Swordfish Scalable Storage Management API Specification

Version: 1.2.2a

Abstract: The Swordfish Scalable Storage Management API defines a RESTful interface and a standardized data model to provide a scalable, customer-centric interface for managing storage and related data services.

SNIA Standard

This document has been released and approved by the SNIA. The SNIA believes that the ideas, methodologies, and technologies described in this document accurately represent the SNIA goals and are appropriate for widespread distribution. Suggestion for revision should be directed to http://www.snia.org/feedback/.

Last Updated: 14 June 2021

Contents

List of Tables

List of Figures

USAGE

DISCLAIMER

The information contained in this publication is subject to change without notice. The
SNIA makes no warranty of any kind with regard to this publication, including, but
not limited to, the implied warranties of merchantability and fitness for a particular
purpose. The SNIA shall not be liable for errors contained herein or for incidental or
consequential damages in connection with the furnishing, performance, or use.

Suggestions for revisions should be directed to http://www.snia.org/feedback/.

Current Revision

SNIA is actively engaged in expanding and refining the Swordfish documen-
tation. The most current revision can be found on the SNIA web site at
https://www.snia.org/tech_activities/standards/curr_standards/swordfish.

Contact SNIA

Current SNIA practice is to make updates and other information available through
their web site at http://www.snia.org.

FEEDBACK AND INTERPRETATIONS

Requests for interpretation, suggestions for improvement and addenda, or de-
fect reports are welcome. They should be sent via the SNIA Feedback Portal at

http://www.snia.org/feedback/ or by mail to the Storage Networking Industry Association, 4360 ArrowsWest Drive, Colorado Springs, Colorado 80907, U.S.A.

INTENDED AUDIENCE

This document is intended for use by individuals and companies engaged in storage management.

VERSIONING POLICY

This document is versioned material. Versioned material shall have a three-level revision identifier, comprised of a version number 'v', a release number 'r' and an errata number 'e'. Future publications of this document are subject to specific constraints on the scope of change that is permissible from one revision to the next and the degree of interoperability and backward compatibility that should be assumed between products designed to this standard. This versioning policy applies to all SNIA Swordfish versioned materials.

Version Number: Versioned material having version number 'v' shall be backwards compatible with all of revisions of that material that have the same version number 'v'. There is no assurance of interoperability or backward compatibility between revisions of a versioned material with different version numbers.

Release Number: Versioned material with a version number 'v' and release number 'r' shall be backwards compatible with previous revisions of the material with the same version number, and a lower release number. A minor revision represents a technical change to existing content or an adjustment to the scope of the versioned material. Each minor revision causes the release number to be increased by one.

Errata Number: Versioned material having version number 'v', a release number 'r', and an errata number 'e' should be backwards compatible with previous revisions of the material with the same version number and release number ("errata versions"). An errata revision of versioned material is limited to minor corrections or clarifications of existing versioned material. An errata revision may be backwards incompatible, if the incompatibility is necessary for correct operation of implementations of the versioned material. ### Revision History { - }

The evolution of this document is summarized in Table 1.

Table 1: Revision history

Date	Rev	Notes
19 September 2016	1.0.0	Initial Release
12 October 2016	1.0.1	Errata release for general clean up and formatting consistency
1 November 2016	1.0.2	Errata release to change multiple collections' types from collections (arrays) to ResourceCollections to conform to Redfish usage guidelines
		Change multiple collections' types from collections (arrays) to ResourceCollections to conform to Redfish usage guidelines and move NavigationProperties from Links section.
24 January 2017	1.0.3	Errata release to move complex types and enum to versioned namespace
		Schedule schema: add property
		json schema fix (Swordfish to swordfish)
		Specification enhancements, multiple areas
		User's guide: multiple new use cases and new document section
25 April 2017	1.0.4	Errata release with minor updates to schema: move FileShare collection, integrate DMTF and SNIA versions of Volume, fix incorrect property references and update descriptions. Update mockups. User's guide: Update cross-references.
3 October 2017	1.0.5	Errata release to include schema simplifications and other lessons from initial implementations, as well as general cleanup of specification.
13 February 2018	1.0.6	Updated Storage Systems model – added notion of Integrated Service Configuration in addition to (and named) Hosted Service Configuration
		Added ComplexType common definition section

Date	Rev	Notes
		Added/updated common Redfish property definitions
		Updates to conform to new SNIA templates.
12 October 2018	1.0.7	Enhanced Spare Capacity Management Model; Deprecated Remaining Capacity
		Added OpenAPI support: schema references and OpenAPI YAML files
		Added iSCSI properties for CHAP
		Event usage enhancements and guidance
		Volume schema updates – RAID Type enum (deprecating VolumeType usage), add ReplicaTargets
		Schema updates: Annotations enhancements: Capabilities designations, owning entities, Redfish.Required usage
		Clarified and updated ClassOfService IsDefault property usage
		Updated Capabilities location in hierarchy
		Fix cardinality issue of StorageReplicaInfo usage in StorageGroups and Volume
		Consolidate Client and Server Endpoint Groups into single Endpoint Group entity (deprecate usage of separate Client Endpoint Group and Server Endpoint Group)
		Add MappedVolume construct to StorageGroup – adds LUN info and other properties
		Clarified and updated ClassOfService IsDefault property usage
		Updated Capabilities location in hierarchy
		Fix cardinality issue of StorageReplicaInfo usage in StorageGroups and Volume

Date	Rev	Notes
		Consolidate Client and Server Endpoint Groups into single Endpoint Group entity (deprecate usage of separate Client Endpoint Group and Server Endpoint Group)
		Add MappedVolume construct to StorageGroup – adds LUN info and other properties
8 November 2018	1 .0.7a	Restored RAIDType property that was missing from 1.0.7
		Minor correction to schema versioning
22 August 2019	1.1.0	Restructured to add features and profiles
		Add description of SupportedFeatures usage and requirements
		Add requirements for subsets of Add language to clarify support for use with and without the class of service (now an optional feature)
		Added descriptions of support for seamless extension of Redfish Storage model to Swordfish
		Add updated model diagrams to reflect new model permutations
		Added descriptions of new constructs (e.g., Consistency Groups)
		Cleaned up references to Redfish Specification based on latest version
		Add Status Codes clarification and constraints section
12 November 2019	1.1.0	Released as Technical Position
12 November 2019	1 .1.0a	Released as Corrected Technical Position
		Formatting fixes – word wrap in pdf doc format to fix truncated lines
		Consistent object labeling in images (replace drive with disk)

Date	Rev	Notes
		Editorial and grammar changes and cleanup to status code guidance section
24 March 2020	1 .1.0b	Released as Corrected Technical Position
		TLS requirements now based on both ISO and SNIA standards
		Redfish references now based on both ISO and SNIA standards
		Bibliography added
29 May 2020	1.2.0	Note: This release is done in conjunction with the DMTF's Redfish Forum Work-in-Progress June 2020 release of DSP-IS0014 (v0.95), which contains multiple schema to support this work. Both are released as Working Drafts / work-in-progress for public review, and plan simultaneous releases in early fall 2020 to support full technical specification level capability and availability.
		Functionality availability in Swordfish includes:
		• Enhancements to Volume, StoragePools
		• New schema: NVMeDomain
		Other supporting documentation released in conjunction with this specification and schema bundle:
		• Multiple mockups reflecting multiple implementation permutation options (available on swordfishmockups.com)
		• Model overview documentation (NVMe to RF/SF Model Mapping Working Draft, dated May 2020)
18 August 2020	1.2.1	Note: This release is done in conjunction with the DMTF's Redfish Forum 2020.3 Release of the Redfish Specification, schema bundle and other supporting materials.

Date	Rev	Notes
		Functionality availability in Swordfish includes:
		• NVMe Mapping Support, Enhancements to Volume, StoragePools
		Additional Enhancements in the Specification and schema:
		• Added InitializeMethod property to Volume.
		• Made DedicateSpareDrives ReadWrite-able
		• Added enhanced Volume Access Capabilities and usage in StorageGroup.
		• Fix multiple URI issues across various schema.
		Updated formatting of tables to support automatic table numbering and ISO compatible table representation.
29 September 2020	1 .2.1a	Added bibliography and updated TLS references
20 October 2020	1 .2.1c	Updated with additional Redfish.URI annotations.
31 October 2020	1 .2.1c	Released as SNIA Standard
2 March 2021	1.2.2	Added sections to document use of complex types.
		Updated common properties sections.
		Schema changes:
		Add actions to Add and Remove drives directly from StoragePool.
		Split NVMeFirmwareImage and NVMeDomains schemas.
		Deprecate use of NetworkPort; replace with Port.
		Update Redfish.URI references.
		Corrected $ref references in JSON schema files.
		Fix incorrect references in deprecated JSON files.

About SNIA

The Storage Networking Industry Association (SNIA) is a non-profit organization made up of member companies spanning information technology. A globally recognized and trusted authority, SNIA's mission is to lead the storage industry in developing and promoting vendor-neutral architectures, standards and educational services that facilitate the efficient management, movement and security of information.

Acknowledgements

The SNIA Scalable Storage Management Technical Work Group, which developed and reviewed this work in progress, would like to recognize the significant contributions made by the following members listed in Table 2.

Table 2: Contributors

Member	Representatives (* – prior employer)
Broadcom Inc.	Richelle Ahlvers(*)
Cisco Systems, Inc.	Krishnakumar Gowravaram
Dell Inc.	Patrick Boyd
	George Ericson
	Jim Pendergraft
	Sean McGinnis
	Michael Raineri
	Rich Roscoe
Futurewei Inc.	Sean McGinnis(*)
Hitachi Data Systems	Eric Hibbard
Hewlett Packard Enterprise	Jeff Hilland
	Chris Lionetti
	John Mendonca
	Doug Voigt
Inova Development Inc.	Karl Schopmeyer

Member	Representatives (* – prior employer)
Intel Corporation	Richelle Ahlvers
	Rajalaxmi Angadi
	Phil Cayton
	Klaudia Jablonska
	Mariusz Krzywienski
	Mateusz Mania
	Slawek Putyrski
	Paul von Behren
Microsemi Corporation	Anand Nagarjan
Microsoft Corporation	Hector Linares
	Jim Pinkerton
	Michael Pizzo
	Scott Seligman
NetApp, Inc.	Don Deel
	Nilesh Maheshwari
ScienceLogic	Patrick Strick
VMware, Inc.	Murali Rajagopal

1 Abstract

The Swordfish Scalable Storage Management API ("Swordfish") defines a RESTful interface and a standardized data model to provide a scalable, customer-centric interface for managing storage and related data services. It extends the Redfish Scalable Platforms Management API Specification (DSP0266) from the DMTF.

2 Scope

2.1 Document Goals

Swordfish extends the Redfish Scalable Platforms Management API Specification to define a comprehensive, RESTful API for storage management that addresses block storage, file systems, object storage, and storage network infrastructure. It is centered around common operational and business concerns of storage management, including:

- Configuration and provisioning
- Monitoring
- Event and log management
- Performance assessment
- Diagnostics
- Fault detection and remediation
- Security
- Accounting and resource consumption

Swordfish's storage model is built around well-defined classes of service, which provide a means to map high-level business goals and objectives to specific, storage-based actions and requirements, in a clear and consistent way that can be applied uniformly across a broad spectrum of storage configurations and storage types (e.g., block storage, file systems, object stores). Common storage management functionality covered by class of service includes snapshots, replication, mapping and masking, and provisioning.

The Redfish specification provides the protocols and a core set of data models and behaviors for the management of systems. It defines the elements and behaviors that are mandatory for all Redfish implementations. Additionally it defines additional elements and behaviors that can be chosen by system vendors or manufacturers. The specifications also defines points at which OEM (system vendor) extensions can be provided by a given implementation. The specifications specifies normative requirements for Redfish Services and associated materials, such as Redfish Schema files. The Redfish specifications does not set requirements for Redfish clients, but will indicate what a Redfish client should do in order to access and utilize a Redfish Service successfully and effectively.

The Swordfish specification defines additional data models and behaviors for the management of storage systems and storage infrastructure. A Swordfish implementation shall conform to all requirements specified in the Redfish specifications.

Swordfish is suitable for a wide range of storage, from small-scale object drives, integrated RAID cards or RBODs providing storage services, to external disk arrays or file servers, to infrastructure providing storage services for converged, hyperscale and large scale cloud environments.

This document defines the Swordfish Scalable Storage Management API.

2.2 Audience Assumptions

As Swordfish is designed as an extension of the Redfish specification, this document is written with the presumption that the reader has a detailed understanding of the Redfish specification. This document cannot be fully understood without that context.

3 Normative References

3.1 Overview

The documents referenced in Table TBL_nn++ are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

3.2 Approved references

The approved references that contribute to this document are summarized in Table 3.

Table 3: Approved normative references

Tag	Title (Version)	A uthor	URL
I SO-8601	Data elements and interchange formats – Information interchange – Representation of dates and times – Part 1: Basic rules	IS O/IEC	<http://www.i so.org/iso/home/sto re/catalogue_ics/ca talogue_detail_ics. htm?csnumber=70907>
ISO -Direct	ISO/IEC Directives, Part 2Principles and rules for the structure and drafting of ISO and IEC documents (Seventh Edition, 2016)	IS O/IEC	h ttp://isotc.iso.org /livelink/livelink/ fetch/2000/2 122/4230450/4230456 /ISO_IEC_Directives _Part_2 _Principles_ and_rules_for_the_ s tructure_and_drafti ng_of_ISO_and_IEC_ documents_-_ 2016_%287th_editio n%29_-_PDF.pdf?no deid=17667902&vernu m=-2

Tag	Title (Version)	A uthor	URL
Redfish	Redfish Scalable Platforms Management API Specification (v1.11.0)	DMTF	<htt p://www.dmtf.org/si tes/default/files/s tandards/documents/ DSP0266_1.11.0.pdf>
OData	Open Data Protocol (v. 4.0)	OASIS	<https://w ww.oasis-open.org/s tandards#odatav4.0>
RFC3986	Uniform Resource Identifier (URI): Generic Syntax (2005)	The Int ernet So ciety	<htt p://www.rfc-base.or g/txt/rfc-3986.txt>
CSDL	Common Schema Definition Language (4.0)	OASIS	<http://docs.oa sis-open.org/odata/ odata/v4.0/odata-v4 .0-part3-csdl.html>
ITIL	ITIL Glossary (2011)	ITIL	h ttps://www.axelos.c om/Corporate/media/ Files/ Glossaries/ITIL_201 1_Glossary_GB-v1-0. pdf
Units	The Unified Code for Units of Measure (v2.0.1)	R egens trief Insti tute, Inc . and the UCUM Or ganiz ation	<http://units ofmeasure.org/trac>
IS O-20648	Information technology — TLS specification for storage systems	IS O/IEC	<ht tps://www.iso.org/s tandard/68622.html>

Tag	Title (Version)	A uthor	URL
SPC-4	SCSI Primary Commands - 4 (SPC-4) INCITS 513-2015	T10	<http://www.te chstreet.com/cgi-bi n/joint.cgi/incits>
F eatures	Swordfish Features Registry, version 1.0.1	SNIA	<https://r edfish.dmtf.org/reg istries/swordfish/v 1/SwordfishFeatureR egistry.1.0.1.json>
M essages	Swordfish Message Registry, version 1.0.2	SNIA	<https://redfi sh.dmtf.org/registr ies/swordfish/v1/Sw ordfish.1.0.2.json>
Ene rgyStar	ENERGY STAR Data Center Storage Version 1.1 Updated Program Requirements – April 1, 2019	EPA	https://www.ene rgystar.gov/sites/d efault/files/ENERGY STAR Data Center Storage Final Version 1.1 Specification Rev. April 2019.pdf

3.3 References under development

Documents referenced in Table 4 are under active development, and subject to revision or replacement at any time. In the event that the provided URL is no longer valid, refer to the related parent page to locate a replacement.

Table 4: References under development

Tag	Title (Version)	Author	URL	Parent Page
RedfishResource	Redfish Resource and Schema Guide	DMTF	\<http://www.dmtf.org/sites/default/files/standards/documents/DSP2046_2017.0a.pdf>	\<http://www.dmtf.org/redfish>

3.4 Other references

None defined in this document.

4 Terms and Definitions

4.1 Overview

In this document, some terms have a specific meaning beyond the normal English meaning. Those terms are defined in this clause. New terms, frequently used Redfish terms.

4.2 Swordfish-specific Terms

4.2.1 Definitions

The terms listed in Table 5 are used in this document.

Table 5: Swordfish terms

Term	Definition
Entity	An instance of a schema element.
Model	A set of entities and the relationships between them that define the semantics, behavior and state of that set.
OData service	A REST-based service that allows resources, identified using Uniform Resource Locators (URLs) and defined in a model, to be published and edited by Web clients using simple HTTP messages.
Resource	A central element in a model, which represents a physical construct or a logical service, and is further defined by other model entities.
Schema	A formal language representation of a model that conforms to a metamodel.
Service Document	A particular resource that is directly accessed via an OData service entry point.This resource serves as a starting point for locating and accessing the other resources and associated metadata that together make up an instance of a Swordfish service.
Swordfish service	An extension to the Redfish Service that conforms to the Swordfish specification, and provides REST-ful storage management functionality.

4.2.2 Symbols and abbreviated terms

None in this document.

4.3 Reference to Redfish terms

Many terms in this document were originally defined in the Redfish Specification. Some of the more common terms and definitions are reproduced in Table 6, as an aid to the reader.

Table 6: Redfish terms

Term	Definition (as of 16 August 2019)
OData	The Open Data Protocol, as defined in OData-Protocol.
OData Service Document	Resource that provides information about the service root for generic OData clients.
Redfish Schema	Defines Redfish Resources according to OData schema representation. You can directly translate a Redfish Schema to a JSON Schema representation.
Redfish service	Implementation of the protocols, resources, and functions that deliver the interface that this specification defines and its associated behaviors for one or more managed systems.
Request	A message from a client to a service.
Service Root	Resource that serves as the starting point for locating and accessing the other resources and associated metadata that together make up an instance of a Redfish Service.

4.4 Keywords (normative language terms)

This document conforms to ISO/IEC Directives, Part 2 for keyword usage. The most common terms and their intended meanings are summarized Table 7.

Table 7: Normative language terms

Term(s)	Meaning
shall / shall not	Used to identify objectively verifiable criteria to be fulfilled and from which no deviation is permitted if compliance with the document is to be claimed
should / should not	Used to identify a suggested possible choice or course of action deemed to be particularly suitable without necessarily mentioning or excluding others
may / need not	Used to convey consent or liberty (or opportunity) to do something
can / cannot	Expected or conceivable material, physical or causal outcome
must	Identifies a constraint or obligation on the user of the document, typically due to one or more legal requirements or laws of nature, that is not stated as a provision of the standard_NB:_ "must" is not an alternative for "shall", and should only be used for constraints that arise from outside this standard

5 Swordfish Overview

5.1 Introduction

The Swordfish Scalable Storage Management API ("Swordfish") defines a RESTful interface and a standardized data model to provide a scalable, customer-centric interface for managing storage and related data services. It extends the Redfish Scalable Platforms Management API Specification (DSP0266) from the DMTF.

5.2 Relation to Redfish
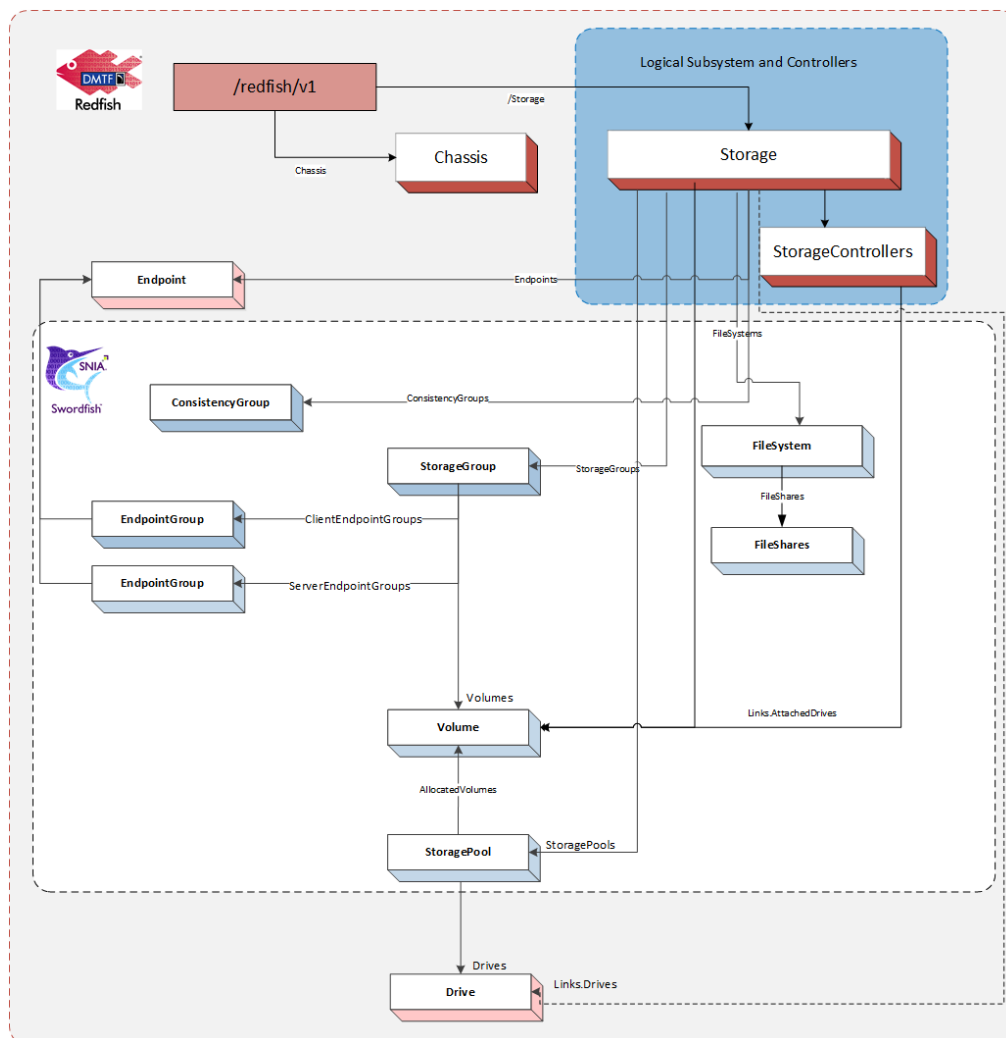
Figure 1: Model Overview

The Swordfish service interface extends the Redfish service interface. As such, a Swordfish service is a Redfish service and includes all required elements of the Redfish model, as illustrated by Figure 1.

The storage systems shall be instantiated in one of two places in the hierarchy: - directly in the `Storage` resource collection, or - attached to a `ComputerSystems`, with an associated reference link in the `StorageSystems` resource collection at the Service Root. In this case, there shall also be a reference link to the `Storage` resource in the `Storage` resource collection at the Service Root.

As a result, a Swordfish client is always to locate the storage systems managed by the Swordfish service in the `ServiceRoot` via the `Storage` resource collection; this may be a combination of references to instances and instantiated instances.

The physical infrastructure is modeled using Redfish `Chassis`.

As modeling for storage systems may cover both logical and physical constructs, Swordfish management clients that are focused on logical storage management use cases may choose to manage functionality entirely by way of logical resources.

