate that an automatic recovery procedure was initiated in response to a detected fault or policy violation.

Version Added: v1.5

Severity: OK

Resolution: Check additional diagnostic data if available.

Message and Arguments: "Automatic recovery initiated for '`<1>`' with image '`<2>`'."

1. *string*: The URI or name of the target device.
   - This argument shall contain the URI of the resource or a unique name of the device.
2. *string*: The identifier for the image.
   - This argument shall contain the identifier for the image.

2.24.8 AutomaticRecoverySuccessful

Example: Automatic recovery completed for 'NIC1' with image 'FWUpdate1'.

Indicates that automatic recovery completed successfully.

- This message shall indicate that an automatic recovery attempt for the component completed successfully.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "Automatic recovery completed for '`<1>`' with image '`<2>`'."

1. *string*: The URI or name of the target device.
   - This argument shall contain the URI of the resource or a unique name of the device.
2. *string*: The identifier for the image.
   - This argument shall contain the identifier for the image.

2.24.9 AwaitToActivate

Example: Awaiting for an action to proceed with activating image 'FWUpdate1' on 'NIC1'.

Indicates that the resource or device is waiting for an action to proceed with activating an image.

- This message shall indicate that the resource or device is waiting for an action to proceed with activating an image.

Version Added: v1.0

Severity: OK

Resolution: Perform the requested action to advance the update operation.

Message and Arguments: "Awaiting for an action to proceed with activating image '`<1>`' on '`<2>`'."

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 awating 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.

2.24.10 AwaitToUpdate

Example: Awaiting for an action to proceed with installing image 'FWUpdate1' on 'NIC1'.

Indicates that the resource or device is waiting for an action to proceed with installing an image.

- This message shall indicate that the resource or device is waiting for an action to proceed with installing an image.

Version Added: v1.0

Severity: OK

Resolution: Perform the requested action to advance the update operation.

Message and Arguments: "Awaiting for an action to proceed with installing image '`<1>`' on '`<2>`'."

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 awating 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.

2.24.11 CleanupFailed

Example: Cleaning of image 'FWUpdate1' on 'NIC1' failed.

Indicates that the component failed to perform post-update cleanup of the image.

- This message shall indicate that the component failed to perform post-update cleanup of the image.

Version Added: v1.4

Severity: Critical

Resolution: None.

Message and Arguments: "Cleaning of image '`<1>`' on '`<2>`' 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 cleaning the image.
   - This argument shall contain the URI of the resource or a unique name of the component cleaning the image.

2.24.12 CleanupOnComponent

Example: Image 'FWUpdate1' is being cleaned from 'NIC1'.

Indicates that a component is performing post-update cleanup of an image.

- This message shall indicate that a component is performing post-update cleanup an image.

Version Added: v1.4

Severity: OK

Resolution: None.

Message and Arguments: "Image '`<1>`' is being cleaned from '`<2>`'."

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 cleaning the image.
   - This argument shall contain the URI of the resource or a unique name of the component cleaning the image.

2.24.13 InstallingOnComponent

Example: Image 'FWUpdate1' is being installed on 'NIC1'.

Indicates that a component is installing an image.

- This message shall indicate that a component is installing an image.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Image '`<1>`' is being installed on '`<2>`'."

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.

2.24.14 NoTargetsDetermined

Example: No target device will be updated with image 'FWUpdate1'.

Indicates that no target resource or device for an image was determined for update.

- This message shall indicate that no target resource or device for an image was determined for update.

Version Added: v1.1

Severity: OK

Resolution: None.

Message and Arguments: "No target device will be updated with image '`<1>`'."

1. *string*: The identifier for the image.
    - This argument shall contain the identifier for the image.

2.24.15 OperationTransitionedToJob

Example: The update operation has transitioned to the job at URI '/redfish/v1/JobService/Jobs/1'.

Indicates that the update operation transitioned to a job for managing the progress of the operation.

- This message shall indicate that the update operation transitioned to a job for managing the progress of the operation.