Each Swordfish service is accessed via well known URLs on the system supporting the Swordfish Service. Since Swordfish is an extension of Redfish, these URLs are the same as for accessing the Redfish defined aspects of the service.

5.3 Storage System Models

Swordfish has been designed to support a broad range of configurations, requirements, size and complexity, as well as logical and physical architectures. As a result, there are two primary methods of modelling the storage system for a Swordfish implementation:

1. Swordfish Standalone Configuration

The standalone configuration instantiates the logical storage system instance representation in the `Storage` resource collection directly off the Service Root. The logical storage system is modeled using the Redfish `Storage` and `StorageController` resources, as shown in Figure 2. Managed resources are connected to the `Storage` resource, including `Volumes` and `StoragePools`.
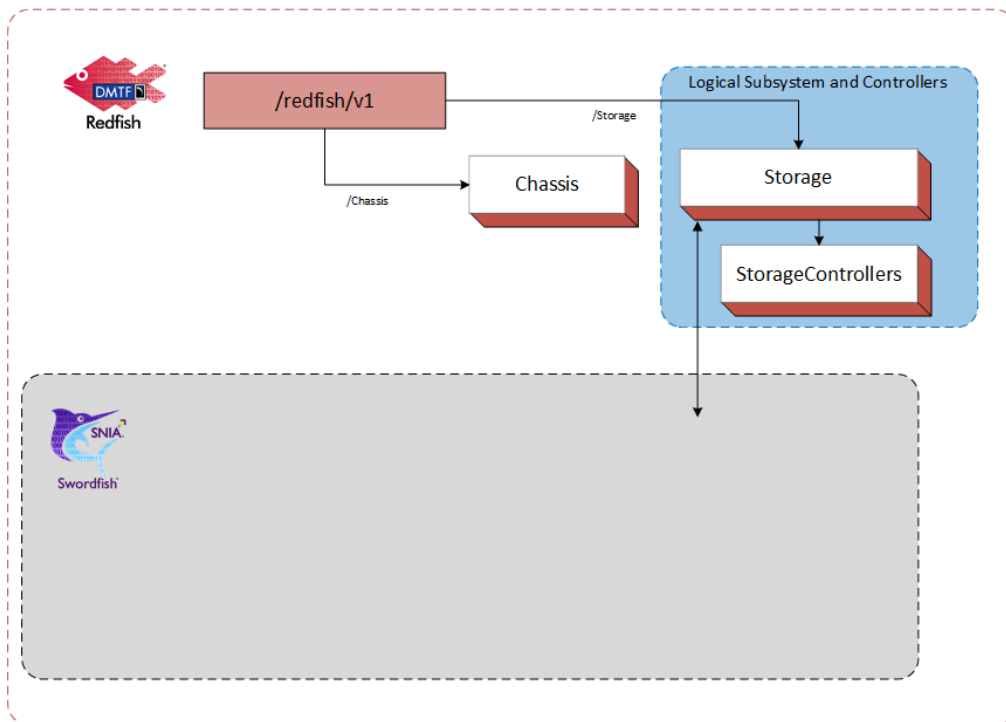
Figure 2: Logical Subsystem in Swordfish Standalone Configuration

This configuration works well for standalone devices or storage systems. An example of a Storage System for an standalone configuration is shown in Figure 3.
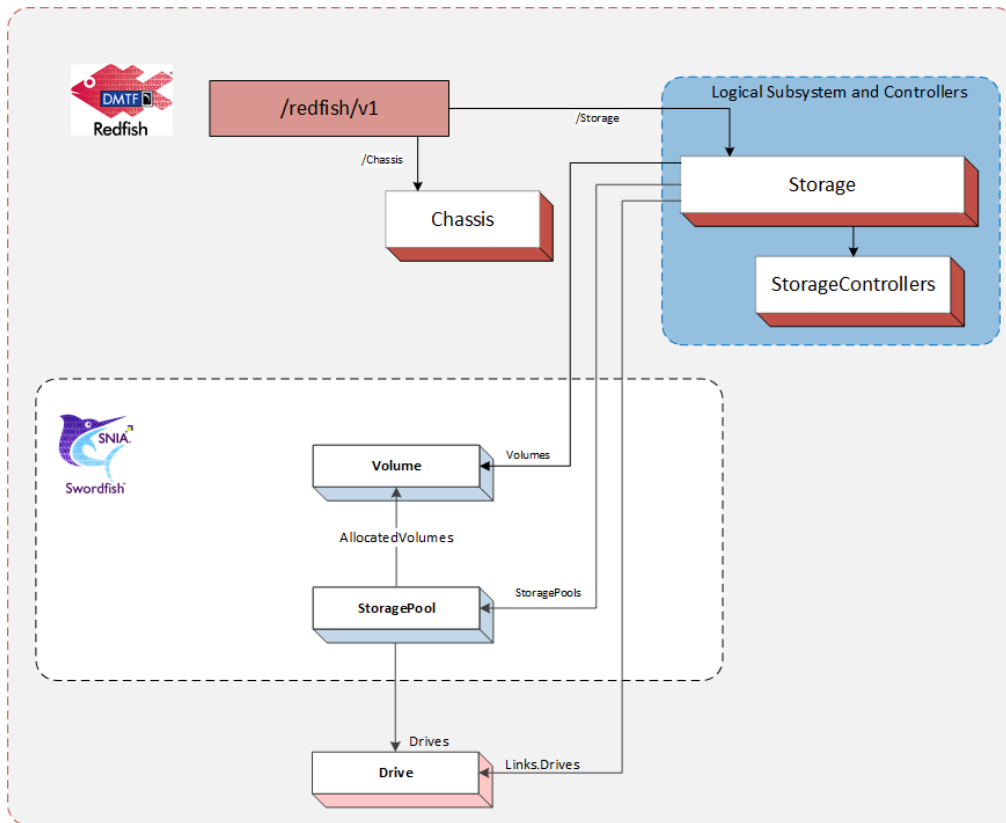
Figure 3: Swordfish Standalone Configuration Example

2. Swordfish Integrated Configuration

The integrated configuration attaches to the `Storage` collection within the same ComputerSystem model instantiation as the server where the physical element resides.

The logical storage system is modeled using the Redfish `Storage` and `StorageController` resources. The `Storage` resource is located in the Redfish hierarchy contained by `ComputerSystems`, typically running as ApplicationServers. The physical infrastructure is modeled using Redfish `Chassis`. Managed resources are connected to the `Storage` resource, including `Volumes` and `StoragePools`.

The integrated configuration is illustrated in Figure 4.
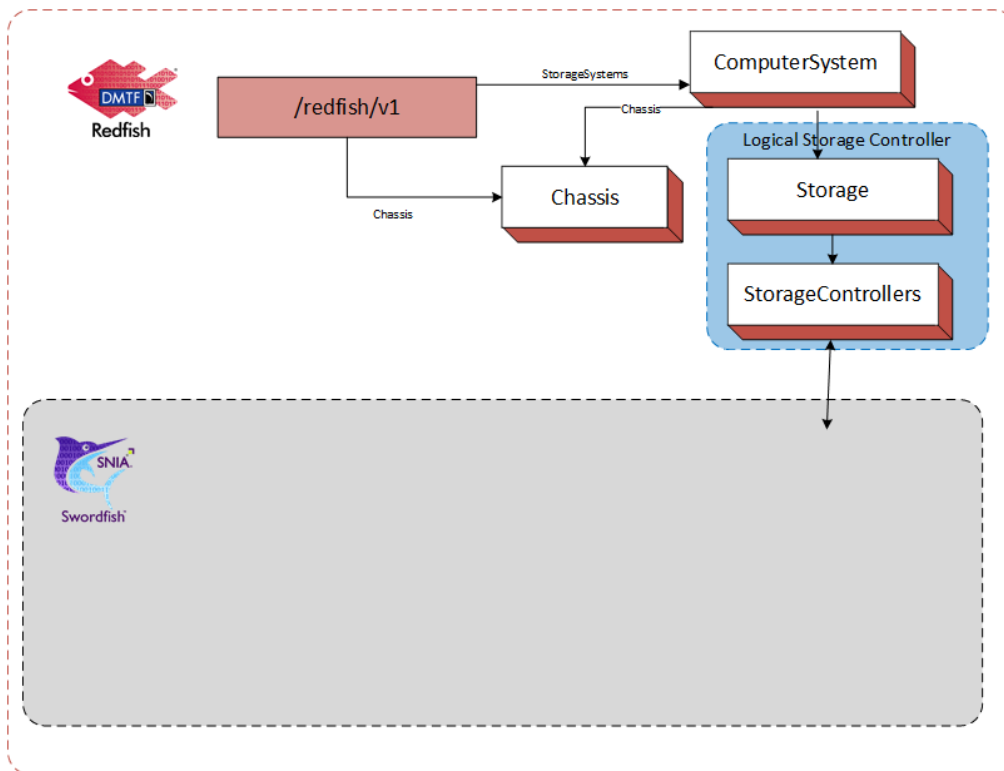
Figure 4: Logical Subsystem in Swordfish Integrated Configuration

This configuration works well when the storage system can be modeled by simply instantiating a new Storage object within an existing computer system. An example of a Storage System for an integrated configuration is shown in Figure 5.
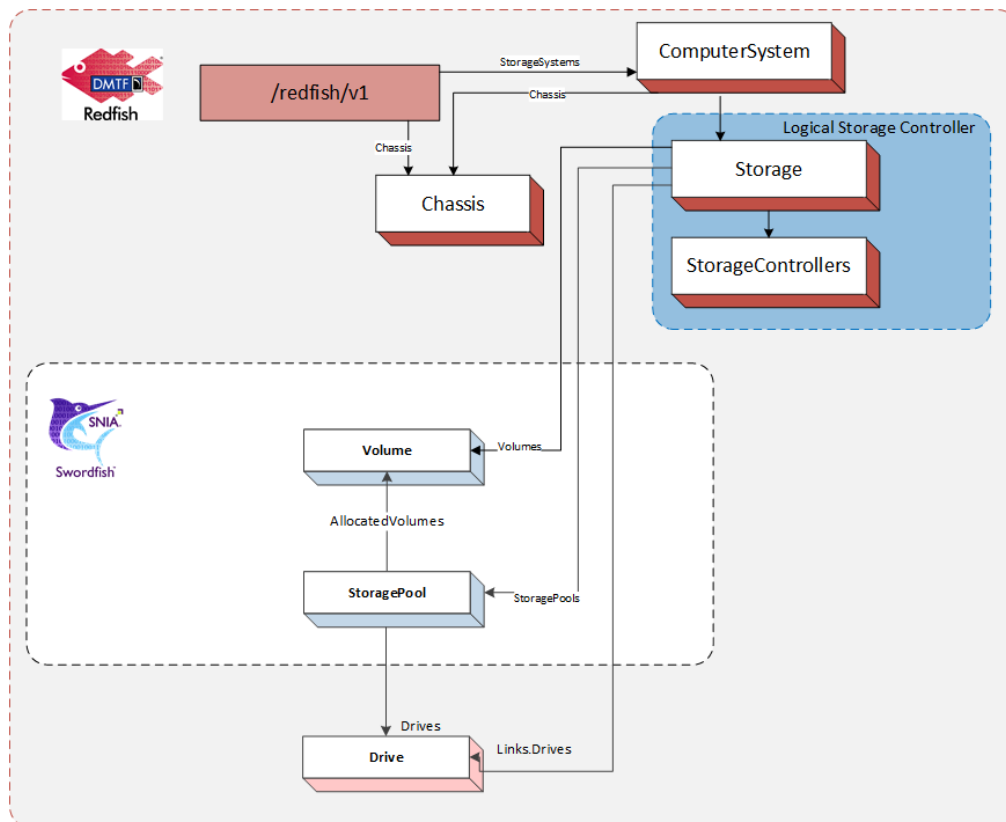
Figure 5: Swordfish Integrated Configuration Example

5.4 The ServiceRoot and ServiceContainer entities

5.4.1 Overview

A **GET** of `/redfish/v1` will return the `ServiceRoot` entity. A **GET** of `/redfish/v1/odata` will return the `ServiceContainer` instances that represents the OData service document. Each of these instances provides links to the remainder of the system.

The following are the elements utilized for Swordfish management.
- `Storage`: A reference to the Storage resource collection. - `Systems`: A reference to a Systems resource collection; - `Chassis`: A reference to a Chassis resource collection; - `StorageSystems`: A reference to a StorageSystems resource collection.

5.4.2 The Storage resource collection

A resource collection that references a set of `Storage` resources that each represents a storage subsystem. This collection can contain either resources or references to

instances of `Storage` resources. Each `Storage` resource represents an instance of a storage subsystem. For Swordfish subsystems, refer to the details in the Swordfish model overview for details on required elements.

5.4.3 The Systems resource collection

A resource collection that references a set of `ComputerSystem` resources that each represents a general purpose application server. Each `ComputerSystem` resource will have an entry with the value of "ApplicationServer" in its `HostingRoles` property. A particular `ComputerSystem` resource can be in both the StorageSystems collection and the Systems collection.

5.4.4 The Chassis resource collection

A resource collection that references a set of `Chassis` resources. Each Chassis resource represents physical containers, (i.e. sheet-metal confined spaces and logical zones like racks, enclosures, chassis and all other containers). Subsystems (like sensors), which operate outside of a system's data plane (meaning the resources are not accessible to software running on the system) are linked either directly or indirectly through this resource.

5.4.5 The StorageSystems resource collection

A reference to a `ComputerSystemCollection` with members of type `ComputerSystem` that support storage services. These `ComputerSystem` resources represent systems that support Swordfish storage management services. They will have an entry with the value of "StorageServer" in their `HostingRoles` property. This collection, then, is a resource collection that references a set of `ComputerSystem` resources that each represents a storage server. Each `ComputerSystem` resource will have an entry with the value of "StorageServer" in its `HostingRoles` property. A particular `ComputerSystem` resource can be a member of both the `StorageSystems` resource collection and the `Systems` resource collection.

5.5 Swordfish model overview

5.5.1 The Storage resource

The storage system exposes logical storage, associated resources and related functionality. Storage resources can be found in the service root or service container via the `Storage` resource collection, and are attached to the `Storage` object within the `Storage` resource collection.

The storage system typically provides the ability to create, manage and present block, file or object store from a set of back-end media, presented to one or more hosts. Storage controllers can work in coordinated sets of one or more to present value-add capabilities, such as failover, data protection, and data path management within the storage system, that are represented through the various resources within the storage system.

The following are the principal properties of `Storage` that point to resources managed or defined by the storage system:

- `Controllers`: A reference to a resource collection that collects `StorageController` resources.
- `Drives`: A reference to a collection that collects `Drive` resources used for storage.
- `Enclosures`: A reference to a resource collection that collects Chassis resources that contain storage related resources.
- `Endpoints`: A reference to a resource collection that collects `Endpoint` resources used to access storage.
- `EndpointGroups`: A reference to a resource collection that collects `EndpointGroups` resources.
- `FileSystems`: A reference to a resource collection that collects `FileSystem` resources.
- `StorageGroups`: A reference to a resource collection that collects `StorageGroup` resources.
- `ConsistencyGroups`: A reference to a resource collection that collects `ConsistencyGroup` resources.
- `StoragePools`: A reference to a resource collection that collects `StorageGroup` resources.
- `Volumes`: A reference to a resource collection that collects `Volume` resources.

5.5.1.1 The StorageController resource   The storage controller presents the foundational resources used by the storage system. It generally contains connectivity

resources between the system and connected consumers.

5.5.1.2 The Endpoint resource   Endpoints represent one end of a protocol specific connection that supports sending or receiving messages according to a particular protocol.

5.5.1.3 The Endpoint Collection resource   The `Endpoint` Group is resource collection that references a set of `Endpoint` resources.

5.5.1.4 The ConsistencyGroup resource   ConsistencyGroups represent a set of volumes that are managed consistently and collectively as a group, to allow system and application level activities to be performed on a set of data that spans volumes. This activities include device-level replication activities as well as system level functions, such as reset.

When ConsistencyGroups are implemented, they are attached to a `Storage` resource and its internal `Volumes` collection is constructed from a subset of the `Volumes` collection of the `Storage` resource.

5.5.1.5 The ConsistencyGroup Collection resource   The `ConsistencyGroupCollection` is a resource collection that references a set of `ConsistencyGroup` resources.

5.5.1.6 The StorageGroup resource   StorageGroups represent a set of volumes that are managed as a group in order to facilitate mapping and masking, in which the volumes of a storage group are collectively exposed or hidden to a set of clients.

The set of volumes is specified by the `Mapped Volumes` attribute. `MappedVolumes` is a resource collection of the `Mapped Volume` construct (a tuple of a pointer to a volume and a corresponding Logical Unit Number for that volume).

The set of client endpoints to which the volumes can be exposed is specified by the `ClientEndpointGroups` attribute. The `ClientEndpointGroup` resource specifies a collection of `EndpointGroup` resources.

The set of server endpoints to which the volumes can be exposed is specified by the `ServerEndpointGroups` attribute. The `ServerEndpointGroup` resource specifies a collection of `EndpointGroup` resources.

5.5.1.7 The StoragePool resource   The `StoragePool` resource represents unassigned storage capacity that can be used to produce storage volumes or other storage pools.

The following are the principal properties of `StoragePool` that are used to create or identify resources provisioned or supported by the storage pool:

- `AllocatedVolumes`: A reference to a resource collection that collects `Volume` resources that have been provisioned from the storage pool.
- `AllocatedPools`: A reference to a resource collection that collects `StoragePool` resources that have been provisioned from the storage pool.
- `CapacitySources`: A reference to a resource collection that provides pointers to the capacity sources that are used to provide the underlying capacity for this storage pool.
- `RAIDTypes[]`: The set of RAIDTypes supported by this StoragePool. This may be set upon StoragePool creation, or may be a reflection of the implementation's ability to support different RAID types. Consumers may use this property to determine what RAID types are available from specific StoragePool instances for additional Volume creation requests, or what RAIDTypes have been applied to Volumes already allocated.

5.5.1.8 The Volume resource   `Volume` resource represents a block-addressable container of storage, sometimes referred to as a "Logical Unit", "LU", "LUN", or "StorageVolume" in the storage industry.

5.5.1.9 The FileSystem resource   This `FileSystem` resource represents a file system. Each `FileSystem` may contain a collection of `FileShares` that can be presented to hosts.

6 Features and Profiles

6.1 Overview

Features are high-level descriptions of functionality which an implementation uses to advertise what functionality it currently supports, and for some features, is capable of supporting.

The detailed definitions required to describe to implementers how to implement a feature are written in profile definition files. A feature is generally represented in one (but may be more) profile definition file, or profile.

Profiles are detailed descriptions that describe down to the individual property level what functionality is required in order to advertise features. Different profile definitions can exist for the same feature type but for various types of storage configurations: Swordfish.Block.Provisioning, Swordfish.File.Provisioning

The Swordfish Features Registry shall be used to advertise what standard and Oem Features an implementation supports.

6.2 Requirement for SupportedFeatures

`SupportedFeatures` entries in the `Features` registry represent the client's primary initial runtime view of the capabilities of a Swordfish implementation. Without properly formed entries in this registry, there is no visibility to an implementation's functionality.

Swordfish implementations shall implement the `Features` registry and advertise at least the `SNIA.Swordfish.Discovery` supported feature in order to be considered a Swordfish implementation.

Features define coarse-grained sets of functionality. In order to advertise a feature (using the `SupportedFeature` mechanism in the SupportedFeatures Registry), the implementation must support the complete set of functionality as defined in the corresponding profile.

The Swordfish Features Registry publishes the official list of supported SNIA Features, and provides a high-level description of their functionality. Many of those features are self-explanatory (e.g., local replication, remote replication), but there are some features where additional context is appropriate:

- Class of Service
- Energy Star for Storage

6.3 EnergyStar for Storage Feature

The EnergyStar for Storage Feature and profile has been created to formalize the requirements from the ENERGY STAR Data Center Storage Program Requirements on storage products. The profile indicates what properties Swordfish implementations need to support in order to properly instrument EnergyStar reporting capability. This functionality is intended to support EnergyStar data gathering requirements as part of the EnergyStar certification process.

6.4 Class of Service Feature

6.4.1 Overview

Swordfish supports a `ClassOfService` feature. The `ClassOfService` functionality supports systems that are capable of providing a greater level of management automation, where a higher-level set of goals is provided as direction rather than requiring parameterized inputs for all configuration actions.

The Class of Service feature uses a combination of device-defined capabilities to structure LinesOfService, which are sets of available functionality in a given system, that can then be grouped together to provide classes of service.

When Class of service functionality is implemented, the Swordfish functionality may be entirely exposed through the StorageService resource. Each Swordfish StorageService is located in the ServiceRoot (and ServiceContainer) via the StorageServices resource collection.

6.4.2 Class of Service Model

For Swordfish with a class of service interface, the following two models apply. Either model choice results in the same storage service, regardless of the storage system model.

  1. Integrated Service Configuration

The storage systems managed by the Swordfish storage service are modeled using the Redfish `Storage` resource and `StorageController` resource collections. The `Storage` resource is located in the Redfish hierarchy contained by `ComputerSystems`, typically running as ApplicationServers. The physical infrastructure is modeled using Redfish `Chassis`, as shown in Figure 6.
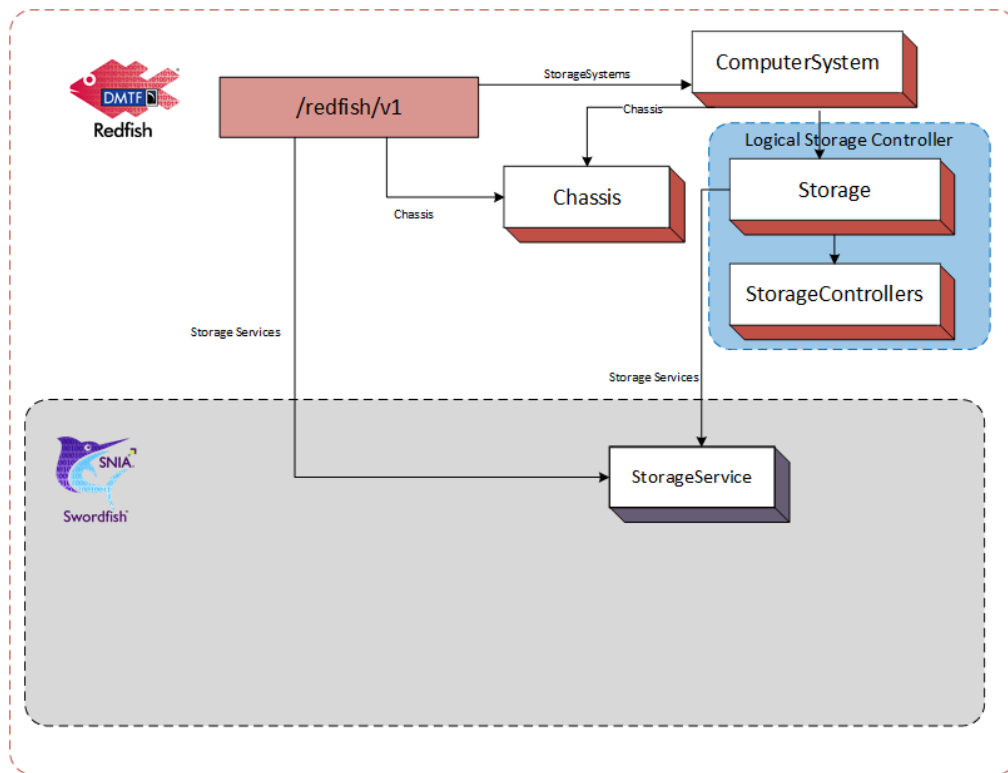
Figure 6: Logical Subsystem in Integrated Service Configuration

This configuration works well when the storage service is hosted by a storage resource within a computer system. An example of a Storage Service for an integrated service configuration is shown in figure 7.

> Note: This diagram and the discussion of the configuration description have been simplified slightly to avoid confusion. A full implementation would likely include additional links to the logical storage controller resources.
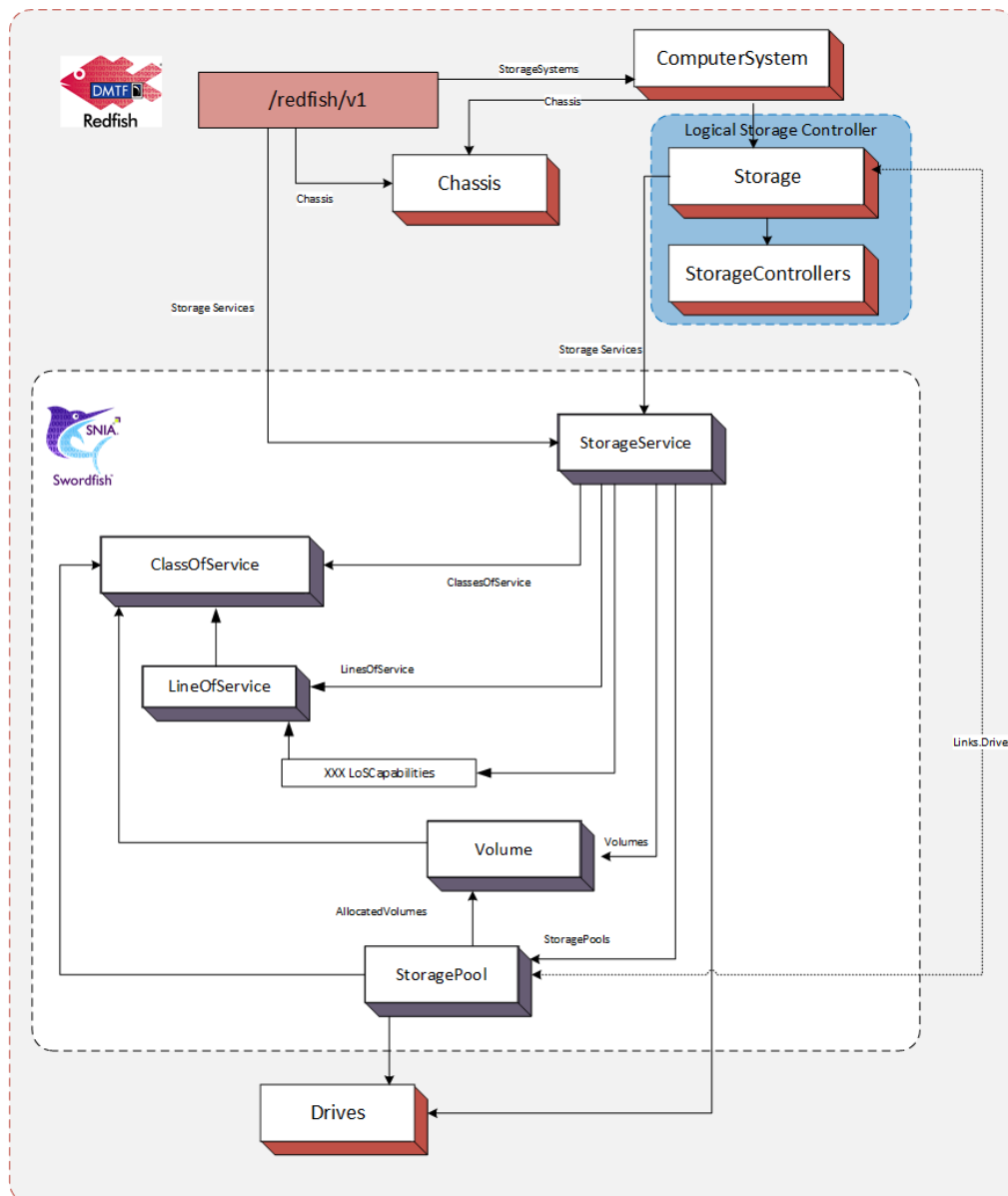
Figure 7: Integrated Service Configuration Example

2. Standalone Service Configuration

The storage systems managed by the Swordfish storage service are located in the `ServiceRoot` (and `ServiceContainer`) via the `Storage` resource collection. They model the logical storage system using Redfish `Storage` and 'StorageController' resources. The physical infrastructure is modeled using Redfish `Chassis`. This is shown in Figure 8.
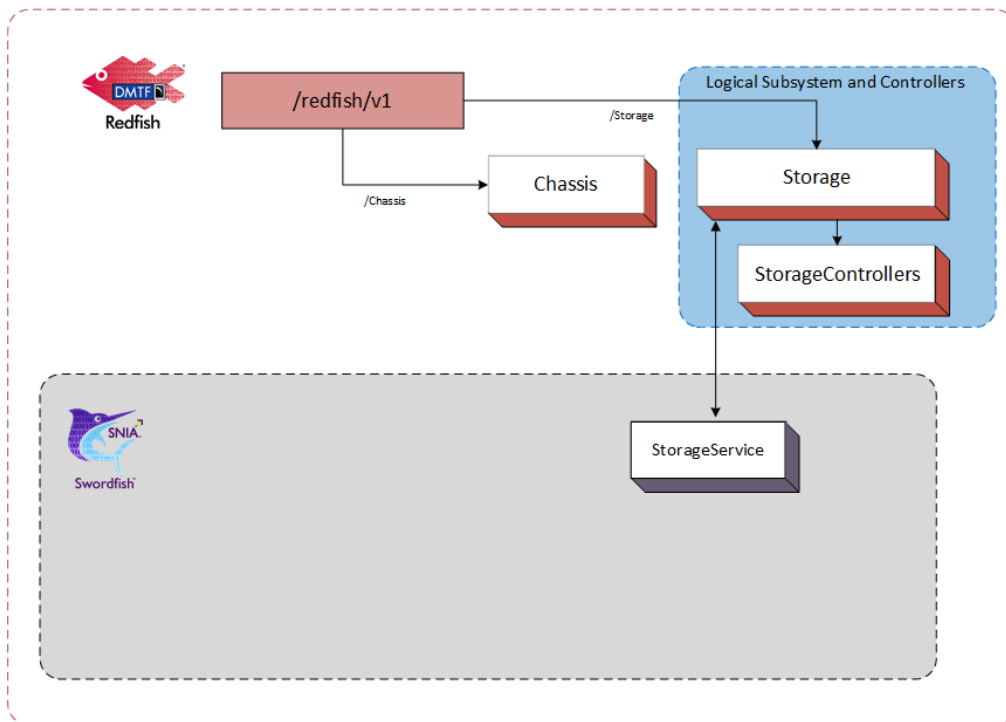
Figure 8: Logical Subsystem in Standalone Service Configuration

This configuration works well when the standalone storage system directly hosts the storage service(s). An example of a Storage Service for a hosted service configuration is shown in Figure 9.

> Note: This diagram and the discussion of the configuration description have been simplified slightly to avoid confusion. A full implementation would likely include additional links to the logical storage controller resources.
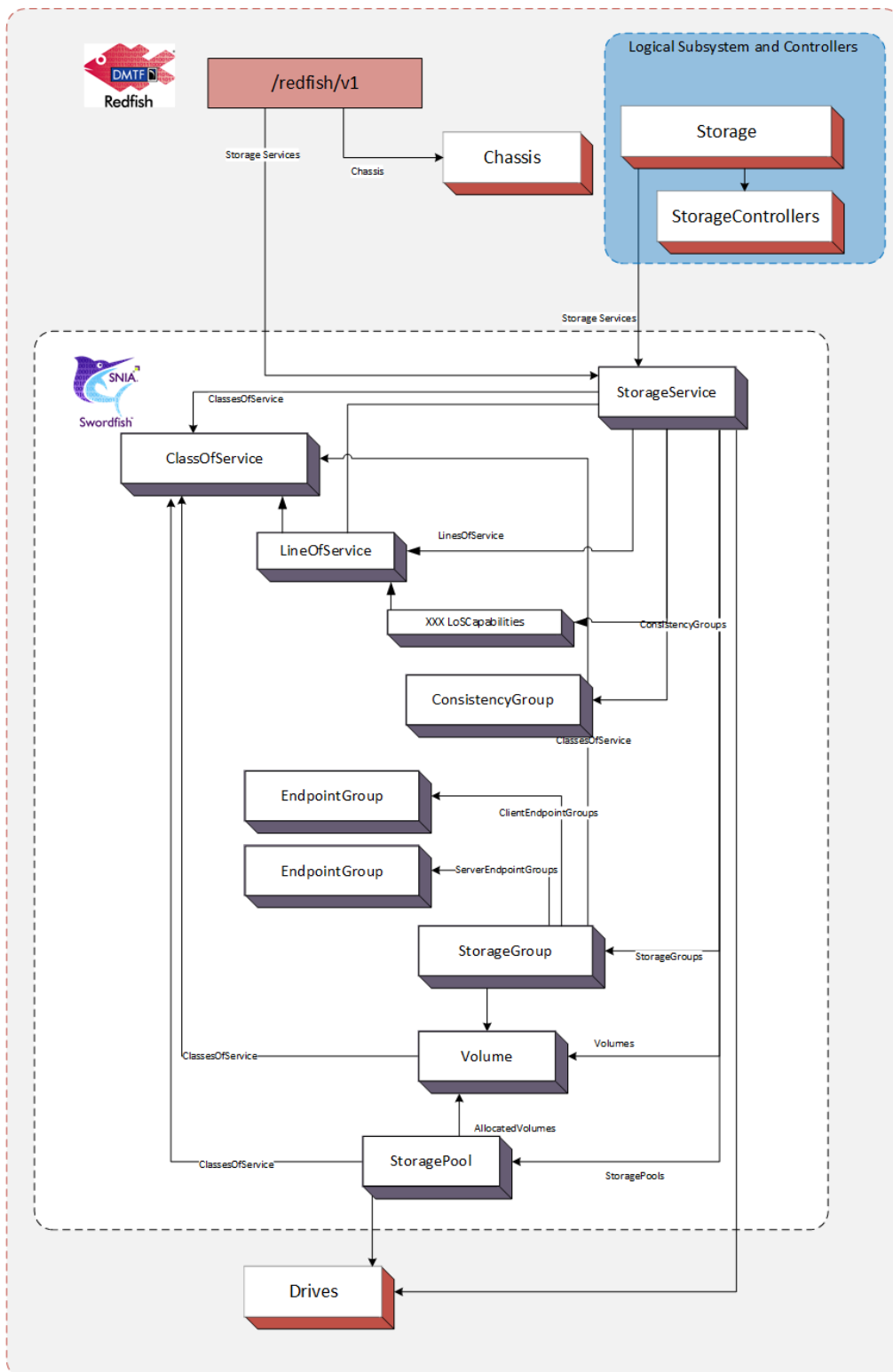
Figure 9: Standalone Service Configuration Example

6.4.3 ServiceRoot Additions

When the StorageService feature is implemented, the following is added to the `ServiceRoot`:

- `StorageServices`: A resource collection that references a set of `StorageService` resources. Each `StorageService` resource represents the resources and behaviors supported by that storage service.

6.4.4 The StorageService resource

6.4.4.1 Principal Properties   The storage service is hosted on a storage system and exposes logical storage, associated resources and related functionality. Storage service resources can be found in the service root or service container via the `StorageServices` resource collection.

The following are the principal properties of `StorageService` that point to resources managed or defined by the storage service:

- `ClassesOfService`: A reference to a resource collection that specifies the supported `ClassOfService` resources.
- `Drives`: A reference to a resource collection that collects `Drive` resources used for storage.
- `Enclosures`: A reference to a resource collection that collects Chassis resources that contain storage related resources.
- `Endpoints`: A reference to a resource collection that collects`Endpoint` resources used to access storage.
- `FileSystems`: A reference to a resource collection that collects `FileSystem` resources.
- `EndpointGroups`: A reference to a resource collection that collects `EndpointGroups` resources.
- `StorageGroups`: A reference to a resource collection that collects `StorageGroup` resources.
- `StoragePools`: A reference to a resource collection that collects `StorageGroup` resources.
- `Volumes`: A reference to a resource collection that collects `Volume` resources.
- `HostingSystem`: A reference to the `ComputerSystem` instance that hosts this `StorageService`.

6.4.4.2 Capabilities and Lines of ServiceRoot   The following properties each define a set of attributes, which describe capabilities that the storage service may support:

- `DataProtectionLoSCapabilities`: Replicas that protects data from loss.
- `DataSecurityLoSCapabilities`: Data security service level requirements. The data security characteristics enable the storage system to be used in an environment where compliance with an externally-specified security standard or standards is required. Examples of such standards include FIPS-140, HIPAA and PCI.
- `DataStorageLoSCapabilities`: Provisioning and access characteristics for storage of the data.
- `IOConnectivityLoSCapabilities`: IO connectivity requirements for access to the data.
- `IOPerformanceLoSCapabilities`: IO performance requirements for access to the data.

In each of the above, not all combinations of attribute values defined within a capability are likely to be supported by the storage service.

Known, supported combinations of attribute values are used to construct entries in the `LinesOfService` array property. Not all attributes of a line of service entry need be specified (i.e. some may be Null). If an attribute has no value, the storage service may choose any supported values when provisioning for that entry. Otherwise, the line of service attribute values specifies the kind or level of service to be provided.

6.4.4.3 The ClassOfService resource   A class of service represents a choice of utility or warranty offered to customers by a service. (ITIL uses the term `service option`. See the Normative References.)

Each `ClassOfService` resource is a uniquely named description of the characteristics of one choice of utility or warranty for a service. Each `ClassOfService` is a description of the kind and quality of service to provide and is not intended to describe how the service provides that service.

Each `ClassOfService` is defined by an aggregation of lines of service. Supported lines of service are listed in the corresponding capabilities attributes of the storage service, (see above).

Currently defined lines of service are:

- Data Protection: Describes the characteristics of a replica that protects data from loss.

- Data Security: Describe data security service level requirements. The data security characteristics enable the storage system to be used in an environment where compliance with an externally-specified security standard or standards is required. Examples of such standards include FIPS-140, HIPAA and PCI.
- Data Storage: Describes provisioning and access characteristics for storage of the data.
- IO Connectivity: Describes IO connectivity requirements for access to the data.
- IO Performance: Describes the IO performance requirements for access to the data under a particular workload.

Some advertised `ClassOfService` resources are created by the service implementation. These are generally not changeable and are intrinsic to the implementation.

A service may support creation or modification of `ClassOfService` resources. All must be consistent with the capabilities of the service.

6.4.4.4 The StoragePool resource   When a Swordfish implementation advertises support for the Class of Service feature, the `StoragePool` resource now presents a new method to the client to allocate unassigned storage capacity. This is automated by the system as conformance to one or more classes of service. Requests to `StoragePool` shall automatically allocate capacity based on the constraints of the selected class of service and any other selected parameters, with priority given to the class of service constraints.

The following are the principal properties of `StoragePool` that are used to identify resources provisioned or supported by the storage pool related to Class of Storage:

- `ClassesOfService`: A reference to a resource collection that specifies the set `ClassOfService` resources that can be specified when provisioning resources from the storage pool.
- `DefaultClassOfService`: A reference to the default `ClassOfService` resources used for provisioning from the storage pool.

6.4.4.5 The Volume resource   `Volume` resource represents a block-addressable container of storage, sometimes referred to as a "Logical Unit", "LU", "LUN", or "StorageVolume" in the storage industry. Volumes optionally adhere to a `ClassOfService`, which defines added functionality. Examples include:

- Access capabilities
- Capacity and capacity sources

- Consumption tracking (e.g., LowSpaceWarningThresholdPercents)
- Replication details
- StorageGroup Information

6.4.4.6 The FileSystem resource   In a Swordfish implementation that advertises support for the Class of Service feature, File systems represent file-addressable capacity that are conformant to a `ClassOfService`.

7 Schema Considerations

7.1 Schema Introduction

7.1.1 Overview

A Swordfish implementation is a Redfish implementation, and as such it minimally includes support for some Redfish-defined schema, including ServiceRoot and ComputerSystem. Swordfish implementations include support for Swordfish-defined schema. The Swordfish model focuses primarily on the logical model of a storage system and does not require full representation of a physical instantiation. This is left to the implementer to complete from available Redfish schema models.

Swordfish schema is conformant with the rules used to define Redfish schema. Redfish schema is conformant with the Common Schema Definition Language, see CSDL. This section provides additional definition and context for the CSDL elements used to define Swordfish schema.

7.1.2 Swordfish Extension of the Redfish ServiceRoot

The Redfish `ServiceRoot` has properties that provide access to Swordfish resources.

The first is `StorageSystems`. This property references a collection of `ComputerSystem` resources that each support Swordfish functionality. Each such `ComputerSystem` shall have an entry in its `HostingRoles` property with the value of `StorageServer`.

For implementations that advertise support for the `ClassOfService` feature, the implementation shall instantiate a collection of `StorageServices`at the `ServiceRoot` with at least one member. The collection provides the client an efficient means to search across all `StorageService` resources, regardless of which `ComputerSystem` is supporting the service.

7.2 Default values and NULLABLE attributes

The interaction of `Nullable` and `DefaultValue` needs to be clearly understood by both implementers and client developers. The possible combinations of are summarized in Table 8. The table contains:

- **Nullable**: True, if a given property may be NULL
- **DefaultValue**: True, if a default value is provided for a given property

- **Client**: True, if a client value is supplied for a given property in a query or response
- **Result**: The resultant value of the given property. One of:
    - *C*: The client-provided value
    - *D*: The default value
    - *Null*: Null
    - *I*: Implementation defined
    - *Error*: Error state

Table 8: Default and Nullable Interaction

Nullable	DefaultValue	Client	Value
T	T	T	C
T	T	F	D
T	F	T	C
T	F	F	I or Null
F	T	T	C
F	T	F	D
F	F	T	C
F	F	F	I or Error

7.3 Common schema annotations

Table 9 lists common annotation used in the definition of Swordfish, for details see OData Capabilities Vocabulary, OData Core Vocabulary, OData Measures Vocabulary, and Redfish Extensions.

Table 9: Schema annotations

Name	Applies to	Description
AllowableValues	Parameter	The set of allowable values for a parameter

Name	Applies to	Description
AutoExpand	NavigationProperty	If true, return expand the target element
AutoExpandReferences	NavigationProperty	If true, return references to the target element
ConformanceLevel	EntityContainer	Specifies OData conformance level
Deprecated	All	Specifies that the element may be removed in future major revisions, but shall continue to be supported as specified in the current revision.
Description	All	A brief description of a model element
LongDescription	All	A normative description of a model element
Maximum	Parameter, Property	Maximum value that an integer property or parameter may have
Minimum	Parameter, Property	Minimum value that an integer property or parameter may have
Pattern	Parameter, Property	Specifies a pattern that the value shall match
Permissions	NavigationProperty, Property	Access permission for the property.
Required	NavigationProperty, Property	If true, property is required to be supported by the service. The default is optional. See *Required Properties*

Name	Applies to	Description
RequiredOnCreate	NavigationProperty, Property	If true, property is required on creation. See *Required Properties*
Unit	Property	The unit of measure for the value.

7.4 Property implementation requirements

The client and the implementer should understand that, regardless of the schema declaration, an implementer may choose to not implement a property. If not implemented, a representation of the property will not be present in a reply. This should not be confused with a response that indicates that a property has been implemented, but has no value (i.e. *propertyName = null*).

There are several factors that could affect the implementation choice. Implementation requirements can be defined in many documents. At a minimum, a developer should review, in order: 1. the Redfish specification, 2. this document, and 3. associated profile specifications.

7.5 Schema repository

The primary online source for the Swordfish schema shall be co-located on the DMTF schema site with the Redfish schema: http://redfish.dmtf.org/schemas/swordfish Developers may also download the schema as part of the Swordfish bundle from snia.org (refer to snia.org/swordfish for pointers to the bundle locations).

Implementations should refer either to the versions available on the dmtf.org site or to locally provided instances of the schema.

7.6 Referencing other schemas

Swordfish directly references many Redfish schemas when functionality is already defined and can be leveraged. Other Redfish schema may be added by inference or directly to implementations. Examples are available in the Swordfish mockups.

8 Implementation requirements

8.1 Security

This document generally adheres to the security requirements defined in the Redfish Specification. It extends the Redfish security model in one important way:

- Swordfish implementations shall implement TLS as per the guidance in ISO/IEC 20648 and the TLS Specification for Storage Systems.

8.2 General constraints

8.2.1 Redfish elements

The Swordfish service interface extends the Redfish service interface. As such, a Swordfish service is a Redfish service and all required elements of the Redfish model shall be present in a Swordfish model.

Swordfish functionality shall not conflict with any previously defined Redfish functionality but it may add to or extend it, and it may add additional constraints on Redfish functionality.

Additionally, any functionality desired in a Swordfish implementation that is specified in Redfish shall follow the requirements as specified in the Redfish specification.

8.2.2 Storage Events

8.2.2.1 Overview   A Swordfish implementation should implement an event service. Redfish defines the Event Service framework, client subscription model, event delivery mechanism, as well as standard message registries. Swordfish extends the standard message registries to provide additional message registries that correspond to Swordfish-specific services and properties.

The Redfish event service publishes a list of event types supported, and maintains a list of clients that have subscribed. Each subscription maps clients, subscribed events, and the resources that generate them.

8.2.2.2 Message Registry Selection and Management   Swordfish constrains the existing event model to provide a more consistent handling of event notifications and the related messages, in order to assure that client systems can easily and consistently parse and respond to system-level events.

8.2.2.3 Required Usage

- The Resource Event Message Registry defines the underlying messaging model, and shall be used to map messages to resources for storage implementations.
- The Redfish Base Message Registry shall be used to support HTTP connection/error/protocol issues, and general errors.
- The Swordfish Message Registry shall be used as a supplement for the resource event message registry.
- If the Swordfish service implements Redfish tasks (i.e., long-running operations), the implementation shall use the messages defined in the Task Event Message Registry to report status.

8.2.2.4 Recommended Usage

- Standard Messages should be used, wherever possible.
- OEM messages should be avoided. Suggestions for clarification or expansion of the existing registries are encouraged. (submissions should be sent to the SNIA Feedback Portal)

8.3 Discovering Swordfish resources

Each Swordfish implementation supports the following well-known URLs, as defined in Redfish. Specifically:

- `/Redfish`, which contains one or more version properties for the integrated Swordfish and Redfish implementation, starting with v1.
- `/Redfish/v1`, which addresses a `ServiceRoot` instance, which defines the Redfish default principal starting information for version 1 implementation of an integrated Redfish and Swordfish service. A GET operation to it shall retrieve the value of an instance of a `ServiceRoot EntityType` as defined in the ServiceRoot_v1.xml file.
- `/Redfish/v1/odata`, which addresses a `ServiceContainer` instance, which defines OData conformant principal starting information for the same version 1 implementation of an integrated Redfish and Swordfish service. A GET operation shall retrieve the value of an instance of a `ServiceContainer EntityContainer` as defined in the ServiceRoot_v1.xml file.