Version Added: v1.0

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 '`<1>`'."

1. *string*: The job URI.
    - This argument shall contain the URI of the `Job` resource that is managing the progress of the update operation.

2.24.16 StageFailed

Example: Staging of image 'FWUpdate1' on 'NIC1' failed.

Indicates that the component failed to stage the image.

- This message shall indicate that the component failed to stage the image.

Version Added: v1.4

Severity: Critical

Resolution: None.

Message and Arguments: "Staging of image '`<1>`' on '`<2>`' 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 staging the image.
    - This argument shall contain the URI of the resource or a unique name of the component staging the image.

2.24.17 StagingOnComponent

Example: Image 'FWUpdate1' is being staged on 'NIC1'.

Indicates that a component is staging an image.

- This message shall indicate that a component is staging an image.

Version Added: v1.4

Severity: OK

Resolution: None.

Message and Arguments: "Image '`<1>`' is being staged on '`<2>`'."

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 staging the image.
    - This argument shall contain the URI of the resource or a unique name of the component staging the image.

2.24.18 TargetDetermined

Example: The target device 'Disk1' will be updated with image 'FWUpdate1'.

Indicates that a target resource or device for an image was determined for update.

- This message shall indicate that a target resource or device for an image was determined for update.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "The target device '`<1>`' will be updated with image '`<2>`'."

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.

2.24.19 TransferFailed

Example: Transfer of image 'FWUpdate1' to 'NIC1' failed.

Indicates that the service failed to transfer an image to a component.

- This message shall indicate that the service failed to transfer an image to a component.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "Transfer of image '`<1>`' to '`<2>`' failed."

1. *string*: The identifier for the image, or the filename of the image if an identifier cannot be determined.
    - This argument shall contain the identifier for the image. If the identifier cannot be determined, this argument shall contain the filename of 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.

2.24.20 TransferringToComponent

Example: Image 'FWUpdate1' is being transferred to 'NIC1'.

Indicates that the service is transferring an image to a component.

- This message shall indicate that the service is transferring an image to a component.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Image '`<1>`' is being transferred to '`<2>`'."

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.

2.24.21 UpdateInProgress

Example: An update is in progress.

Indicates that an update is in progress.

- This message shall indicate that an update is in progress.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "An update is in progress."

2.24.22 UpdateNotApplicable

Example: Image 'FWUpdate1' was not applicable to device 'NIC1'.

Indicates that the update was not applicable to the resource or device.

- This message shall indicate that the update was not applicable to the resource or device.

Version Added: v1.2

Severity: Warning

Resolution: None.

Message and Arguments: "Image '`<1>`' was not applicable to device '`<2>`'."

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 device.
    - This argument shall contain the URI of the resource or a unique name of the device.

2.24.23 UpdatePending

Example: An update is pending.

Indicates that an update operation is pending.

- This message shall indicate that an update operation is pending, such as an update request that contains the `@Redfish.OperationApplyTime` annotation and has not started.

Version Added: v1.5

Severity: OK

Resolution: None.

Message and Arguments: "An update is pending."

2.24.24 UpdateSkipped

Example: Device 'NIC1' skipped the update with image 'FWUpdate1'.

Indicates that an image was applicable to a resource or device, but the update was skipped due to policies in the service.

- This message shall indicate that an image was applicable to a resource or device, but the update was skipped due to policies in the service. Services should use the `UpdateSkippedSameVersion` message if the update was skipped due to the version of the image and the version on the device are the same. Services should use the `UpdateSkippedVersionTooLow` message if the update was skipped because the image version is lower than the minimum requirement.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "Device ' `<1>` ' skipped the update with image ' `<2>` '."

1. *string*: The URI or name of the device that skipped the update.
   - This argument shall contain the URI of the resource or a unique name of the device that skipped the update.
2. *string*: The identifier for the image.
   - This argument shall contain the identifier for the image.