> Note: Since the `ServiceContainer` is required to return an `@odata.context` value

> of `/redfish/v1`, all other elements accessed via it will be the same elements found via the `ServiceRoot`.

> Note: A Swordfish service is a Redfish service with extensions to support storage management. No additional service entry-points are necessary.

Both the `ServiceRoot` and `ServiceContainer` contain a resource collection named `Systems` that lists `ComputerSystem` instances. A `ComputerSystem` instance that supports Swordfish defined services will have a value of "StorageServer" in an entry of its `HostingRoles` property.

The `ServiceContainer` additionally has a Service attribute that references the `ServiceRoot` resource.

Regardless of starting point, the property values of the `ServiceRoot` instance enable navigation to all other resources exposed by the Swordfish service.

8.4 ClassOfService requirements

Each `ClassOfService` shall include at least one line of service. The providing server shall assure that the line of service values of a `ClassOfService` collectively represent a supported choice of service.

8.5 StorageSystems requirements

For Hosted Service Configurations, this property of the `ServiceRoot` references a collection of `ComputerSystem` resources that each support Swordfish functionality. Each `ComputerSystem` included in the `StorageSystems` entry in the `ServiceRoot` shall have:

- an entry in its `HostingRoles` property with the value of `StorageServer`
- at least one entry in its `StorageServices.Members` property.

For Integrated Service Configurations, the StorageSystems concept is realized through the `StorageController` resource. Each `StorageController` instantiated as a Swordfish StorageSystem shall have:

- at least one entry in its `StorageController.Links` property `StorageServices` collection identifying related `StorageServices`

8.6 Entity Sets

The Swordfish model does not currently expose any explicitly defined entity sets. OData specifies that an entity set is defined for each NavigationProperty that is defined as a collection and that has the ContainsTarget attribute set to true. In all other cases, Swordfish assumes that an entity set is defined globally within the implementation for each entity type. This is effectively the same as if the entity sets were explicitly defined in the ServiceRoot entity container.

8.7 Addressing entities within a collection

An instance (entity) of an EntityType is uniquely identified within its entity set by its key. The URI for the reference may specify the key using one of two general strategies

1. OData recommends specifying the key value within parenthesis following the path segment that identifies the referencing entity set. (See clause "Canonical URL" in in OData)

2. Redfish common practice is to use an alternative form that adds a path segment having the value of the key following the path segment that identifies the referencing collection. (See clause "Alternate Key-as-Segment Syntax" in OData.)

A Swordfish implementation shall support both strategies.

8.8 Addressing members of a ResourceCollection

Redfish specifies that subclasses of ResourceCollection shall include a Members collection property (See clause "Collection resource response" in DSP0266)

Redfish allows a POST request to a ResourceCollection to be equivalent to the same POST request to the Members property of that ResourceCollection. For a particular ResourceCollection, if a Swordfish implementation supports either form, it shall support both.

It is common practice in Redfish to also eliminate the Members property from any request URI that navigates through a type hierarchy that includes a Member within a ResourceCollection. Care should be taken when defining and using a ResourceCollection subclass to not introduce ambiguities when an explicit reference to a Members property is dropped from a request URI.

8.9 HTTP status codes

8.9.1 Overview

Status codes are generally defined as part of the general HTTP protocol definition. In addition, the Redfish specification calls out general usage for HTTP status codes. This section provides additional usage guidance and constraints for Swordfish implementations.

In some instances, Redfish and Swordfish expand the standard use of HTTP status codes by associating additional system status with specific status codes. In addition, error response data may be included via standardized message registry entries. The specific messaging requirements will be defined in the following sections.

In cases where Swordfish adds additional constraints or expands on the Redfish handling of a given issue, this document will include both a clause reference (relative to the 1.7.0 version of the Redfish specification), and a small wording extract for additional context. For example:

Swordfish refines the requirements in **x.y.z** of the *Redfish Specification*: `Redfish has no constraint on external storage functionality` to require that all references to external storage functionality shall be compliant with the current release of Swordfish.

8.9.2 Create

If a request to create a resource can be completed successfully without additional time, the Redfish service shall return a status code of 201, and the body of the response shall contain the JSON representation of the newly created resource.

If the create resource request has been accepted, but no information about the resource can be returned at this point, the Redfish service shall return a status code of 204. The payload of the response shall be empty, but the Location header shall contain the resource URI. The client will be required to poll the appropriate resource to determine both when and if the operation is complete.

Swordfish refines the requirements in clauses **7.5.1** and **12.2** of the *Redfish Specification*.

If a request to create a resource cannot be completed without additional time, the implementation shall:

- Populate an initial object. It shall contain, at a minimum, a valid URI, required properties (e.g., ID, name), and Status.State;

- Set Status.State of the partially populated resource to "Creating";

- Return the appropriate status code, based on the following guidance:

    – If a Task Service has been implemented, the Redfish service shall return a status code of 202, with the Location header set to the URI of the Task Monitor. Once the provider has returned a Task Monitor to the client, the Client can then query the provided task URI to track the task completion status. Upon task completion, a GET against the task monitor may return a status code of 201, and the body of the message shall contain the created resource, provided the task monitor URI remains valid . Refer to the Redfish Task Manager documentation for the lifecycle of the task monitor URI.

    – If a Task Service has not been implemented, the Redfish service shall return a status code of 201, and the body of the response shall contain the URI of the skeletal resource created as part of accepting the request. The client will be required to poll the URI provided to determine when the operation is complete.

- Update Status.State for the object, once the create operation completes.

8.9.3 Update, Replace, Delete

If a request to modify or delete a resource can be completed without additional time, the Redfish service shall return a status code of 200, and the body of the response shall contain the JSON representation of the modified (or deleted) resource.

If the resource modification or deletion request has been accepted, but no information about the resource can be returned at this point, the Redfish service shall return a status code of 204. The payload of the response shall be empty. The client will be required to poll the appropriate collection to determine both when and if the operation is complete.

If a request to modify a resource cannot be completed without additional time, the implementation shall:

- Set Status.State of the partially populated resource to "Updating" or "Deleting", as appropriate;

- Return the appropriate status code, based on the following guidance:

- If a Task Service has been implemented, the Redfish service shall return a status code of 202, with the Location header set to the URI of the Task Monitor. Once the provider has returned a Task Monitor to the client, the Client can then query the provided task URI to track the task completion status. Upon task completion, a GET against the task monitor may return a status code of 201, and the body of the message shall contain the created resource, provided the task monitor URI remains valid . Refer to the Redfish Task Manager documentation for the lifecycle of the task monitor URI.

- If a Task Service has not been implemented, the Redfish service shall return a status code of 200, and the body of the response shall contain the URI of the skeletal resource created as part of accepting the request. The client will be required to poll the URI provided to determine when the operation is complete.

- For an update or replace request, the implementation shall update Status.State for the resource, once the modify operation completes.

8.9.4 Actions

Swordfish supports the approach to Actions in **5.6.3** of the *Redfish Specification*: `Ac-tions are Redfish operations that do not easily map to RESTful interface semantics. These types of operations may not directly affect properties in the Redfish Resources.`

Swordfish refines the requirements in **7.10** of the *Redfish Specification*: `Services shall support the POST method to send actions to Resources.`

If a Task Service has been implemented, the Redfish service shall return a status code of 202, with the Location header set to the URI of the Task Monitor. Once the provider has returned a Task Monitor to the client, the Client can then query the provided task URI to track the task completion status. Once the task has completed successfully, a GET against the task monitor shall return the created object.

If a Task Service has not been implemented, the Redfish service shall return a status code of 200, and the body of the response shall contain the URI of the skeletal resource created as part of accepting the request. The client will be required to poll the URI provided to determine when the operation is complete. When processing ACTIONS, the handling of HTTP status codes is slightly different than that seen when processing CREATE or MODIFY requests. The HTTP status code is used to reflect the acceptance and formatting of the request. The outcome of any requested processing is reflected in the body of the returned message and its associated Error response structure. For

example, a properly formatted request to execute a system reset may return an HTTP status code of 200 (OK), to reflect that the request has been received, was validly formatted, and has been accepted for processing, while the reset of the system may not complete successfully. The Error response structure would contain further detail of the success of failure of the system reset. The implementation must check both the HTTP status code and the underlying Error response message structure to confirm the successful execution of the ACTION.

9 Swordfish type definitions

9.1 Overview

The following sections define the schema and type definitions that make up a Swordfish implementation. Each data type or entity within the schema includes a description that defines its implementation requirements and their interaction.

9.2 Common properties

The following properties are defined for inclusion in every Redfish schema, and therefore may be encountered in any response payload. They are documented here to avoid repetition in the property tables. Note that several of these properties are payload annotations, but appear here because they are required for all Redfish and Swordfish Resources.

9.2.1 Properties

Property	Type	Attributes	Notes
@odata.context	string(URI)	• read-only*	The value of this property shall be the context URL that describes the resource according to OData-Protocol and shall be of the form defined in the Redfish specification.
@odata.etag	string	• read-only*	The value of this property shall be a string that is defined by the ETag HTTP header definition in RFC7232.
@odata.id	string(URI)	*read-only* *required*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
@odata.type	string	*read-only* *required*	The value of this property shall be an absolute URL that specifies the type of the resource and it shall be of the form defined in the Redfish specification.

Property	Type	Attributes	Notes
Descr iption	string	• read-only*	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.
Id	string	• read-only*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specification-described requirements.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		The manufacturer- or provider-specific extension moniker that divides the Oem object into sections.

9.3 Frequently used properties

In addition, the following properties are frequently defined in Redfish schemas. Their definition and usage is the same throughout the Redfish data model.

9.3.1 Properties

P roperty	Type	Att ributes	Notes
Ac tions {}	object		The Redfish actions available for this Resource.

P roperty	Type	Att ributes	Notes
Links {}	object		The links associated with the Resource, as defined by that Resource's schema definition. All associated reference properties defined for a Resource are nested under the Links property. Find all directly referenced, or subordinate, Resource properties from the root of the Resource.
• *Relate dItem** [{	array		An array of links. Each link points to a Resource or part of a Resource as defined by that Resource's schema. This representation is not intended to be a strong linking methodology like other references. Instead, it shows a relationship between elements or subelements in disparate parts of the service. For example, fans might be in one area of the system and processors in another. The relationship between the two might not be obvious. This property can show that one is related to the other. In this example, it might indicate that a specific fan cools a specific processor.
@oda ta.id	string(URI)	*rea d-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

9.4 Common Swordfish Objects

The following structures are included in multiple Swordfish schema, and therefore may be encountered in any Response payload. They are documented here to avoid

repetition in the Swordfish Specification tables for each schema.

9.4.1 Capacity

9.4.1.1 Description   This composition may be used to represent storage capacity. The sum of the values in Data, Metadata, and Snapshot shall be equal to the total capacity for the data store.

9.4.1.2 Properties

Property	Type	Attributes	Notes
Data {}	o bject		The value shall be capacity information relating to provisioned user data. For property details, see CapacityInfo.
I sThinProvisioned	bo olean	*r ead-o nly(n ull)*	If the value is false, the capacity shall be fully allocated. The default value shall be false.
Metadata {}	o bject		The value shall be capacity information relating to provisioned system (non-user accessible) data. For property details, see CapacityInfo.
Snapshot {}	o bject		The value shall be capacity information relating to provisioned snapshot or backup data. For property details, see CapacityInfo.

9.4.2 CapacityInfo

9.4.2.1 Description   This composition may be used to represent the utilization of storage capacity.

9.4.2.2 Properties

Property	Type	Attributes	Notes
A llocatedBytes	integer(By)	• read-write(null)*	The value shall be the number of bytes currently allocated by the storage system in this data store for this data type.
ConsumedBytes	integer(By)	*read-only(null)*	The value shall be the number of logical bytes currently consumed in this data store for this data type.
Gu aranteedBytes	integer(By)	• read-write(null)*	The value shall be the number of bytes the storage system guarantees can be allocated in this data store for this data type.
Pro visionedBytes	integer(By)	• read-write(null)*	The value shall be the maximum number of bytes that can be allocated in this data store for this data type.

9.4.3 Identifier

9.4.3.1 Description   This type shall contain any additional identifiers for a resource.

9.4.3.2 Properties

Property	Type	Attributes	Notes
DurableName *(v1.1+)*	string	*r ead-only(null)*	This property shall contain the world-wide unique identifier for the resource. The string shall be in the Identifier.Du rableNameFormat property value format.
Dura bleNameFormat *(v1.1+)*	string(enum)	*read-only(null)*	This property shall represent the format of the DurableName property. *For the possible property values, see Du rableNameFormat in Property details.*

9.4.3.3 Property details

9.4.3.3.1 DurableNameFormat: This property shall represent the format of the DurableName property.

string	Description
EUI	This durable name shall contain the hexadecimal representation of the IEEE-defined 64-bit Extended Unique Identifier (EUI), as defined in the IEEE's Guidelines for 64-bit Global Identifier (EUI-64) Specification. The DurableName property shall follow the pattern '^([0-9A-Fa-f]{2}$[:-]){7}([0-9A-Fa-f]{2})', where the most significant octet is first.|$ where the most significant octet is first.
iQN	This durable name shall be in the iSCSI Qualified Name (iQN) format, as defined in RFC3720 and RFC3721.
NAA	This durable name shall contain a hexadecimal representation of the Name Address Authority structure, as defined in the T11 Fibre Channel - Framing and Signaling - 3 (FC-FS-3) specification. The DurableName property shall follow the pattern '^(([0-9A-Fa-f]{2}){8}){1,2}', where the most significant octet is first.||NGUID$ $(v1.10+)*$ $\|This durable name shall be in the Namespace Globally Unique Id... 9A-Fa-f]2)16$', where the most significant octet is first.
NQN *(v1.6+)*	This durable name shall be in the NVMe Qualified Name (NQN) format, as defined in the NVN Express over Fabric Specification.

string	Description
NSID *(v1.6+, deprecated v1.12)*	This durable name shall be in the NVM Namespace Identifier (NSID) format, as defined in the NVN Express Specification. *Deprecated in v1.12 and later. This value has been deprecated due to its non-uniqueness and* NGUID *should be used.*
UUID	This durable name shall contain the hexadecimal representation of the UUID, as defined by RFC4122. The DurableName property shall follow the pattern '([0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12})'.

9.4.4 IOStatistics

9.4.4.1 Description   The properties of this type shall be used to represent the IO statistics of the requested object.

9.4.4.2 Properties

Property	Type	Attributes	Notes
NonIORequests	integer({tot})	• read-write(null)*	The value shall represent the total count from the time of last reset or wrap of non IO requests.

Property	Type	Attributes	Notes
Non IORequestTime	string	*re ad-write(null)*	The value shall be an ISO 8601 conformant duration describing the time that the resource is busy processing non IO requests.
Read HitIORequests	integer({tot})	• read-write(null)*	The value shall represent the total count from the time of last reset or wrap of read IO requests satisfied from memory.
ReadIOKiBytes	integer(KiBy)	• read-write(null)*	The value shall represent the total number of kibibytes read from the time of last reset or wrap.
R eadIORequests	integer({tot})	• read-write(null)*	The value shall represent the total count from the time of last reset or wrap of read IO requests satisfied from either media or memory (i.e. from a storage device or from a cache).

Property	Type	Attributes	Notes
Read IORequestTime	string	*re ad-write(null)*	The value shall be an ISO 8601 conformant duration describing the time that the resource is busy processing read requests.
Write HitIORequests	integer({tot})	• read-write(null)*	The value shall represent the total count from the time of last reset or wrap of write IO requests coalesced into memory.
W riteIOKiBytes	integer(KiBy)	• read-write(null)*	The value shall represent the total number of kibibytes written from the time of last reset or wrap.
Wr iteIORequests	integer({tot})	• read-write(null)*	The value shall represent the total count from the time of last reset or wrap of write IO requests.

Property	Type	Attributes	Notes
Write IORequestTime	string	*re ad-write(null)*	The value shall be an ISO 8601 conformant duration describing the time that the resource is busy processing write requests.

9.4.5 IOWorkload

9.4.5.1 Description   This structure may be used to describe an IO Workload.

9.4.5.2 Properties

Property	Type	Attributes	Notes
Components [{ }]	array (object)	*(null)*	The value shall be an array of IO workload component descriptions. For property details, see IOWor kloadComponent.

Property	Type	Attributes	Notes
Name	string	*re ad-write(null)*	The value shall be a name of the workload. It should be constructed as Or gID:WorkloadID. Examples: ACME:DSS, ACME:DSS-REP, ACME:Exchange, ACME:OLTP, ACME:OLTP-REPA. An organization may define a set of well known workloads.

9.4.6 IOWorkloadComponent

9.4.6.1 Description   This structure may be used to describe a component of an IO workload.

9.4.6.2 Properties

Property	Type	Attributes	Notes
A verageIOBytes	integer(By)	• read-write(null)*	The value shall be the expected average I/O size.

Property	Type	Attributes	Notes
Duration	string(s)	• read-write(null)*	The value of each entry shall be an ISO 8601 duration that shall specify the expected length of time that this component is applied to the workload. This attribute shall be specified if a schedule is specified and otherwise shall not be specified.
IO AccessPattern	string(enum)	• read-write(null)*	The enumeration literal shall be the expected access pattern. *For the possible property values, see IOAccessPattern in Property details.*
PercentOfData	integer(%)	• read-write(null)*	The value shall be the expected percent of the data referenced by the workload that is covered by this component.

Property	Type	Attributes	Notes
PercentOfIOPS	integer(%)	• read-write(null)*	The value shall be the expected percent of the total IOPS for this workload that is covered by this component.
Schedule {}	object		The value shall specifies when this workload component is applied to the overall workload. For property details, see Schedule v1.2.2).

9.4.6.3 Property details

9.4.6.3.1 **IOAccessPattern:**   The enumeration literal shall be the expected access pattern.

string	Description
RandomReadAgain	Use of this enumeration literal shall indicate an access pattern of random reads of cached data.
RandomReadNew	Use of this enumeration literal shall indicate an access pattern of random reads of uncached data.
ReadWrite	Use of this enumeration literal shall indicate a Uniform distribution of reads and writes.

string	Description
SequentialRead	Use of this enumeration literal shall indicate a sequential read pattern of access.
SequentialWrite	Use of this enumeration literal shall indicate a sequential write pattern of access.

9.4.7 Location

9.4.7.1 Description   This type shall describe the location of a resource.

9.4.7.2 Properties

Property	Type	Attributes	Notes
A ltitudeMeters *(v1.6+)*	number(m)	• read-write(null)*	This property shall contain the altitude of the resource in meters.
Contacts *(v1.7+)* [{	array		This property shall contain an array of contact information for an individual or organization responsible for this resource.
ContactName *(v1.7+)*	string	*re ad-write(null)*	This property shall contain the name of a person or organization to contact for information about this resource.

Property	Type	Attributes	Notes
EmailAddress** (v1.7+)	string	re ad-write(null)	This property shall contain the email address for a person or organization to contact for information about this resource.
PhoneNumber (v1.7+)	string	re ad-write(null)	This property shall contain the phone number for a person or organization to contact for information about this resource.
}]			
Info (v1.1+, deprecated v1.5	string	r ead-only(null)	This property shall represent the location of the resource. *Deprecated in v1.5 and later. This property has been deprecated in favor of the PostalAddress, Placement, and PartLocation properties.*

Property	Type	Attributes	Notes
InfoFormat *(v1.1+, deprecated v1.5*	string	*r ead-only(null)*	This property shall represent the Info property format. *Deprecated in v1.5 and later. This property has been deprecated in favor of the PostalAddress, Placement, and PartLocation properties.*
Latitude *(v1.6+)*	number(deg)	• read-write(null)*	This property shall contain the latitude of the resource specified in degrees using a decimal format and not minutes or seconds.
Longitude *(v1.6+)*	number(deg)	• read-write(null)*	This property shall contain the longitude of the resource specified in degrees using a decimal format and not minutes or seconds.

Property	Type	Attributes	Notes
Oem *(v1.1+)* {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.
• *PartLocation** *(v1.5+)* {	object		This property shall contain the part location for a resource within an enclosure. This representation shall indicate the location of a part within a location specified by the Placement property.

Property	Type	Attributes	Notes
Locatio nOrdinalValue (v1.5+)	integer	*r ead-only(null)*	This property shall contain the number that represents the location of the part based on the LocationType. Locat ionOrdinalValue shall be measured based on the Orientation value starting with `0`.
LocationType** (v1.5+)	string(enum)	*read-only(null)*	This property shall contain the type of location of the part, such as slot, bay, socket, or slot. *For the possible property values, see LocationType in Property details.*
Orientation (v1.5+)	string(enum)	*read-only(null)*	This property shall contain the orientation for the ordering used by the Locat ionOrdinalValue property. *For the possible property values, see Orientation in Property details.*

Property	Type	Attributes	Notes
Reference (v1.5+)	string(enum)	*read-only(null)*	This property shall contain the general location within the unit of the part. *For the possible property values, see Reference in Property details.*
ServiceLabel** (v1.5+)	string	*r ead-only(null)*	This property shall contain the label assigned for service at the part location.
}			
Placement (v1.3+) {	object		This property shall contain a place within the addressed location.
A dditionalInfo (v1.7+)	string	*re ad-write(null)*	This property shall contain additional information, such as Tile, Column (Post), Wall, or other designation that describes a location that cannot be conveyed with other properties defined for the Placement object.

Property	Type	Attributes	Notes
Rack *(v1.3+)*	string	*re ad-write(null)*	This property shall contain the name of the rack within a row.
RackOffset *(v1.3+)*	integer	*re ad-write(null)*	The vertical location of the item in the rack. Rack offset units shall be measured from bottom to top, starting with 0.
Ra ckOffsetUnits *(v1.3+)*	string(enum)	• read-write(null)*	This property shall contain a RackUnit enumeration literal that indicates the type of rack units in use. *For the possible property values, see RackOffsetUnits in Property details.*
Row *(v1.3+)*	string	*re ad-write(null)*	This property shall contain the name of the row.
}			
PostalAddress *(v1.3+)* {	object		This property shall contain a postal address of the resource.
A dditionalCode *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the ADDCODE field.

Property	Type	Attributes	Notes
A dditionalInfo (v1.7+)	string	re ad-write(null)	The value shall conform to the requirements of the LOC field as defined in RFC5139. Provides additional information.
Building (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the BLD field. Names the building.
City (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the A3 field. Names a city, township, or shi (JP).
Community (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the PCN field. A postal community name.
Country (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the Country field.

Property	Type	Attributes	Notes
District *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the A2 field. Names a county, parish, gun (JP), or district (IN).
Division *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the A4 field. Names a city division, borough, city district, ward, or chou (JP).
Floor *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the FLR field. Provides a floor designation.

Property	Type	Attributes	Notes
GPSCoords *(v1.3+, deprecated v1.6*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the ADDCODE field. Shall contain the GPS coordinates of the location. If furnished, expressed in the '[-] [nn]n.nnnnnn, [-'][nn]n.nnnnn format. For example, two comma-separated positive or negative numbers with six decimal places of precision. *Deprecated in v1.6 and later. This property has been deprecated in favor of the Longitude and Latitude properties.*
HouseNumber *(v1.3+)*	integer	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the HNO field. The numeric portion of the house number.

Property	Type	Attributes	Notes
Hous eNumberSuffix (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the HNS field. Provides a suffix to a house number, (F, B, or 1/2).
Landmark (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the LMK field. Identifies a landmark or vanity address.
LeadingSt reetDirection (v1.3+)	string	re ad-write(null)	The value shall conform to the requirements of the PRD field as defined in RFC5139. Names a leading street direction, (N, W, or SE).

Property	Type	Attributes	Notes
Location *(v1.3+, deprecated v1.7*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the LOC field. Provides additional information. *Deprecated in v1.7 and later. This property has been deprecated in favor of the AdditionalInfo property.*
Name *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the NAM field. Names the occupant.
Neighborhood** *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the A5 field. Names a neighborhood or block.

Property	Type	Attributes	Notes
PlaceType (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the PLC field. Examples include office and residence.
POBox (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the POBOX field. A post office box (PO box).
PostalCode (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the PC field. A postal code (or zip code).
Road (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the RD field. Designates a primary road or street.

Property	Type	Attributes	Notes
RoadBranch (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the RDBR field. Shall contain a post office box (PO box) road branch.
Roa dPostModifier (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the POM field. For example, Extended.
Ro adPreModifier (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the PRM field. For example, Old or New.
RoadSection (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the RDSEC field. A road section.
RoadSub- Branch (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the RDSUBBR field.

Property	Type	Attributes	Notes
Room *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the ROOM field. A name or number of a room to locate the resource within the unit.
Seat *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the SEAT field. A name or number of a seat, such as the desk, cubicle, or workstation.
Street *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the A6 field. Names a street.
StreetSuffix** *(v1.3+)*	string	*re ad-write(null)*	The value shall conform to the RFC5139-defined requirements of the STS field. Names a street suffix.

Property	Type	Attributes	Notes
Territory (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the A1 field when it names a territory, state, region, province, or prefecture within a country.
Trailin gStreetSuffix (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the POD field. Names a trailing street suffix.
Unit (v1.3+)	string	re ad-write(null)	The value shall conform to the RFC5139-defined requirements of the UNIT field. The name or number of a unit, such as the apartment or suite, to locate the resource.

}

9.4.7.3 Property details

9.4.7.3.1 LocationType: This property shall contain the type of location of the part, such as slot, bay, socket, or slot.

string	Description
Backplane *(v1.12+)*	This value shall indicate the part is a backplane in an enclosure.
Bay	This value shall indicate the part is located in a bay.
Connector	This value shall indicate the part is located in a connector or port.
Slot	This value shall indicate the part is located in a slot.
Socket	This value shall indicate the part is located in a socket.

9.4.7.3.2 Orientation: This property shall contain the orientation for the ordering used by the LocationOrdinalValue property.

string	Description
BackToFront	This value shall indicate the ordering for LocationOrdinalValue is back to front.
BottomToTop	This value shall indicate the ordering for LocationOrdinalValue is bottom to top.
FrontToBack	This value shall indicate the ordering for LocationOrdinalValue is front to back.
LeftToRight	This value shall indicate the ordering for LocationOrdinalValue is left to right.
RightToLeft	This value shall indicate the ordering for LocationOrdinalValue is right to left.
TopToBottom	This value shall indicate the ordering for LocationOrdinalValue is top to bottom.

9.4.7.3.3 RackOffsetUnits: This property shall contain a RackUnit enumeration literal that indicates the type of rack units in use.

string	Description
EIA_310	Rack units shall conform to the EIA-310 standard.
OpenU	Rack units shall be specified in terms of the Open Compute Open Rack Specification.

9.4.7.3.4 Reference: This property shall contain the general location within the unit of the part.

string	Description
Bottom	This value shall indicate the part is in the bottom of the unit.
Front	This value shall indicate the part is in the front of the unit.
Left	This value shall indicate the part is on the left side of of the unit.
Middle	This value shall indicate the part is in the middle of the unit.
Rear	This value shall indicate the part is in the rear of the unit.
Right	This value shall indicate the part is on the right side of the unit.
Top	This value shall indicate the part is in the top of the unit.

9.4.8 Oem

9.4.8.1 Description   This object represents the OEM properties. The resource values shall comply with the Redfish Specification-described requirements.

9.4.8.2 Properties

Property	Type	Attributes	Notes
(pattern) {}	object		Property names follow regular expression pattern "^[A- Za-z0-9_]+$"

9.4.9 ReplicaInfo

9.4.9.1 Description   The value shall define the characteristics of a replica.

9.4.9.2 Properties

Property	Type	Attributes	Notes
Consi stencyEnabled	boolean	*r ead-only(null)*	If true, consistency shall be enabled across the source and its associated target replica(s). The default value for this property is false.
Con sistencyState	string(enum)	*read-only(null)*	The C onsistencyState enumeration literal shall indicate the current state of consistency. *For the possible property values, see C onsistencyState in Property details.*

Property	Type	Attributes	Notes
Cons istencyStatus	string(enum)	*read-only(null)*	The Co nsistencyStatus enumeration literal shall specify the current status of consistency. Consistency may have been disabled or is experiencing an error condition. *For the possible property values, see Co nsistencyStatus in Property details.*
Co nsistencyType	string(enum)	*read-only(null)*	The ConsistencyType enumeration literal shall indicate the consistency type used by the source and its associated target group. *For the possible property values, see ConsistencyType in Property details.*

Property	Type	Attributes	Notes
• *DataProtection LineOfSer- vice** (v1.1+) {	object		The value shall be a pointer to the data protection line of service that describes this replica. See the *DataProtectio nLineOfService* schema for details on this property.
@odata.id	string	*read-write*	Link to a DataProtecti onLineOfService resource. See the Links section and the *DataProtectio nLineOfService* schema for details.
}			
FailedCo pyStopsHostIO	boolean	*r ead-only(null)*	If true, the storage array shall stop receiving data to the source element if copying to a remote element fails. The default value for this property is false.

Property	Type	Attributes	Notes
PercentSynced	integer(%)	*read-only(null)*	Specifies the percent of the work completed to reach s ynchronization. Shall not be instantiated if implementation is not capable of providing this information. If related to a group, then PercentSynced shall be an average of the PercentSynced across all members of the group.
Replica {	object		Deprecated - Use Source Replica. The value shall reference the resource that is the source of this replica.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.

Property	Type	Attributes	Notes
}			
Repli caFaultDomain *(v1.3+)*	string(enum)	*read-only(null)*	The Rep licaFaultDomain enumeration literal shall describe the fault domain (local or remote) of the replica relationship. *For the possible property values, see Rep licaFaultDomain in Property details.*
Re plicaPriority	string(enum)	*read-only(null)*	The enumeration literal shall specify the priority of background copy engine I/O to be managed relative to host I/O operations during a sequential background copy operation. *For the possible property values, see ReplicaPriority in Property details.*

Property	Type	Attributes	Notes
ReplicaP rogressStatus	string(enum)	*read-only(null)*	The Replic aProgressStatus enumeration literal shall specify the status of the session with respect to Replication activity. *For the possible property values, see Replic aProgressStatus in Property details.*
ReplicaR eadOnlyAccess	string(enum)	*read-only(null)*	The enumeration literal shall specify whether the source, the target, or both elements are read only to the host. *For the possible property values, see Replic aReadOnlyAccess in Property details.*
Replic aRecoveryMode	string(enum)	*read-only(null)*	The enumeration literal shall specify whether the copy operation continues after a broken link is restored. *For the possible property values, see Repl icaRecoveryMode in Property details.*

Property	Type	Attributes	Notes
ReplicaRole	string(enum)	*read-only(null)*	The ReplicaRole enumeration literal shall represent the source or target role of this replica as known to the containing resource. *For the possible property values, see ReplicaRole in Property details.*
Rep licaSkewBytes	integer(By)	*read-only(null)*	Applies to Adaptive mode and it describes maximum number of bytes the SyncedElement (target) can be out of sync. If the number of out-of-sync bytes exceeds the skew value, Re plicaUpdateMode shall be switched to synchronous.

Property	Type	Attributes	Notes
• *ReplicaState**	string(enum)	read-only(null)	The ReplicaState enumeration literal shall specify the state of the relationship with respect to Replication activity. *For the possible property values, see ReplicaState in Property details.*
ReplicaType	string(enum)	read-only(null)	The ReplicaType enumeration literal shall describe the intended outcome of the replication. *For the possible property values, see ReplicaType in Property details.*
Repl icaUpdateMode	string(enum)	read-only(null)	The enumeration literal shall specify whether the target elements will be updated synchronously or asynchronously. *For the possible property values, see Re plicaUpdateMode in Property details.*

Property	Type	Attributes	Notes
Requeste dReplicaState	string(enum)	*read-only(null)*	The last requested or desired state for the relationship. The actual state of the relationship shall be represented by ReplicaState. When RequestedState reaches the requested state, this property shall be null. *For the possible property values, see Reques tedReplicaState in Property details.*
SourceReplica *(v1.2+)* {	object		The value shall reference the resource that is the source of this replica.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			

Property	Type	Attributes	Notes
S yncMaintained	boolean	*r ead-only(null)*	If true, Synchronization shall be maintained. The default value for this property is false.
Undisc overedElement	string(enum)	*read-only(null)*	The enumeration literal shall specify whether the source, the target, or both elements involved in a copy operation are undiscovered. An element is considered undiscovered if its object model is not known to the service performing the copy operation. *For the possible property values, see Undi scoveredElement in Property details.*

Property	Type	Attributes	Notes
WhenActivated	string(%)	*read-only(null)*	The value shall be an ISO 8601 conformant time of day that specifies when the point-in-time copy was taken or when the replication relationship is activated, reactivated, resumed or re-established. This property shall be null if the implementation is not capable of providing this information.
Wh enDeactivated	string(%)	*read-only(null)*	The value shall be an ISO 8601 conformant time of day that specifies when the replication relationship is deactivated. Do not instantiate this property if implementation is not capable of providing this information.

Property	Type	Attributes	Notes
Wh enEstablished	string(%)	*read-only(null)*	The value shall be an ISO 8601 conformant time of day that specifies when the replication relationship is established. Do not instantiate this property if implementation is not capable of providing this information.
WhenSuspended	string(%)	*read-only(null)*	The value shall be an ISO 8601 conformant time of day that specifies when the replication relationship is suspended. Do not instantiate this property if implementation is not capable of providing this information.
WhenSynced	string	*r ead-only(null)*	The value shall be an ISO 8601 conformant time of day that specifies when the elements were synchronized.

Property	Type	Attributes	Notes
Whe nSynchronized	string(%)	*read-only(null)*	The value shall be an ISO 8601 conformant time of day that specifies when the replication relationship is synchronized. Do not instantiate this property if implementation is not capable of providing this information.

9.4.9.3 Property details

9.4.9.3.1 ConsistencyState: The ConsistencyState enumeration literal shall indicate the current state of consistency.

string	Description
Consistent	This enumeration literal shall indicate that the source and target shall be consistent.
Inconsistent	This enumeration literal shall indicate that the source and target are not required to be consistent.

9.4.9.3.2 ConsistencyStatus: The ConsistencyStatus enumeration literal shall specify the current status of consistency. Consistency may have been disabled or is experiencing an error condition.

string	Description
Consistent	This enumeration literal shall indicate that the source and target are consistent.
Disabled	This enumeration literal shall indicate that the source and target have consistency disabled.
InError	This enumeration literal shall indicate that the source and target are not consistent.
InProgress	This enumeration literal shall indicate that the source and target are becoming consistent.

9.4.9.3.3 ConsistencyType: The ConsistencyType enumeration literal shall indicate the consistency type used by the source and its associated target group.

string	Description
SequentiallyConsistent	This enumeration literal shall indicate that the source and target shall be sequentially consistent.

9.4.9.3.4 ReplicaFaultDomain: The ReplicaFaultDomain enumeration literal shall describe the fault domain (local or remote) of the replica relationship.

string	Description
Local	This enumeration literal shall indicate that the source and target replicas are contained within a single fault domain.
Remote	This enumeration literal shall indicate that the source and target replicas are in separate fault domains.

9.4.9.3.5 ReplicaPriority: The enumeration literal shall specify the priority of background copy engine I/O to be managed relative to host I/O operations during a sequential background copy operation.

string	Description
High	Copy engine I/O shall have a higher priority than host I/O.
Low	Copy engine I/O shall have a lower priority than host I/O.
Same	Copy engine I/O shall have the same priority as host I/O.
Urgent	Regardless of the host I/O requests, the Copy operation shall be performed as soon as possible.

9.4.9.3.6 ReplicaProgressStatus: The ReplicaProgressStatus enumeration literal shall specify the status of the session with respect to Replication activity.

string	Description
Aborting	This enumeration literal shall indicate that replication has an abort in progress.
Completed	This enumeration literal shall indicate that the request is completed. Data flow is idle.
Detaching	This enumeration literal shall indicate that replication has a detach in progress.
Dormant	This enumeration literal shall indicate that the data flow is inactive, suspended or quiesced.
FailingBack	This enumeration literal shall indicate that replication is undoing the result of failover.

string	Description
FailingOver	This enumeration literal shall indicate that replication is in the process of switching source and target.
Fracturing	This enumeration literal shall indicate that replication has a fracture in progress.
Initializing	This enumeration literal shall indicate that replication is in the process of establishing source/replica relationship and the data flow has not started.
Mixed	This enumeration literal shall indicate that replication status is mixed across element pairs in a replication group. Generally, the individual statuses need to be examined.
Pending	This enumeration literal shall indicate that the flow of data has stopped momentarily due to limited bandwidth or a busy system.
Preparing	This enumeration literal shall indicate that replication has preparation in progress.
RequiresActivate	This enumeration literal shall indicate that the requested operation has completed, however, the synchronization relationship needs to be activated before further copy operations can be issued.
RequiresDetach	This enumeration literal shall indicate that the requested operation has completed, however, the synchronization relationship needs to be detached before further copy operations can be issued.

string	Description
RequiresFracture	This enumeration literal shall indicate that the requested operation has completed, however, the synchronization relationship needs to be fractured before further copy operations can be issued.
RequiresResume	This enumeration literal shall indicate that the requested operation has completed, however, the synchronization relationship needs to be resumed before further copy operations can be issued.
RequiresResync	This enumeration literal shall indicate that the requested operation has completed, however, the synchronization relationship needs to be resynced before further copy operations can be issued.
RequiresSplit	This enumeration literal shall indicate that the requested operation has completed, however, the synchronization relationship needs to be split before further copy operations can be issued.
Restoring	This enumeration literal shall indicate that replication has a restore in progress.
Resyncing	This enumeration literal shall indicate that replication has resynchronization in progress.
Splitting	This enumeration literal shall indicate that replication has a split in progress.
Suspending	This enumeration literal shall indicate that replication has a copy operation in the process of being suspended.
Synchronizing	This enumeration literal shall indicate that replication has synchronization in progress.

string	Description
Terminating	This enumeration literal shall indicate that the replication relationship is in the process of terminating.

9.4.9.3.7 ReplicaReadOnlyAccess: The enumeration literal shall specify whether the source, the target, or both elements are read only to the host.

string	Description
Both	Both the source and the target elements shall be read only to the host.
ReplicaElement	The replica element shall be read-only to the host.
SourceElement	The source element shall be read-only to the host.

9.4.9.3.8 ReplicaRecoveryMode: The enumeration literal shall specify whether the copy operation continues after a broken link is restored.

string	Description
Automatic	The copy operation shall resume automatically.
Manual	The ReplicaState shall be set to Suspended after the link is restored. It is required to issue the Resume operation to continue.

9.4.9.3.9 ReplicaRole: The ReplicaRole enumeration literal shall represent the source or target role of this replica as known to the containing resource.

string	Description
Source	This enumeration literal shall indicate a source element.
Target	This enumeration literal shall indicate target element.

9.4.9.3.10 ReplicaState: The ReplicaState enumeration literal shall specify the state of the relationship with respect to Replication activity.

string	Description
Aborted	This enumeration literal shall indicate that the copy operation is aborted with the Abort operation. The Resync Replica operation can be used to restart the copy operation.
Broken	This enumeration literal shall indicate that the relationship is non-functional due to errors in the source, the target, the path between the two or space constraints.
Failedover	This enumeration literal shall indicate that the reads and writes are sent to the target element. The source element may not be reachable.
Fractured	This enumeration literal shall indicate that the Target is split from the source. The target may not be consistent.
Inactive	This enumeration literal shall indicate that data flow has stopped, writes to source element shall not be sent to target element.
Initialized	This enumeration literal shall indicate that the link to enable replication is established and source/replica elements are associated, but the data flow has not started.

string	Description
Invalid	This enumeration literal shall indicate that the storage server is unable to determine the state of the replication relationship, for example, after the connection is restored; however, either source or target elements have an unknown status.
Mixed	This enumeration literal shall indicate the ReplicaState of GroupSynchronized. The value indicates the StorageSynchronized relationships of the elements in the group have different ReplicaState values.
Partitioned	This enumeration literal shall indicate that the state of replication relationship can not be determined, for example, due to a connection problem.
Prepared	This enumeration literal shall indicate that initialization is completed, however, the data flow has not started.
Restored	This enumeration literal shall indicate that the source element was restored from the target element.
Skewed	This enumeration literal shall indicate that the target has been modified and is no longer synchronized with the source element or the point-in-time view.
Split	This enumeration literal shall indicate that the target element was gracefully (or systematically) split from its source element – consistency shall be guaranteed.

string	Description
Suspended	This enumeration literal shall indicate that the data flow between the source and target elements has stopped. Writes to source element shall be held until the relationship is Resumed.
Synchronized	This enumeration literal shall indicate that for Mirror, Snapshot, or Clone replication, the target represents a copy of the source.
Unsynchronized	This enumeration literal shall indicate that not all the source element data has been copied to the target element.

9.4.9.3.11 ReplicaType: The ReplicaType enumeration literal shall describe the intended outcome of the replication.

string	Description
Clone	This enumeration literal shall indicate that replication shall create a point in time, full copy the source.
Mirror	This enumeration literal shall indicate that replication shall create and maintain a copy of the source.
Snapshot	This enumeration literal shall indicate that replication shall create a point in time, virtual copy of the source.
TokenizedClone	This enumeration literal shall indicate that replication shall create a token based clone.

9.4.9.3.12 ReplicaUpdateMode: The enumeration literal shall specify whether the target elements will be updated synchronously or asynchronously.

string	Description
Active	This enumeration literal shall indicate Active-Active (i.e. bidirectional) synchronous updates.
Adaptive	This enumeration literal shall indicate that an implementation may switch between synchronous and asynchronous modes.
Asynchronous	This enumeration literal shall indicate Asynchronous updates.
Synchronous	This enumeration literal shall indicate Synchronous updates.

9.4.9.3.13 RequestedReplicaState: The last requested or desired state for the relationship. The actual state of the relationship shall be represented by ReplicaState. When RequestedState reaches the requested state, this property shall be null.

string	Description
Aborted	This enumeration literal shall indicate that the copy operation is aborted with the Abort operation. The Resync Replica operation can be used to restart the copy operation.
Broken	This enumeration literal shall indicate that the relationship is non-functional due to errors in the source, the target, the path between the two or space constraints.
Failedover	This enumeration literal shall indicate that the reads and writes are sent to the target element. The source element may not be reachable.

string	Description
Fractured	This enumeration literal shall indicate that the Target is split from the source. The target may not be consistent.
Inactive	This enumeration literal shall indicate that data flow has stopped, writes to source element shall not be sent to target element.
Initialized	This enumeration literal shall indicate that the link to enable replication is established and source/replica elements are associated, but the data flow has not started.
Invalid	This enumeration literal shall indicate that the storage server is unable to determine the state of the replication relationship, for example, after the connection is restored; however, either source or target elements have an unknown status.
Mixed	This enumeration literal shall indicate the ReplicaState of GroupSynchronized. The value indicates the StorageSynchronized relationships of the elements in the group have different ReplicaState values.
Partitioned	This enumeration literal shall indicate that the state of replication relationship can not be determined, for example, due to a connection problem.
Prepared	This enumeration literal shall indicate that initialization is completed, however, the data flow has not started.
Restored	This enumeration literal shall indicate that the source element was restored from the target element.

string	Description
Skewed	This enumeration literal shall indicate that the target has been modified and is no longer synchronized with the source element or the point-in-time view.
Split	This enumeration literal shall indicate that the target element was gracefully (or systematically) split from its source element – consistency shall be guaranteed.
Suspended	This enumeration literal shall indicate that the data flow between the source and target elements has stopped. Writes to source element shall be held until the relationship is Resumed.
Synchronized	This enumeration literal shall indicate that for Mirror, Snapshot, or Clone replication, the target represents a copy of the source.
Unsynchronized	This enumeration literal shall indicate that not all the source element data has been copied to the target element.

9.4.9.3.14 UndiscoveredElement: The enumeration literal shall specify whether the source, the target, or both elements involved in a copy operation are undiscovered. An element is considered undiscovered if its object model is not known to the service performing the copy operation.

string	Description
ReplicaElement	This enumeration literal shall indicate that the replica element is undiscovered.
SourceElement	This enumeration literal shall indicate that the source element is undiscovered.

9.4.10 ReplicaRequest

9.4.10.1 Description   A ReplicaRequest shall contain information about the ReplicaSource and the ReplicaName.

9.4.10.2 Properties

Property	Type	Attributes	Notes
ReplicaName *(v1.1+)*	string	*re ad-write(null)*	The value shall be the names of the replica.
ReplicaSource *(v1.1+)* {	object		The value shall reference a resource to be replicated.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			

9.4.11 Schedule

9.4.11.1 Description   The properties of this type shall schedule a series of occurrences.

9.4.11.2 Properties

Property	Type	Attributes	Notes
Enabl edDaysOfMonth [ ]	array (integer, null)	*read-write*	This property shall contain the days of the month when scheduled occurrences are enabled, for enabled days of week and months of year. If the array contains a single value of `0`, or if the property is not present, all days of the month shall be enabled.
Enab ledDaysOfWeek [ ]	array (string(enum))	• read-write(null)*	Days of the week when scheduled occurrences are enabled. If not present, all days of the week shall be enabled. *For the possible property values, see En abledDaysOfWeek in Property details.*
Ena bledIntervals *(v1.1+)* [ ]	array (string, null)	*read-write*	Each value shall be an ISO 8601 conformant interval specifying when occurrences are enabled.

Property	Type	Attributes	Notes
Enable dMonthsOfYear [ ]	array (string(enum))	• read-write(null)*	This property shall contain the months of the year when scheduled occurrences are enabled, for enabled days of week and days of month. If not present, all months of the year shall be enabled. *For the possible property values, see Enab ledMonthsOfYear in Property details.*
Ini tialStartTime	st ring(date-time)	• read-write(null)*	This property shall contain the date and time when the initial occurrence is scheduled to occur.
Lifetime	string	*re ad-write(null)*	This property shall contain a Redfish Duration that describes the time after provisioning when the schedule expires. Pattern: - ?P(+D)?(T(+H)?(+M)?(+(.+)?S)?)?

Property	Type	Attributes	Notes
M axOccurrences	integer	*re ad-write(null)*	This property shall contain the maximum number of scheduled occurrences.
Name	string	*re ad-write(null)*	The name of the schedule, which is constructed as OrgI D:ScheduleName. Examples include ACME:Daily, ACME:Weekly, and ACM E:FirstTuesday.
Recur renceInterval	string	*re ad-write(null)*	This property shall contain a Redfish Duration that describes the time until the next occurrence. Pattern: - ?P(+D)?(T(+H)?(+M)?(+(.+)?S)?)?