2.24.25 UpdateSkippedSameVersion

Example: Device 'NIC1' skipped the update with image 'FWUpdate1' because the same version is already installed.

Indicates that a resource or device update was skipped because the same version is already installed.

- This message shall indicate that a resource or device update was skipped because the same version is already installed. If reinstallation of this version is required, clients may use the `ForceUpdate` parameter with the update action.

Version Added: v1.2

Severity: OK

Resolution: None.

Message and Arguments: "Device '`<1>`' skipped the update with image '`<2>`' because the same version is already installed."

1. *string*: The URI or name of the device that skipped the update.
   - This argument shall contain the URI of the resource or a unique name of the device that skipped the update.
2. *string*: The identifier for the image.
   - This argument shall contain the identifier for the image.

2.24.26 UpdateSkippedVersionTooLow

Example: Device 'NIC1' skipped the update with image 'FWUpdate1' due to the minimum version requirement.

Indicates that a resource or device update was skipped because the version of the provided image was lower than the minimum required version.

- This message shall indicate that a resource or device update was skipped because the version of the provided image was lower than the minimum required version. Some reasons include a security version requirements or hardware compatibility checks.

Version Added: v1.3

Severity: OK

Resolution: None.

Message and Arguments: "Device '<1>' skipped the update with image '<2>' due to the minimum version requirement."

1. *string*: The URI or name of the device that skipped the update.
   - This argument shall contain the URI of the resource or a unique name of the device that skipped the update.
2. *string*: The identifier for the image.
   - This argument shall contain the identifier for the image.

2.24.27 UpdateSuccessful

Example: Device 'Disk1' successfully updated with image 'FWUpdate1'.

Indicates that a resource or device was updated.

- This message shall indicate that a resource or device was updated.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Device '<1>' successfully updated with image '<2>'."

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.

2.24.28 VerificationFailed

Example: Verification of image 'FWUpdate1' at 'NIC1' failed.

Indicates that the service or component failed to verify an image.

- This message shall indicate that the service or component failed to verify an image. Some examples of image verification failure include: invalid signature, invalid checksum, or unknown image format.

Version Added: v1.0

Severity: Critical

Resolution: None.

Message and Arguments: "Verification of image '<1>' at '<2>' failed."

1. *string*: The identifier for the image, or the filename of the image if an identifier cannot be determined.
   - This argument shall contain the identifier for the image. If the identifier cannot be determined, this argument shall contain the filename of 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.

2.24.29 VerifyingAtComponent

Example: Image 'FWUpdate1' is being verified at 'NIC1'.

Indicates that a component is verifying an image.

- This message shall indicate that a component is verifying an image.

Version Added: v1.0

Severity: OK

Resolution: None.

Message and Arguments: "Image '<1>' is being verified at '<2>'."

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
2026.2	2026-08-06	Document build from Redfish message registries in DSP8011 release 2026.2.
2026.1	2026-04-02	Document build from Redfish message registries in DSP8011 release 2026.1.
2025.4	2025-12-04	Document build from Redfish message registries in DSP8011 release 2025.4.
2025.3	2025-09-04	Document build from Redfish message registries in DSP8011 release 2025.3.
2025.2	2025-05-01	Document build from Redfish message registries in DSP8011 release 2025.2.
2025.1	2025-02-05	Document build from Redfish message registries in DSP8011 release 2025.1.
2024.4	2024-11-27	Document build from Redfish message registries in DSP8011 release 2024.4.
2024.3	2024-08-01	Document build from Redfish message registries in DSP8011 release 2024.3.
2024.1	2024-04-02	Document build from Redfish message registries in DSP8011 release 2024.1.
2023.3	2023-11-30	Document build from Redfish message registries in DSP8011 release 2023.3.
2023.2	2023-08-03	Document build from Redfish message registries in DSP8011 release 2023.2.
2023.1	2023-04-06	Document build from Redfish message registries in DSP8011 release 2023.1.
2022.3	2022-12-08	Document built from Redfish message registries in DSP8011 release 2022.3.
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.