9.4.11.3 Property details

9.4.11.3.1 EnabledDaysOfWeek: Days of the week when scheduled occurrences are enabled. If not present, all days of the week shall be enabled.

string	Description
Every	This value indicates that every day of the week has been selected. When used in array properties, such as for enabling a function on certain days, it shall be the only member in the array.
Friday	
Monday	
Saturday	
Sunday	
Thursday	
Tuesday	
Wednesday	

9.4.11.3.2 EnabledMonthsOfYear: This property shall contain the months of the year when scheduled occurrences are enabled, for enabled days of week and days of month. If not present, all months of the year shall be enabled.

string	Description
April	
August	
December	
Every	This value indicates that every month of the year has been selected. When used in array properties, such as for enabling a function for certain months, it shall be the only member in the array.
February	
January	
July	
June	

string	Description
March	
May	
November	
October	
September	

9.4.12 Status

9.4.12.1 Description   This type shall contain any status or health properties of a resource.

9.4.12.2 Properties

Property	Type	Attributes	Notes
Conditions *(v1.11+)* [{	array		This property shall represent the active conditions requiring attention in this or a related resource that affects the Health or HealthRollup of this resource.
LogEntry {	object		This property shall contain a link to a resource of type LogEntry that represents the log entry created for this condition.

Property	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			
Message	string	*read-only*	This property shall contain a human-readable message describing this condition.
MessageArgs [ ]	array (string)	*read-only*	This property shall contain an array of message arguments that are substituted for the arguments in the message when looked up in the message registry. It has the same semantics as the MessageArgs property in the Redfish MessageRegistry schema.

Property	Type	Attributes	Notes
MessageId	string	*read-only required*	This property shall contain a MessageId, as defined in the 'MessageId format' clause of the Redfish Specification.
Orig inOfCondition {	object		This property shall contain a link to the resource or object that originated the condition. This property shall not be present if the condition was caused by this resource.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			

Property	Type	Attributes	Notes
Severity	string(enum)	*read-only*	This property shall contain the severity of the condition. Services can replace the value defined in the message registry with a value more applicable to the implementation. *For the possible property values, see Severity in Property details.*
Timestamp	st ring(date-time)	*read-only*	This property shall indicate the time the condition occurred.

}]

Property	Type	Attributes	Notes
Health	string(enum)	*read-only(null)*	This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification. *For the possible property values, see Health in Property details.*
• *HealthRollup**	string(enum)	*read-only(null)*	This property shall represent the health state of the resource and its dependent resources. The values shall conform to those defined in the Redfish Specification. *For the possible property values, see HealthRollup in Property details.*

Property	Type	Attributes	Notes
Oem {	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements.
(pattern) {}	object		Property names follow regular expression pattern "^[A- Za-z0-9_]+$"
}			

Property	Type	Attributes	Notes
State	string(enum)	*read-only(null)*	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but it can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable. *For the possible property values, see State in Property details.*

9.4.12.3 Property details

9.4.12.3.1 Health: This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification.

string	Description
Critical	A critical condition requires immediate attention.
OK	Normal.
Warning	A condition requires attention.

9.4.12.3.2 HealthRollup: This property shall represent the health state of the resource and its dependent resources. The values shall conform to those defined in the Redfish Specification.

string	Description
Critical	A critical condition requires immediate attention.
OK	Normal.
Warning	A condition requires attention.

9.4.12.3.3 Severity: This property shall contain the severity of the condition. Services can replace the value defined in the message registry with a value more applicable to the implementation.

string	Description
Critical	A critical condition requires immediate attention.
OK	Normal.
Warning	A condition requires attention.

9.4.12.3.4 State: This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource

has been intentionally made unavailable but it can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.

string	Description
Absent	This function or resource is either not present or detected.
Deferring *(v1.2+)*	The element does not process any commands but queues new requests.
Disabled	This function or resource is disabled.
Enabled	This function or resource is enabled.
InTest	This function or resource is undergoing testing, or is in the process of capturing information for debugging.
Qualified *(v1.9+)*	The element quality is within the acceptable range of operation.
Quiesced *(v1.2+)*	The element is enabled but only processes a restricted set of commands.
StandbyOffline	This function or resource is enabled but awaits an external action to activate it.
StandbySpare	This function or resource is part of a redundancy set and awaits a failover or other external action to activate it.
Starting	This function or resource is starting.
UnavailableOffline *(v1.1+)*	This function or resource is present but cannot be used.
Updating *(v1.2+)*	The element is updating and might be unavailable or degraded.

9.5 Swordfish Schema Types

9.5.1 CapacitySource 1.2.0

9.5.1.1 Description   This composition may be used to represent the source and type of storage capacity. At most one of the ProvidingDrives, ProvidingVolumes, ProvidingMemoryChunks, ProvidingMemory or ProvidingPools properties may have a value. If any of ProvidingDrives, ProvidingVolumes, ProvidingMemory or ProvidingPools reference more than one resource, allocation of capacity across those resources is implementation dependent.

9.5.1.2 URIs   /redfish/v1/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*}
/redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourceId*}
/redfish/v1/Storage/{*StorageId*}/Volumes/{*VolumeId*}/CapacitySources/{*CapacitySourceId*}
/redfish/v1/StorageServices/{*StorageServiceId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*}
/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourceId*}
/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*}/CapacitySources/{*CapacitySourceId*}
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*}
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourceId*}
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/Volumes/{*VolumeId*}/CapacitySources/{*CapacitySourceId*}

9.5.1.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1.2+)* {}	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
Pro videdCapacity {}	object		The value shall be the amount of space that has been provided from the P rovidingDrives, Pr ovidingVolumes, ProvidingMemory or ProvidingPools. For property details, see Capacity.

Property	Type	Attributes	Notes
ProvidedC lassOfService {	object		The value shall reference the provided ClassOfService from the P rovidingDrives, Pr ovidingVolumes, Providi ngMemoryChunks, ProvidingMemory or ProvidingPools. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-only*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			
Pr ovidingDrives {	object		If present, the value shall be a reference to a contributing drive or drives.

Property	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			
Pr ovidingMemory *(v1.1+)* {	object		If present, the value shall be a reference to the contributing memory.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			
Providin gMemoryChunks *(v1.1+)* {	object		If present, the value shall be a reference to the contributing memory chunks.

Property	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			
P rovidingPools {	object		If present, the value shall be a reference to a contributing storage pool or storage pools. Contains a link to a resource.
@odata.id	string	*read-only*	Link to Collection of *StoragePool*. See the StoragePool schema for details.
}			
Pro vidingVolumes {	object		If present, the value shall be a reference to a contributing volume or volumes. Contains a link to a resource.

Property	Type	Attributes	Notes
@odata.id	string	*read-only*	Link to Collection of *Volume*. See the Volume schema for details.
}			

9.5.2 CapacitySourceCollection

9.5.2.1 URIs   /redfish/v1/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources
/redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources
/redfish/v1/Storage/{*StorageId*}/Volumes/{*VolumeId*}/CapacitySources   /red-
fish/v1/StorageServices/{*StorageServiceId*}/FileSystems/{*FileSystemId*}/CapacitySources
/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/CapacitySources
/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*}/CapacitySources
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/Volumes/{*VolumeId*}/CapacitySources

9.5.2.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Members [{	array		The value of each member entry shall reference a CapacitySource resource.
@odata.id	string	*read-only*	Link to a CapacitySource resource. See the Links section and the *CapacitySource* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.

Property	Type	Attributes	Notes
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.3 ClassOfService 1.2.0

9.5.3.1 Description   This resource shall define a service option composed of one or more line of service entities. ITIL defines a service option as a choice of utility or warranty for a service.

9.5.3.2 URIs   /redfish/v1/StorageServices/*{StorageServiceId}*/ClassesOfService/*{ClassOfServiceId}*
/redfish/v1/StorageServices/*{StorageServiceId}*/StoragePools/*{StoragePoolId}*/ClassesOfService/*{ClassOfServic...*

9.5.3.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.
ClassOfS erviceVersion	string	*re ad-write(null)*	The version describing the creation or last modification of this service option specification. The string representing the version shall be in the form: M + ':' + N + ':' + U Where: M - The major version (in numeric form). N - The minor version (in numeric form). U - The update (e.g. errata or patch in numeric form).

Property	Type	Attributes	Notes
DataProtectionLinesOfService *(v1.1.1+)* [{	array		The value shall be a set of data protection service options. Within a class of service, one data protection service option shall be present for each replication session.
@odata.id	string	*read-write*	Link to a DataProtectionLineOfService resource. See the Links section and the *DataProtectionLineOfService* schema for details.
}]			
DataSecurityLinesOfService *(v1.1.1+)* [{	array		The value shall be a set of data security service options.
@odata.id	string	*read-write*	Link to a DataSecurityLineOfService resource. See the Links section and the *DataSecurityLineOfService* schema for details.
}]			

Property	Type	Attributes	Notes
DataStorageL inesOfService *(v1.1.1+)* [{	array		The value shall be a set of data protection service options.
@odata.id	string	*read-write*	Link to a DataStora geLineOfService resource. See the Links section and the *DataStorag eLineOfService* schema for details.
}]			
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Identifier {}	object		The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).
IOConnectivityLinesOfService *(v1.1.1+)* [{	array		The value shall be a set of IO connectivity service options. Within a class of service, at most one IO connectivity service option may be present for a value of AccessProtocol.
@odata.id	string	*read-write*	Link to a IOConnectivityLineOfService resource. See the Links section and the *IOConnectivityLineOfService* schema for details.
}]			
• *IOPerformanceLinesOfService** *(v1.1.1+)* [{	array		The value shall be a set of IO performance service options.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a IOPerforman ceLineOfService resource. See the Links section and the *IOPerformanc eLineOfService* schema for details.
}]			
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.4 ClassOfServiceCollection

9.5.4.1 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService /red-
fish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/ClassesOfService

9.5.4.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Members [{	array		The value of each member entry shall reference a ClassOfService or LineOfService resource.
@odata.id	string	*read-only*	Link to a LineOfService resource. See the Links section and the *LineOfService* schema for details.
}]			
Members@odata.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.

Property	Type	Attributes	Notes
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.5 ConsistencyGroup 1.0.1

9.5.5.1 Description   A collection of volumes grouped together to ensure write order consistency across all those volumes. A management operation on a consistency group, such as configuring replication properties, applies to all the volumes within the consistency group.

9.5.5.2 URIs   /redfish/v1/Storage/*{StorageId}*/ConsistencyGroups/*{ConsistencyGroupId}*
/redfish/v1/StorageServices/*{StorageServiceId}*/ConsistencyGroups/*{ConsistencyGroupId}*
/redfish/v1/StorageServices/*{StorageServiceId}*/Volumes/*{VolumeId}*/ConsistencyGroups/*{ConsistencyGroupId}*
/redfish/v1/Systems/*{ComputerSystemId}*/Storage/*{StorageId}*/ConsistencyGroups/*{ConsistencyGroupId}*

9.5.5.3 Properties

Property	Type	Attributes	Notes
Actions {	object		The Actions property shall contain the available actions for this resource.
#ConsistencyGroup.AssignReplicaTarget {}	object		This action shall be used to establish a replication relationship by assigning an existing consistency group to serve as a target replica for an existing source consistency group. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#ConsistencyGroup.CreateReplicaTarget {}	object		This action shall be used to create a new consistency group resource to provide expanded data protection through a replica relationship with the specified source consistency group. *For more information, see the Actions section below.*
#Consistency-Group.RemoveReplicaRelationship {}	object		This action shall be used to disable data synchronization between a source and target consistency group, remove the replication relationship, and optionally delete the target consistency group. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#ConsistencyGroup.ResumeReplication {}	object		This action shall be used to resume the active data synchronization between a source and target consistency group, without otherwise altering the replication relationship. *For more information, see the Actions section below.*
#ConsistencyGroup.ReverseReplicationRelationship {}	object		This action shall be used to reverse the replication relationship between a source and target consistency group. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#Consistency Group.SplitReplication {}	object		This action shall be used to split the replication relationship and suspend data synchronization between a source and target consistency group. *For more information, see the Actions section below.*
#ConsistencyGroup.SuspendReplication {}	object		This action shall be used to suspend active data synchronization between a source and target consistency group, without otherwise altering the replication relationship. *For more information, see the Actions section below.*

}

Property	Type	Attributes	Notes
Cons istencyMethod	string(enum)	• read-write(null)*	The property shall set the consistency method used by this group. *For the possible property values, see Co nsistencyMethod in Property details.*
Co nsistencyType	string(enum)	• read-write(null)*	This property shall set the consistency type used by this group. *For the possible property values, see ConsistencyType in Property details.*
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
• *IsConsistent**	boolean	*r ead-only(null)*	The value of this property shall be set to true when the consistency group is in a consistent state.
Links {	object		This property shall contain links to other resources that are related to this resource.
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
}			

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
ReplicaInfo {}	object		This property shall describe the replication relationship between this storage group and a corresponding source storage group. For property details, see ReplicaInfo v1.3.0).
R eplicaTargets [{	array		The value shall reference the target replicas that are sourced by this replica.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			
Status {}	object		The property shall contain the status of the Co nsistencyGroup. For property details, see Status.

Property	Type	Attributes	Notes
Volumes [{	array		An array of references to volumes managed by this storage group.
@odata.id	string	*read-write*	Link to a Volume resource. See the Links section and the *Volume* schema for details.
}]			

9.5.5.4 Actions

9.5.5.4.1 AssignReplicaTarget   Description

This action shall be used to establish a replication relationship by assigning an existing consistency group to serve as a target replica for an existing source consistency group.

Action URI: {Base URI of target resource}/Actions/ConsistencyGroup.AssignReplicaTarget

Action parameters

Parameter Name	Type	Attributes	Notes
ReplicaType	string(enum)	*required*	This parameter shall contain the type of replica relationship to be created. *For the possible property values, see ReplicaType in Property details.*

Parameter Name	Type	Attributes	Notes
Repl icaUpdateMode	string(enum)	*required*	This parameter shall specify the replica update mode. *For the possible property values, see Re plicaUpdateMode in Property details.*
TargetCon sistencyGroup	string	*required*	This parameter shall contain the Uri to the existing consistency group.

9.5.5.4.2 CreateReplicaTarget   Description

This action shall be used to create a new consistency group resource to provide expanded data protection through a replica relationship with the specified source consistency group.

Action URI: {Base URI of target resource}/Actions/ConsistencyGroup.CreateReplicaTarget

Action parameters

Parameter Name	Type	Attributes	Notes
Consist encyGroupName	string	*required*	This parameter shall contain the Name for the target consistency group.

Parameter Name	Type	Attributes	Notes
ReplicaType	string(enum)	*required*	This parameter shall contain the type of replica relationship to be created. *For the possible property values, see ReplicaType in Property details.*
Repl icaUpdateMode	string(enum)	*required*	This parameter shall specify the replica update mode. *For the possible property values, see Re plicaUpdateMode in Property details.*
Targ etStoragePool	string	*required*	This parameter shall contain the Uri to the existing StoragePool in which to create the target consistency group.

9.5.5.4.3 RemoveReplicaRelationship   Description

This action shall be used to disable data synchronization between a source and target consistency group, remove the replication relationship, and optionally delete the target consistency group.

Action URI: {Base URI of target resource}/Actions/ConsistencyGroup.RemoveReplicaRelationship

Action parameters

Parameter Name	Type	Attributes	Notes
DeleteTarget-Con sistencyGroup	boolean	*optional*	This parameter shall indicate whether or not to delete the target consistency group as part of the operation. If not specified, the system should use its default behavior.
TargetCon sistencyGroup	string	*required*	This parameter shall contain the Uri to the existing target consistency group.

9.5.5.4.4 ResumeReplication   Description

This action shall be used to resume the active data synchronization between a source and target consistency group, without otherwise altering the replication relationship.

Action URI: {Base URI of target resource}/Actions/ConsistencyGroup.ResumeReplication

Action parameters

Parameter Name	Type	Attributes	Notes
TargetCon sistencyGroup	string	*required*	This parameter shall contain the Uri to the existing target consistency group.

9.5.5.4.5 ReverseReplicationRelationship   Description

This action shall be used to reverse the replication relationship between a source and target consistency group.

Action URI: {Base URI of target resource}/Actions/ConsistencyGroup.ReverseReplicationRelationship

Action parameters

Parameter Name	Type	Attributes	Notes
TargetCon sistencyGroup	string	*required*	This parameter shall contain the Uri to the existing target consistency group.

9.5.5.4.6 SplitReplication   Description

This action shall be used to split the replication relationship and suspend data synchronization between a source and target consistency group.

Action URI: {Base URI of target resource}/Actions/ConsistencyGroup.SplitReplication

Action parameters

Parameter Name	Type	Attributes	Notes
TargetCon sistencyGroup	string	*required*	This parameter shall contain the Uri to the existing target consistency group.

9.5.5.4.7 SuspendReplication   Description

This action shall be used to suspend active data synchronization between a source and target consistency group, without otherwise altering the replication relationship.

Action URI: {Base URI of target resource}/Actions/ConsistencyGroup.SuspendReplication

Action parameters

Parameter Name	Type	Attributes	Notes
TargetCon sistencyGroup	string	*required*	This parameter shall contain the Uri to the existing target consistency group.

9.5.5.5 Property details

9.5.5.5.1 ConsistencyMethod: The property shall set the consistency method used by this group.

string	Description
HotStandby	Supports consistency method commonly orchestrated using application-specific code.
Other	Supports consistency method orchestrated using vendor-specific code.
VASA	Supports VMware consistency requirements, such as for VASA and VVOLs.
VDI	Supports Microsoft virtual backup device interface (VDI).
VSS	Supports Microsoft VSS.

9.5.5.5.2 ConsistencyType: This property shall set the consistency type used by this group.

string	Description
ApplicationConsistent	Orchestration exists to either flush or halt pending IO to ensure operations occur in a transactionally consistent manner.

string	Description
CrashConsistent	Requested operations are either triggered or instituted without regard to pending IO.

9.5.5.5.3 ReplicaType: This parameter shall contain the type of replica relationship to be created.

string	Description
Clone	This enumeration literal shall indicate that replication shall create a point in time, full copy the source.
Mirror	This enumeration literal shall indicate that replication shall create and maintain a copy of the source.
Snapshot	This enumeration literal shall indicate that replication shall create a point in time, virtual copy of the source.
TokenizedClone	This enumeration literal shall indicate that replication shall create a token based clone.

9.5.5.5.4 ReplicaUpdateMode: This parameter shall specify the replica update mode.

string	Description
Active	This enumeration literal shall indicate Active-Active (i.e. bidirectional) synchronous updates.
Adaptive	This enumeration literal shall indicate that an implementation may switch between synchronous and asynchronous modes.

string	Description
Asynchronous	This enumeration literal shall indicate Asynchronous updates.
Synchronous	This enumeration literal shall indicate Synchronous updates.

9.5.6 ConsistencyGroupCollection

9.5.6.1 URIs   /redfish/v1/Storage/{*StorageId*}/ConsistencyGroups /redfish/v1/StorageServices/{*StorageServi*
/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*}/ConsistencyGroups
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/ConsistencyGroups

9.5.6.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		The value of each member entry shall reference a C onsistencyGroup resource.

Property	Type	Attributes	Notes
@odata.id	string	*read-only*	Link to a C onsistencyGroup resource. See the Links section and the *Co nsistencyGroup* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.

9.5.7 DataProtectionLineOfService 1.3.0

9.5.7.1 Description   This service option describes a replica that protects data from loss. The requirements must be met collectively by the communication path and the replica.

9.5.7.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/DataProte
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/DataProtectionLinesOfService/{*DataProtection*

9.5.7.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.2+)* {	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
#DataProtec-tion LineOfService.C reateReplicas {} }	object		This action shall create an on-demand replica that conforms to the bound DataProtectio nLineOfService. *For more information, see the Actions section below.*
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
IsIsolated	boolean	*re ad-write(null)*	True shall indicate that the replica is in a separate fault domain from its source. The default value of this property is false.
MinLifetime	string	*re ad-write(null)*	The value shall be an ISO 8601 duration that specifies the minimum required lifetime of the replica. Note: The maximum number of replicas can be determined using this value together with the r eplicaSchedule.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
• *RecoveryGeogra phicObjec- tive**	string(enum)	• read- write(null)*	The value specifies the geographic scope of the failure domain. *For the possible property values, see RecoveryGeog raphicObjective in Property details.*
RecoveryPoint ObjectiveTime	string	*re ad-write(null)*	The value shall be an ISO 8601 duration that specifies the maximum time over which source data may be lost on failure. In the case that IsIsolated = false, failure of the domain is not a consideration.

Property	Type	Attributes	Notes
Recovery TimeObjective	string(enum)	• read-write(null)*	The value shall be an enumeration that indicates the maximum time required to access an alternate replica. In the case that IsIsolated = false, failure of the domain is not a consideration. *For the possible property values, see Recove ryTimeObjective in Property details.*
ReplicaA ccessLocation {}	object		This value shall be used if the data access location of the replica is required to be at a specific location. Note 1: The location value may be granular. Note 2: A value may be required for some regulatory compliance. For property details, see Location v1.3.0).

Property	Type	Attributes	Notes
ReplicaC lassOfService {	object		The value shall reference the class of service that defines the required service levels of the replica. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-write*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			
ReplicaType	string(enum)	• read-write(null)*	The type of replica shall conform to this value. *For the possible property values, see ReplicaType in Property details.*
Schedule {}	object		If a replica is made periodically, the value shall define the schedule. For property details, see Schedule v1.2.2).

9.5.7.4 Actions

9.5.7.4.1 CreateReplicas   Description

This action shall create an on-demand replica that conforms to the bound DataProtectionLineOfService.

Action URI: {Base URI of target resource}/Actions/DataProtectionLineOfService.CreateReplicas

Action parameters

Parameter Name	Type	Attributes	Notes
Replica LineOfService {	object	*required*	The value shall reference the data protection line of service this operation is bound to.
@odata.id	string	*read-only*	Link to another DataProtecti onLineOfService resource.
}			
Re plicaRequests [{	array	*optional*	Each value shall reference a source resource and provide a name for the replica.
ReplicaName *(v1.1+)*	string	*re ad-write(null)*	The value shall be the names of the replica.
ReplicaS-ource *(v1.1+)* {	object		The value shall reference a resource to be replicated.

Parameter Name	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.

```
    }
  }]
```

9.5.7.5 Property details

9.5.7.5.1 RecoveryGeographicObjective: The value specifies the geographic scope of the failure domain.

string	Description
Datacenter	A facility that provides communication, power, or cooling infrastructure to a co-located set of servers, networking and storage.
Rack	A container within a datacenter that provides communication, power, or cooling to a set of components.
RackGroup	A set of racks that may share common communication, power, or cooling.
Region	A set of resources that are required to be either geographically or politically isolated from resources not in the resources.

string	Description
Row	A set of adjacent racks or rackgroups that may share common communication, power, or cooling.
Server	Components of a CPU/memory complex that share the same infrastructure.

9.5.7.5.2 RecoveryTimeObjective: The value shall be an enumeration that indicates the maximum time required to access an alternate replica. In the case that IsIsolated = false, failure of the domain is not a consideration.

string	Description
Nearline	Access to a replica shall be consistent with switching access to a different path through a different front-end interconnection infrastructure. Some inconsistency may occur. A restore step may be required before recovery can commence.
Offline	Access to a replica may take a significant amount of time. No direct connection to the replica is assumed. Some inconsistency loss may occur. A restore step is likely to be required.
OnlineActive	Access to synchronous replicas shall be instantaneous.
OnlinePassive	Access to a synchronous replica shall be consistent with switching access to a different path the same front-end interconnect. A restore step shall not be required.

9.5.7.5.3 ReplicaType: The type of replica shall conform to this value.

string	Description
Clone	This enumeration literal shall indicate that replication shall create a point in time, full copy the source.
Mirror	This enumeration literal shall indicate that replication shall create and maintain a copy of the source.
Snapshot	This enumeration literal shall indicate that replication shall create a point in time, virtual copy of the source.
TokenizedClone	This enumeration literal shall indicate that replication shall create a token based clone.

9.5.8 DataProtectionLoSCapabilities 1.2.0

9.5.8.1 Description   The capabilities to protect data from loss by the use of a replica. The requirements shall be met collectively by the communication path and the replica. There should be one instance associated to a class of service for each replica. Each replica independently should have a class of service that describes its characteristics.

9.5.8.2 URIs   /redfish/v1/StorageServices/*{StorageServiceId}*/DataProtectionLoSCapabilities

9.5.8.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Identifier {}	object		The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).
Links {	object		The value of this property shall contains links to other resources that are not contained in this resource.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.
SupportedR eplicaOptions [{	array		The collection shall contain known and supported replica Classes of Service.
@odata.id	string	*read-write*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.

   }]

}

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specification-described requirements. For property details, see Oem.
SupportedLinesOfService [{	array		The collection shall contain known and supported DataProtection LinesOfService.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a DataProtecti onLineOfService resource. See the Links section and the *DataProtectio nLineOfService* schema for details.
}]			
Supporte dMinLifetimes [ ]	array (string, null)	*read-write*	The value of each entry shall be an ISO 8601 duration that specifies the minimum lifetime required for the replica.
Supported RecoveryGeograp hicObjectives [ ]	array (string(enum))	• read-write(null)*	The value of each entry shall specify a supported failure domain. *For the possible property values, see Support edRecoveryGeogr aphicObjectives in Property details.*

Property	Type	Attributes	Notes
Supporte dRecoveryPointO bjectiveTimes [ ]	array (string, null)	*read-write*	The value of each entry shall specify a supported ISO 8601 time interval defining the maximum source information that may be lost on failure. In the case that IsIsolated = false, failure of the domain is not a consideration.
Sup portedRecoveryT imeObjectives [ ]	array (string(enum))	• read-write(null)*	The value of each entry shall specify an enumerated value that indicates a supported expectation for the time required to access an alternate replica. In the case that IsIsolated = false, failure of the domain is not a consideration. *For the possible property values, see S upportedRecover yTimeObjectives in Property details.*

Property	Type	Attributes	Notes
Supporte dReplicaTypes [ ]	array (string(enum))	• read-write(null)*	The value of each entry shall specify a supported replica type. *For the possible property values, see Suppor tedReplicaTypes in Property details.*
Sup portsIsolated	boolean	*re ad-write(null)*	A value of true shall indicate that allocating a replica in a separate fault domain is supported. The default value for this property is false.

9.5.8.4 Property details

9.5.8.4.1 SupportedRecoveryGeographicObjectives: The value of each entry shall specify a supported failure domain.

string	Description
Datacenter	A facility that provides communication, power, or cooling infrastructure to a co-located set of servers, networking and storage.
Rack	A container within a datacenter that provides communication, power, or cooling to a set of components.

string	Description
RackGroup	A set of racks that may share common communication, power, or cooling.
Region	A set of resources that are required to be either geographically or politically isolated from resources not in the resources.
Row	A set of adjacent racks or rackgroups that may share common communication, power, or cooling.
Server	Components of a CPU/memory complex that share the same infrastructure.

9.5.8.4.2 SupportedRecoveryTimeObjectives: The value of each entry shall specify an enumerated value that indicates a supported expectation for the time required to access an alternate replica. In the case that IsIsolated = false, failure of the domain is not a consideration.

string	Description
Nearline	Access to a replica shall be consistent with switching access to a different path through a different front-end interconnection infrastructure. Some inconsistency may occur. A restore step may be required before recovery can commence.
Offline	Access to a replica may take a significant amount of time. No direct connection to the replica is assumed. Some inconsistency loss may occur. A restore step is likely to be required.
OnlineActive	Access to synchronous replicas shall be instantaneous.

string	Description
OnlinePassive	Access to a synchronous replica shall be consistent with switching access to a different path the same front-end interconnect. A restore step shall not be required.

9.5.8.4.3 SupportedReplicaTypes: The value of each entry shall specify a supported replica type.

string	Description
Clone	This enumeration literal shall indicate that replication shall create a point in time, full copy the source.
Mirror	This enumeration literal shall indicate that replication shall create and maintain a copy of the source.
Snapshot	This enumeration literal shall indicate that replication shall create a point in time, virtual copy of the source.
TokenizedClone	This enumeration literal shall indicate that replication shall create a token based clone.

9.5.9 DataSecurityLineOfService 1.1.1

9.5.9.1 Description   This structure shall be used to describe data security service level requirements.

9.5.9.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/DataSecu
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/DataSecurityLinesOfService/{*DataSecurityLine*

9.5.9.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.
AntivirusE ngineProvider	string	*re ad-write(null)*	The value shall specify an AntiVirus provider.
Antiviru sScanPolicies [ ]	array (string(enum))	• read-write(null)*	The enumeration literal shall specify the policy for triggering an AntiVirus scan. *For the possible property values, see Antivi rusScanPolicies in Property details.*
ChannelEncry ptionStrength	string(enum)	• read-write(null)*	The enumeration literal shall specify a key size in a symmetric encryption algorithm for transport channel encryption. *For the possible property values, see ChannelEnc ryptionStrength in Property details.*

Property	Type	Attributes	Notes
DataSanitizationPolicy	string(enum)	• read-write(null)*	The enumeration literal shall specify the data sanitization policy. *For the possible property values, see DataSanitizationPolicy in Property details.*
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.
HostAuthenticationType	string(enum)	• read-write(null)*	The enumeration literal shall specify the authentication type for hosts (servers) or initiator endpoints. *For the possible property values, see HostAuthenticationType in Property details.*

Property	Type	Attributes	Notes
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
MediaEncry ptionStrength	string(enum)	• read-write(null)*	The enumeration literal shall specify a key size in a symmetric encryption algorithm for media encryption. *For the possible property values, see MediaEnc ryptionStrength in Property details.*

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
SecureCh annelProtocol	string(enum)	• read-write(null)*	The enumeration literal shall specify the protocol that provide encrypted communication. *For the possible property values, see Secure ChannelProtocol in Property details.*
UserAuthe nticationType	string(enum)	• read-write(null)*	The enumeration literal shall specify the authentication type for users (or programs). *For the possible property values, see UserAut henticationType in Property details.*

9.5.9.4 Property details

9.5.9.4.1 AntivirusScanPolicies:　The enumeration literal shall specify the policy for triggering an AntiVirus scan.

string	Description
None	This enumeration literal specifies No trigger.
OnFirstRead	This enumeration literal specifies to trigger on first read.
OnPatternUpdate	This enumeration literal specifies to trigger on antivirus pattern file update.

string	Description
OnRename	This enumeration literal specifies to trigger on object rename.
OnUpdate	This enumeration literal specifies to trigger on object update.

9.5.9.4.2 ChannelEncryptionStrength: The enumeration literal shall specify a key size in a symmetric encryption algorithm for transport channel encryption.

string	Description
Bits_0	This enumeration literal specifies that there is no key.
Bits_112	This enumeration literal specifies a 3DES 112 bit key.
Bits_128	This enumeration literal specifies an AES 128 bit key.
Bits_192	This enumeration literal specifies an AES 192 bit key.
Bits_256	This enumeration literal specifies an AES 256 bit key.

9.5.9.4.3 DataSanitizationPolicy: The enumeration literal shall specify the data sanitization policy.

string	Description
Clear	This enumeration literal specifies to sanitize data in all user-addressable storage locations for protection against simple non-invasive data recovery techniques.

string	Description
CryptographicErase	This enumeration literal specifies to leverages the encryption of target data by enabling sanitization of the target data's encryption key. This leaves only the ciphertext remaining on the media, effectively sanitizing the data by preventing read-access. For more information, see NIST800-88 and ISO/IEC 27040.
None	This enumeration literal specifies no sanitization.

9.5.9.4.4 HostAuthenticationType: The enumeration literal shall specify the authentication type for hosts (servers) or initiator endpoints.

string	Description
None	This enumeration literal specifies No authentication.
Password	This enumeration literal specifies Password/shared-secret: Absent an distributed authentication infrastructure, this is what is typically done.
PKI	This enumeration literal specifies a Public Key Infrastructure. Customers with the highest assurance requirements roll PKI out to hosts and users (it is more common for hosts than users. User PKI-based authentication has significant operational complications and administrative overheads, e.g., smart cards may be involved.

string	Description
Ticket	This enumeration literal specifies Ticket-based (e.g., Kerberos): This is the most common class of authentication infrastructure used in enterprises. Kerberos is the best known example, and Windows usage of that via Active Directory is so widely deployed as to be a de facto standard. In other areas (e.g., academia) there are comparable ticket-based systems.

9.5.9.4.5 MediaEncryptionStrength: The enumeration literal shall specify a key size in a symmetric encryption algorithm for media encryption.

string	Description
Bits_0	This enumeration literal specifies that there is no key.
Bits_112	This enumeration literal specifies a 3DES 112 bit key.
Bits_128	This enumeration literal specifies an AES 128 bit key.
Bits_192	This enumeration literal specifies an AES 192 bit key.
Bits_256	This enumeration literal specifies an AES 256 bit key.

9.5.9.4.6 SecureChannelProtocol: The enumeration literal shall specify the protocol that provide encrypted communication.

string	Description
IPsec	This enumeration literal specifies Internet Protocol Security (IPsec), as defined by IETF RFC 2401.
None	This enumeration literal specifies no encryption.

string	Description
RPCSEC_GSS	This enumeration literal specifies RPC access to the Generic Security Services Application Programming Interface (GSS-API), as defined by IETF RPC 2203.
TLS	This enumeration literal specifies Transport Layer Security (TLS), as defined by IETF RFC 5246.

9.5.9.4.7 UserAuthenticationType: The enumeration literal shall specify the authentication type for users (or programs).

string	Description
None	This enumeration literal specifies No authentication.
Password	This enumeration literal specifies Password/shared-secret: Absent an distributed authentication infrastructure, this is what is typically done.
PKI	This enumeration literal specifies a Public Key Infrastructure. Customers with the highest assurance requirements roll PKI out to hosts and users (it is more common for hosts than users. User PKI-based authentication has significant operational complications and administrative overheads, e.g., smart cards may be involved.

string	Description
Ticket	This enumeration literal specifies Ticket-based (e.g., Kerberos): This is the most common class of authentication infrastructure used in enterprises. Kerberos is the best known example, and Windows usage of that via Active Directory is so widely deployed as to be a de facto standard. In other areas (e.g., academia) there are comparable ticket-based systems.

9.5.10 DataSecurityLoSCapabilities 1.2.0

9.5.10.1 Description   This resource may be used to describe data security capabilities.

9.5.10.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/DataSecurityLoSCapabilities

9.5.10.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Identifier {}	object		The value identifies this resource. The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
Suppo rtedAntivirusEn gineProviders [ ]	array (string, null)	*read-write*	The entry values shall specify supported AntiVirus providers.

Property	Type	Attributes	Notes
Su pportedAntiviru sScanPolicies [ ]	array (string(enum))	• read-write(null)*	The enumeration literal shall specify supported policies that trigger an AntiVirus scan. *For the possible property values, see SupportedAntivi rusScanPolicies in Property details.*
Support edChannelEncryp tionStrengths [ ]	array (string(enum))	• read-write(null)*	The enumeration literal shall specify supported key sizes in a symmetric encryption algorithm (AES) for transport channel encryption. *For the possible property values, see Suppo rtedChannelEncr yptionStrengths in Property details.*
Suppo rtedDataSanitiz ationPolicies [ ]	array (string(enum))	• read-write(null)*	The enumeration literal shall specify supported data sanitization policies. *For the possible property values, see Sup portedDataSanit izationPolicies in Property details.*

Property	Type	Attributes	Notes
Supp ortedHostAuthen ticationTypes [ ]	array (string(enum))	• read-write(null)*	The enumeration literal shall specify supported authentication types for hosts (servers) or initiator endpoints. *For the possible property values, see Su pportedHostAuth enticationTypes in Property details.*
SupportedL inesOfService [{	array		The collection shall contain supported DataSecurity service options.
@odata.id	string	*read-write*	Link to a DataSecuri tyLineOfService resource. See the Links section and the *DataSecurit yLineOfService* schema for details.

}]

Property	Type	Attributes	Notes
Suppo rtedMediaEncryp tionStrengths [ ]	array (string(enum))	• read-write(null)*	The enumeration literal shall specify supported key sizes in a symmetric encryption algorithm (AES) for media encryption. *For the possible property values, see Sup portedMediaEncr yptionStrengths in Property details.*
Sup portedSecureCha nnelProtocols [ ]	array (string(enum))	• read-write(null)*	The enumeration literal shall specify supported protocols that provide encrypted communication. *For the possible property values, see S upportedSecureC hannelProtocols in Property details.*

Property	Type	Attributes	Notes
Supp ortedUserAuthen ticationTypes [ ]	array (string(enum))	• read-write(null)*	The enumeration literal shall specify supported authentication types for users (or programs). *For the possible property values, see Su pportedUserAuth enticationTypes in Property details.*

9.5.10.4 Property details

9.5.10.4.1 SupportedAntivirusScanPolicies: The enumeration literal shall specify supported policies that trigger an AntiVirus scan.

string	Description
None	This enumeration literal specifies No trigger.
OnFirstRead	This enumeration literal specifies to trigger on first read.
OnPatternUpdate	This enumeration literal specifies to trigger on antivirus pattern file update.
OnRename	This enumeration literal specifies to trigger on object rename.
OnUpdate	This enumeration literal specifies to trigger on object update.

9.5.10.4.2 SupportedChannelEncryptionStrengths: The enumeration literal shall specify supported key sizes in a symmetric encryption algorithm (AES) for transport channel encryption.

string	Description
Bits_0	This enumeration literal specifies that there is no key.
Bits_112	This enumeration literal specifies a 3DES 112 bit key.
Bits_128	This enumeration literal specifies an AES 128 bit key.
Bits_192	This enumeration literal specifies an AES 192 bit key.
Bits_256	This enumeration literal specifies an AES 256 bit key.

9.5.10.4.3 SupportedDataSanitizationPolicies: The enumeration literal shall specify supported data sanitization policies.

string	Description
Clear	This enumeration literal specifies to sanitize data in all user-addressable storage locations for protection against simple non-invasive data recovery techniques.
CryptographicErase	This enumeration literal specifies to leverages the encryption of target data by enabling sanitization of the target data's encryption key. This leaves only the ciphertext remaining on the media, effectively sanitizing the data by preventing read-access. For more information, see NIST800-88 and ISO/IEC 27040.
None	This enumeration literal specifies no sanitization.

9.5.10.4.4 SupportedHostAuthenticationTypes: The enumeration literal shall specify supported authentication types for hosts (servers) or initiator endpoints.

string	Description
None	This enumeration literal specifies No authentication.
Password	This enumeration literal specifies Password/shared-secret: Absent an distributed authentication infrastructure, this is what is typically done.
PKI	This enumeration literal specifies a Public Key Infrastructure. Customers with the highest assurance requirements roll PKI out to hosts and users (it is more common for hosts than users. User PKI-based authentication has significant operational complications and administrative overheads, e.g., smart cards may be involved.
Ticket	This enumeration literal specifies Ticket-based (e.g., Kerberos): This is the most common class of authentication infrastructure used in enterprises. Kerberos is the best known example, and Windows usage of that via Active Directory is so widely deployed as to be a de facto standard. In other areas (e.g., academia) there are comparable ticket-based systems.

9.5.10.4.5 SupportedMediaEncryptionStrengths: The enumeration literal shall specify supported key sizes in a symmetric encryption algorithm (AES) for media encryption.

string	Description
Bits_0	This enumeration literal specifies that there is no key.
Bits_112	This enumeration literal specifies a 3DES 112 bit key.

string	Description
Bits_128	This enumeration literal specifies an AES 128 bit key.
Bits_192	This enumeration literal specifies an AES 192 bit key.
Bits_256	This enumeration literal specifies an AES 256 bit key.

9.5.10.4.6 SupportedSecureChannelProtocols: The enumeration literal shall specify supported protocols that provide encrypted communication.

string	Description
IPsec	This enumeration literal specifies Internet Protocol Security (IPsec), as defined by IETF RFC 2401.
None	This enumeration literal specifies no encryption.
RPCSEC_GSS	This enumeration literal specifies RPC access to the Generic Security Services Application Programming Interface (GSS-API), as defined by IETF RPC 2203.
TLS	This enumeration literal specifies Transport Layer Security (TLS), as defined by IETF RFC 5246.

9.5.10.4.7 SupportedUserAuthenticationTypes: The enumeration literal shall specify supported authentication types for users (or programs).

string	Description
None	This enumeration literal specifies No authentication.
Password	This enumeration literal specifies Password/shared-secret: Absent an distributed authentication infrastructure, this is what is typically done.

string	Description
PKI	This enumeration literal specifies a Public Key Infrastructure. Customers with the highest assurance requirements roll PKI out to hosts and users (it is more common for hosts than users. User PKI-based authentication has significant operational complications and administrative overheads, e.g., smart cards may be involved.
Ticket	This enumeration literal specifies Ticket-based (e.g., Kerberos): This is the most common class of authentication infrastructure used in enterprises. Kerberos is the best known example, and Windows usage of that via Active Directory is so widely deployed as to be a de facto standard. In other areas (e.g., academia) there are comparable ticket-based systems.

9.5.11 DataStorageLineOfService 1.3.1

9.5.11.1 Description   This structure may be used to describe a service option covering storage provisioning and availability.

9.5.11.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/DataStor
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/DataStorageLinesOfService/{*DataStorageLineO*

9.5.11.3 Properties

Property	Type	Attributes	Notes
Acces sCapabilities *(v1.1+)* [ ]	array (string(enum))	• read-write(null)*	Each entry specifies a required storage access capability. *For the possible property values, see Acc essCapabilities in Property details.*
Actions *(v1.3+)* {}	object		The Actions property shall contain the available actions for this resource.
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
IsS paceEfficient	boolean	*re ad-write(null)*	A value of true shall indicate that the storage is compressed or deduplicated. The default value for this property is false.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
Provi sioningPolicy	string(enum)	• read-write(null)*	The enumeration literal shall define the provisioning policy for storage. *For the possible property values, see Pro visioningPolicy in Property details.*
Re coverableCapaci tySourceCount *(v1.2+)*	integer	*re ad-write(null)*	The value is minimum required number of available capacity source resources that shall be available in the event that an equivalent capacity source resource fails. It is assumed that drives and memory components can be replaced, repaired or otherwise added to increase an associated resource's R ecoverableCapac itySourceCount.

Property	Type	Attributes	Notes
RecoveryT imeObjectives	string(enum)	• read-write(null)*	The enumeration literal specifies the time after a disaster that the client shall regain conformant service level access to the primary store, typical values are 'immediate' or 'offline'. The expectation is that the services required to implement this capability are part of the advertising system. *For the possible property values, see Recover yTimeObjectives in Property details.*

9.5.11.4 Property details

9.5.11.4.1 AccessCapabilities:　Each entry specifies a required storage access capability.

string	Description
Append	This enumeration literal shall indicate that the storage may be written only to append.
Execute	This value shall indicate that Execute access is allowed by the file share.

string	Description
Read	This enumeration literal shall indicate that the storage may be read.
Streaming	This enumeration literal shall indicate that the storage may be read sequentially.
Write	This enumeration literal shall indicate that the storage may be written multiple times.
WriteOnce	This enumeration literal shall indicate that the storage may be written only once.

9.5.11.4.2 ProvisioningPolicy: The enumeration literal shall define the provisioning policy for storage.

string	Description
Fixed	This enumeration literal specifies storage shall be fully allocated.
Thin	This enumeration literal specifies storage may be over allocated.

9.5.11.4.3 RecoveryTimeObjectives: The enumeration literal specifies the time after a disaster that the client shall regain conformant service level access to the primary store, typical values are 'immediate' or 'offline'. The expectation is that the services required to implement this capability are part of the advertising system.

string	Description
Nearline	Access to a replica shall be consistent with switching access to a different path through a different front-end interconnection infrastructure. Some inconsistency may occur. A restore step may be required before recovery can commence.
Offline	Access to a replica may take a significant amount of time. No direct connection to the replica is assumed. Some inconsistency loss may occur. A restore step is likely to be required.
OnlineActive	Access to synchronous replicas shall be instantaneous.
OnlinePassive	Access to a synchronous replica shall be consistent with switching access to a different path the same front-end interconnect. A restore step shall not be required.

9.5.12 DataStorageLoSCapabilities 1.2.2

9.5.12.1 Description   Each instance of DataStorageLoSCapabilities describes capabilities of the system to support various data storage service options.

9.5.12.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/DataStorageLoSCapabilities

9.5.12.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Identifier {}	object		The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).

Property	Type	Attributes	Notes
MaximumRe coverableCapaci tySourceCount *(v1.2+)*	integer	*re ad-write(null)*	The maximum number of capacity source resources that can be supported for the purpose of recovery when in the event that an equivalent capacity source resource fails.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
• *SupportedAcce sCapabili- ties** [ ]	array (string(enum))	• read- write(null)*	Each entry specifies a storage access capability. *For the possible property values, see SupportedAcc essCapabilities in Property details.*
SupportedL inesOfService [{	array		The collection shall contain known and supported DataStorage LinesOfService.
@odata.id	string	*read-write*	Link to a DataStora geLineOfService resource. See the Links section and the *DataStorag eLineOfService* schema for details.
}]			

Property	Type	Attributes	Notes
S upportedProvisi oningPolicies [ ]	array (string(enum))	• read-write(null)*	This collection specifies supported storage allocation policies. *For the possible property values, see SupportedProvi sioningPolicies in Property details.*
Sup portedRecoveryT imeObjectives [ ]	array (string(enum))	• read-write(null)*	This collection specifies supported expectations for time to access the primary store after recovery. *For the possible property values, see S upportedRecover yTimeObjectives in Property details.*
SupportsSp aceEfficiency	boolean	*re ad-write(null)*	The value specifies whether storage compression or deduplication is supported. The default value for this property is false.

9.5.12.4 Property details

9.5.12.4.1 SupportedAccessCapabilities: Each entry specifies a storage access capability.

string	Description
Append	This enumeration literal shall indicate that the storage may be written only to append.
Execute	This value shall indicate that Execute access is allowed by the file share.
Read	This enumeration literal shall indicate that the storage may be read.
Streaming	This enumeration literal shall indicate that the storage may be read sequentially.
Write	This enumeration literal shall indicate that the storage may be written multiple times.
WriteOnce	This enumeration literal shall indicate that the storage may be written only once.

9.5.12.4.2 SupportedProvisioningPolicies: This collection specifies supported storage allocation policies.

string	Description
Fixed	This enumeration literal specifies storage shall be fully allocated.
Thin	This enumeration literal specifies storage may be over allocated.

9.5.12.4.3 SupportedRecoveryTimeObjectives: This collection specifies supported expectations for time to access the primary store after recovery.

string	Description
Nearline	Access to a replica shall be consistent with switching access to a different path through a different front-end interconnection infrastructure. Some inconsistency may occur. A restore step may be required before recovery can commence.
Offline	Access to a replica may take a significant amount of time. No direct connection to the replica is assumed. Some inconsistency loss may occur. A restore step is likely to be required.
OnlineActive	Access to synchronous replicas shall be instantaneous.
OnlinePassive	Access to a synchronous replica shall be consistent with switching access to a different path the same front-end interconnect. A restore step shall not be required.

9.5.13 FeaturesRegistry 1.1.0

9.5.13.1 Description   This resource shall be used to represent a Feature registry for a Redfish implementation.

9.5.13.2 URIs   /redfish/v1/Registries

9.5.13.3 Properties

Property	Type	Attributes	Notes
Actions {}	object		The Actions property shall contain the available actions for this resource.
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Features {	object	* required*	The pattern property shall represent the suffix to be used in the FeatureId and shall be unique within this message registry.
(pattern) {	object		Property names follow regular expression pattern ” [A-Za-z0-9]+”
Co rrespondingProf ileDefinition	string	*read-only required(null)*	If present, the value shall define a profile definition that contains the named profile declaration.

Property	Type	Attributes	Notes
Description	string	*read-only required(null)*	The value shall be a detailed description of the feature.
FeatureName	string	*read-only required(null)*	The value shall be the unique name of the feature prefixed by the defining organization separated by a period (e.g. 've ndor.feature').
Version	string	*read-only required(null)*	The value shall uniquely identify the version of the feature, using the maj or.minor.errata format.
}			
}			
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Language	string	*read-only required*	The value of this property shall be a string consisting of an RFC 5646 language code.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
• *OwningEntity**	string	*read-only required*	The value of this property shall be a string that represents the publisher of this registry.
R egistryPrefix	string	*read-only required*	The value of this property shall be the prefix used in IDs which uniquely identifies all of the Features in this registry as belonging to this registry.
Re gistryVersion	string	*read-only required*	The value of this property shall be the version of this message registry. The format of this string shall be of the format maj orversion.minor version.errata.

9.5.14 FileShare 1.2.0

9.5.14.1 Description   This resource shall be used to represent a shared set of files with a common directory structure.

9.5.14.2 URIs   /redfish/v1/Storage/{*StorageId*}/FileSystems/{*FileSystemsId*}/ExportedFileShares/{*ExportedFil*
/redfish/v1/StorageServices/{*StorageServiceId*}/FileSystems/{*FileSystemsId*}/ExportedFileShares/{*ExportedFileS*
/redfish/v1/Systems/{*ComputerSystemsId*}/Storage/{*StorageId*}/FileSystems/{*FileSystemsId*}/ExportedFileShare

9.5.14.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.
CASupported	boolean	*re ad-write(null)*	The value of this property shall indicate that Continuous Availability is supported. Client/Server mediated recovery from network and server failure with application transparency. This property shall be NULL unless the FileS haringProtocols property includes SMB. The default value for this property is false.

Property	Type	Attributes	Notes
DefaultAcces sCapabilities [ ]	array (string(enum))	*read-only(null)*	The value of this property shall be an array containing entries for the default access capabilities for the file share. Each entry shall specify a default access privilege. The types of default access can include Read, Write, and/or Execute. *For the possible property values, see DefaultAcc essCapabilities in Property details.*
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Ether netInterfaces {	object		The value shall be a link to an EthernetInte rfaceCollection with members that provide access to the file share.

Property	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			
E xecuteSupport	boolean	*r ead-only(null)*	The value of this property shall indicate whether Execute access is supported by the file share. The default value for this property is false.
FileSharePath	string	*r ead-only(null)*	The value of this property shall be a path (relative to the file system root) to the exported file or directory on the file system where this file share is hosted.

Property	Type	Attributes	Notes
FileS hareQuotaType	string(enum)	• read-write(null)*	If Fil eShareQuotaType is present, a value of Soft shall specify that quotas are not enforced, and a value of Hard shall specify that writes shall fail if the space consumed would exceed the value of the FileShare TotalQuotaBytes property. *For the possible property values, see Fil eShareQuotaType in Property details.*
FileShareRemain ingQuotaBytes	integer(By)	*read-only(null)*	If present, the value of this property shall indicate the remaining number of bytes that may be consumed by this file share.
FileShareTo talQuotaBytes	integer(By)	• read-write(null)*	If present, the value of this property shall indicate the maximum number of bytes that may be consumed by this file share.

Property	Type	Attributes	Notes
FileSha ringProtocols [ ]	array (string(enum))	*read-only(null)*	This property shall be an array containing entries for the file sharing protocols supported by this file share. Each entry shall specify a file sharing protocol supported by the file system. *For the possible property values, see FileS haringProtocols in Property details.*
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Links {	object		The Links property, as described by the Redfish Specification, shall contain references to resources that are related to, but not contained by (subordinate to), this resource.

Property	Type	Attributes	Notes
C lassOfService {	object		This value shall be a link to the ClassOfService for this file share. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-only*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			
FileSystem {	object		The value shall be a link to the file system containing the file share. See the *FileSystem* schema for details on this property.
@odata.id	string	*read-only*	Link to a FileSystem resource. See the Links section and the *FileSystem* schema for details.
}			

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.
}			

Property	Type	Attributes	Notes
LowS paceWarningThre sholdPercents [ ]	array(%) (integer, null)	*read-write*	This property shall be an array containing entries for the percentages of file share capacity at which low space warning events are be issued. A LOW_SPACE_TH RESH- OLD_WARNING event shall be triggered each time the remaining file share capacity value becomes less than one of the values in the array. The following shall be true: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(Consum edBytes))/SUM(A llocatedBytes).

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
RemainingCapacityPercent *(v1.1+)*	integer	*r ead-only(null)*	If present, this value shall return {[(SUM(AllocatedBytes) - S UM(ConsumedByte s)]/SUM(Alloca tedBytes)}*100 represented as an integer value.
RootAccess	boolean	*r ead-only(null)*	The value of this property shall indicate whether Root access is allowed by the file share. The default value for this property is false.
Status {}	object		This value of this property shall indicate the status of the file share. For property details, see Status.
WritePolicy	string(enum)	*read-only(null)*	The value of this property shall define how writes are replicated to the shared source. *For the possible property values, see WritePolicy in Property details.*

9.5.14.4 Property details

9.5.14.4.1 DefaultAccessCapabilities: The value of this property shall be an array containing entries for the default access capabilities for the file share. Each entry shall specify a default access privilege. The types of default access can include Read, Write, and/or Execute.

string	Description
Append	This enumeration literal shall indicate that the storage may be written only to append.
Execute	This value shall indicate that Execute access is allowed by the file share.
Read	This enumeration literal shall indicate that the storage may be read.
Streaming	This enumeration literal shall indicate that the storage may be read sequentially.
Write	This enumeration literal shall indicate that the storage may be written multiple times.
WriteOnce	This enumeration literal shall indicate that the storage may be written only once.

9.5.14.4.2 FileShareQuotaType: If FileShareQuotaType is present, a value of Soft shall specify that quotas are not enforced, and a value of Hard shall specify that writes shall fail if the space consumed would exceed the value of the FileShareTotalQuotaBytes property.

string	Description
Hard	This value shall indicate that quotas are enabled and enforced.
Soft	This value shall indicate that quotas are enabled but not enforced.

9.5.14.4.3 FileSharingProtocols: This property shall be an array containing entries for the file sharing protocols supported by this file share. Each entry shall specify a

file sharing protocol supported by the file system.

string	Description
NFSv3	This value shall indicate that NFSv3, as defined in RFC 1813, is supported by the file system.
NFSv4_0	This value shall indicate that NFSv4, as defined in RFC 7530, is supported by the file system.
NFSv4_1	This value shall indicate that NFSv4.1, as defined in RFC 5661, is supported by the file system.
SMBv2_0	This value shall indicate that Server Message Block version 2.0 is supported by the file system.
SMBv2_1	This value shall indicate that Server Message Block version 2.1 is supported by the file system.
SMBv3_0	This value shall indicate that Server Message Block version 3.0 is supported by the file system.
SMBv3_0_2	This value shall indicate that Server Message Block version 3.0.2 is supported by the file system.
SMBv3_1_1	This value shall indicate that Server Message Block version 3.1.1 is supported by the file system.

9.5.14.4.4 WritePolicy: The value of this property shall define how writes are replicated to the shared source.

string	Description
Active	This enumeration literal shall indicate Active-Active (i.e. bidirectional) synchronous updates.
Adaptive	This enumeration literal shall indicate that an implementation may switch between synchronous and asynchronous modes.
Asynchronous	This enumeration literal shall indicate Asynchronous updates.
Synchronous	This enumeration literal shall indicate Synchronous updates.

9.5.15 FileShareCollection

9.5.15.1 URIs   /redfish/v1/Storage/*{StorageId}*/FileSystems/*{FileSystemsId}*/ExportedFileShares
/redfish/v1/StorageServices/*{StorageServiceId}*/FileSystems/*{FileSystemsId}*/ExportedFileShares

9.5.15.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Members [{	array		This property shall contain references to the members of this FileSystem collection.
@odata.id	string	*read-only*	Link to a FileShare resource. See the Links section and the *FileShare* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.

9.5.16 FileSystem 1.2.2

9.5.16.1 Description   This resource shall be used to represent an instance of a hierarchical namespace of files.

9.5.16.2 URIs   /redfish/v1/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}
/redfish/v1/StorageServices/{*StorageServiceId*}/FileSystems/{*FileSystemId*}

9.5.16.3 Properties

Property	Type	Attributes	Notes
Acces sCapabilities [ ]	array (string(enum))	• read-write(null)*	This property shall be an array containing entries for the supported IO access capabilities. Each entry shall specify a current storage access capability. *For the possible property values, see Acc essCapabilities in Property details.*
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.
B lockSizeBytes	integer(By)	*read-only(null)*	The value of this property shall be the block size of the file system in bytes.
Capacity {}	object		The value of this property shall be the capacity allocated to the file system in bytes. For property details, see Capacity v1.0.0).

Property	Type	Attributes	Notes
Ca pacitySources [{	array		This property shall be an array containing entries for all the capacity sources for the file system. Each entry shall provide capacity allocation information from a named resource.
@odata.id	string	*read-write*	Link to a CapacitySource resource. See the Links section and the *CapacitySource* schema for details.
}]			
CasePreserved	boolean	*re ad-write(null)*	This property shall indicate that the case of file names is preserved by the file system. A value of True shall indicate that case of file names shall be preserved.

Property	Type	Attributes	Notes
CaseSensitive	boolean	*re ad-write(null)*	This property shall indicate that case sensitive file names are supported by the file system. A value of True shall indicate that file names are case sensitive.
Cha racterCodeSet [ ]	array (string(enum))	• read-write(null)*	This property shall be an array containing entries for the character sets or encodings supported by the file system. Each entry shall specify a character set encoding supported by the file system. *For the possible property values, see C haracterCodeSet in Property details.*

Property	Type	Attributes	Notes
Clu sterSizeBytes	integer(By)	• read-write(null)*	This value shall specify the minimum file allocation size imposed by the file system. This minimum allocation size shall be the smallest amount of storage allocated to a file by the file system. Under stress conditions, the file system may allocate storage in amounts smaller than this value.
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
E xportedShares {	object		This property shall be an array of exported file shares of this file system. Each entry shall define an exported file share of this file system. Contains a link to a resource.
@odata.id	string	*read-write*	Link to Collection of *FileShare*. See the FileShare schema for details.
}			
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Identifiers *(v1.1.1+)* [{ }]	array (object)		This property shall contain a list of all known durable names for this file system. For property details, see Identifier v1.12.0).

Property	Type	Attributes	Notes
I mportedShares *(v1.0.1+)* [{	array		The value shall be an array of imported file shares.
ImportedShare }]		*read-write*	
• *IOStatistics** *(v1.2+)* {}	object		The value shall represent IO statistics for this FileSystem. For property details, see IOStatistics.
Links {	object		This property shall contain links to other resources that are related to this resource.
C lassOfService {	object		This value shall be a link to the ClassOfService for this file system. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-only*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.
Repl icaCollection [{	array		This property shall be an array of links to replicas for this file system. Each entry shall be a link to a replica for this file system.
@odata.id	string	*read-only*	Link to another FileSystem resource.
}]			
Spar eResourceSets *(v1.2+)* [{	array		Each referenced S pareResourceSet shall contain resources that may be utilized to replace the capacity provided by a failed resource having a compatible type.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a S pareResourceSet resource. See the Links section and the *Sp areResourceSet* schema for details.

   }]

}

Property	Type	Attributes	Notes
LowS paceWarningThre sholdPercents [ ]	array(%) (integer, null)	*read-write*	This property shall be an array containing entries for the percentages of file system capacity at which low space warning events are be issued. A LOW_SPACE_TH RESH-OLD_WARNING event shall be triggered each time the remaining file system capacity value becomes less than one of the values in the array. The following shall be true: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(Consum edBytes))/SUM(A llocatedBytes).
MaxFileNa meLengthBytes	integer(By)	• read-write(null)*	If specified, this value shall specify the maximum length of a file name within the file system.

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
Re coverableCapaci tySourceCount (v1.2+)	integer	re ad-write(null)	The value is the number of available capacity source resources currently available in the event that an equivalent capacity source resource fails.
Rema iningCapacity {}	object		The value of this property shall be the remaining capacity allocated to the file system in bytes. For property details, see Capacity v1.0.0).
RemainingCa pacityPercent (v1.1+)	integer	r ead-only(null)	If present, this value shall return {[(SUM(AllocatedBytes) - S UM(ConsumedByte s)]/SUM(Alloca tedBytes)}*100 represented as an integer value.

Property	Type	Attributes	Notes
ReplicaInfo {	object		If this file system is a replica, this value shall describe its replication attributes. This value shall not be present if this file system is not a replica. A file system may be both a source and a replica. See the *Stor ageReplicaInfo* schema for details on this property.
@odata.id	string	*read-only*	Link to a ReplicaInfo resource. See the Links section and the *Stor ageReplicaInfo* schema for details.
}			
R eplicaTargets *(v1.2.1+)* [{	array		The value shall reference the target replicas that are sourced by this replica.

Property	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.

}]

9.5.16.4 Property details

9.5.16.4.1 AccessCapabilities: This property shall be an array containing entries for the supported IO access capabilities. Each entry shall specify a current storage access capability.

string	Description
Append	This enumeration literal shall indicate that the storage may be written only to append.
Execute	This value shall indicate that Execute access is allowed by the file share.
Read	This enumeration literal shall indicate that the storage may be read.
Streaming	This enumeration literal shall indicate that the storage may be read sequentially.
Write	This enumeration literal shall indicate that the storage may be written multiple times.

string	Description
WriteOnce	This enumeration literal shall indicate that the storage may be written only once.

9.5.16.4.2 CharacterCodeSet: This property shall be an array containing entries for the character sets or encodings supported by the file system. Each entry shall specify a character set encoding supported by the file system.

string	Description
ASCII	This value shall indicate that the ASCII character encoding is supported by the file system.
ExtendedUNIXCode	This value shall indicate that Extended Unix Code character encoding is supported by the file system.
ISO2022	This value shall indicate that ISO-2022 character encoding is supported by the file system.
ISO8859_1	This value shall indicate that ISO-8859-1 character encoding is supported by the file system.
UCS_2	This value shall indicate that the UCS-2 character encoding is supported by the file system.
Unicode	This value shall indicate that Unicode character encoding is supported by the file system.
UTF_16	This value shall indicate that the UTF-16 character encoding is supported by the file system.

string	Description
UTF_8	This value shall indicate that the UTF-8 character encoding is supported by the file system.

9.5.17 FileSystemCollection

9.5.17.1 URIs   /redfish/v1/Storage/*{StorageId}*/FileSystems /redfish/v1/StorageServices/*{StorageServiceId}*/F

9.5.17.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		This property shall contain references to the members of this FileSystem collection.
@odata.id	string	*read-only*	Link to a FileSystem resource. See the Links section and the *FileSystem* schema for details.
}]			

Property	Type	Attributes	Notes
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.18 HostedStorageServices

9.5.18.1 URIs   /redfish/v1/Systems/*{ComputerSystemId}*/HostedServices

9.5.18.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Members [{	array		The value of each member entry shall reference a StorageService resource.
@odata.id	string	*read-only*	Link to a StorageService resource. See the Links section and the *StorageService* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.

9.5.19 IOConnectivityLineOfService 1.2.1

9.5.19.1 Description   An IO connectivity service option may be used to specify the characteristics of storage connectivity.

9.5.19.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/IOConne
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/IOConnectivityLinesOfService/{*IOConnectivityL*

9.5.19.3 Properties

Property	Type	Attributes	Notes
Ac cessProtocols [ ]	array (string(enum))	• read-write(null)*	The Enumeration Literal shall specify the Access protocol for this service option. NOTE: If multiple protocols are specified, the corresponding M axSupportedIOPS governs the max achieved across all protocol uses. This may be less than the sum of the individual max values, which may be specified by individual Line of Service entries. *For the possible property values, see AccessProtocols in Property details.*
Actions *(v1.2+)* {}	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
MaxB ytesPerSecond *(v1.1+)*	integer(By/s)	• read-write(null)*	The value shall be the maximum bytes per second that a connection can support.
MaxIOPS *(v1.1+)*	in teger([IO]/s)	• read-write(null)*	The value shall be the maximum IOs per second that the connection shall allow for the selected access protocol.

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.19.4 Property details

9.5.19.4.1 AccessProtocols: The Enumeration Literal shall specify the Access protocol for this service option. NOTE: If multiple protocols are specified, the corresponding MaxSupportedIOPS governs the max achieved across all protocol uses. This may be

less than the sum of the individual max values, which may be specified by individual Line of Service entries.

string	Description
AHCI	This value shall indicate conformance to the Intel Advanced Host Controller Interface (AHCI) Specification.
DisplayPort	This value shall indicate conformance to the VESA DisplayPort Specification.
DVI	This value shall indicate conformance to the Digital Display Working Group DVI-A, DVI-D, or DVI-I Specification.
Ethernet	This value shall indicate conformance to the IEEE 802.3 Ethernet specification.
FC	This value shall indicate conformance to the T11 Fibre Channel Physical and Signaling Interface Specification.
FCoE	This value shall indicate conformance to the T11 FC-BB-5 Specification.
FCP	This value shall indicate conformance to the INCITS 481: Information Technology - Fibre Channel Protocol for SCSI.
FICON	This value shall indicate conformance to the ANSI FC-SB-3 Single-Byte Command Code Sets-3 Mapping Protocol for the Fibre Channel (FC) protocol. Fibre Connection (FICON) is the IBM-proprietary name for this protocol.
FTP	This value shall indicate conformance to the RFC114-defined File Transfer Protocol (FTP).
GenZ	This value shall indicate conformance to the Gen-Z Core Specification.
HDMI	This value shall indicate conformance to the HDMI Forum HDMI Specification.

string	Description
HTTP	This value shall indicate conformance to the Hypertext Transport Protocol (HTTP) as defined by RFC3010 or RFC5661.
HTTPS	This value shall indicate conformance to the Hypertext Transfer Protocol Secure (HTTPS) as defined by RFC2068 or RFC2616, which uses Transport Layer Security (TLS) as defined by RFC5246 or RFC6176.
I2C	This value shall indicate conformance to the NXP Semiconductors I2C-bus Specification.
InfiniBand	This value shall indicate conformance to the InfiniBand Architecture Specification-defined InfiniBand protocol.
iSCSI	This value shall indicate conformance to the IETF Internet Small Computer Systems Interface (iSCSI) Specification.
iWARP	This value shall indicate conformance to the RFC5042-defined Internet Wide Area RDMA Protocol (iWARP) that uses the transport layer mechanisms as defined by RFC5043 or RFC5044.
MultiProtocol	This value shall indicate conformance to multiple protocols.
NFSv3	This value shall indicate conformance to the RFC1813-defined Network File System (NFS) protocol.
NFSv4	
NVLink	This value shall indicate conformance to the NVIDIA NVLink protocol.

string	Description
NVMe	This value shall indicate conformance to the Non-Volatile Memory Host Controller Interface Specification.
NVMeOverFabrics	This value shall indicate conformance to the NVM Express over Fabrics Specification.
OEM	This value shall indicate conformance to an OEM-specific architecture and the OEM section may include additional information.
PCIe	This value shall indicate conformance to the PCI-SIG PCI Express Base Specification.
RoCE	This value shall indicate conformance to the InfiniBand Architecture Specification-defined RDMA over Converged Ethernet Protocol.
RoCEv2	This value shall indicate conformance to the InfiniBand Architecture Specification-defined RDMA over Converged Ethernet Protocol version 2.
SAS	This value shall indicate conformance to the T10 SAS Protocol Layer Specification.
SATA	This value shall indicate conformance to the Serial ATA International Organization Serial ATA Specification.
SFTP	This value shall indicate conformance to the RFC114-defined SSH File Transfer Protocol (SFTP) that uses Transport Layer Security (TLS) as defined by RFC5246 or RFC6176.

string	Description
SMB	This value shall indicate conformance to the Server Message Block (SMB), or Common Internet File System (CIFS), protocol.
TCP	This value shall indicate conformance to the IETF-defined Transmission Control Protocol (TCP). For example, RFC7414 defines the roadmap of the TCP specification.
TFTP	This value shall indicate conformance to the IETF-defined Trivial File Transfer Protocol (TFTP). For example, RFC1350 defines the core TFTP version 2 specification.
UDP	This value shall indicate conformance to the IETF-defined User Datagram Protocol (UDP). For example, RFC768 defines the core UDP specification.
UHCI	This value shall indicate conformance to the Intel Universal Host Controller Interface (UHCI) Specification, Enhanced Host Controller Interface Specification, or the Extensible Host Controller Interface Specification.
USB	This value shall indicate conformance to the USB Implementers Forum Universal Serial Bus Specification.
VGA	This value shall indicate conformance to the VESA SVGA Specification.

9.5.20 IOConnectivityLoSCapabilities 1.2.0

9.5.20.1 Description   Each instance of IOConnectivityLoSCapabilities describes capabilities of the system to support various IO Connectivity service options.

9.5.20.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/IOConnectivityLoSCapabilities

9.5.20.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Identifier {}	object		The value identifies this resource. The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).
MaxSupportedBytesPerSecond	integer(By/s)	• read-write(null)*	The value shall be the maximum bytes per second that a connection can support.
Max SupportedIOPS *(v1.1+)*	in teger([IO]/s)	• read-write(null)*	The value shall be the maximum IOPS that a connection can support.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
SupportedAc cessProtocols [ ]	array (string(enum))	• read-write(null)*	Access protocols supported by this service option. NOTE: SMB+NFS* requires that SMB and at least one of NFSv3 or NFXv4 are also selected, (i.e. {'SMB', 'NFSv4', 'SMB+NFS'}). For the possible property values, see Supported AccessProtocols in Property details.*
SupportedL inesOfService [{	array		The collection shall contain known and supported IOConnectivity LinesOfService.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a IOConnectivi tyLineOfService resource. See the Links section and the *IOConnectivit yLineOfService* schema for details.

}]

9.5.20.4 Property details

9.5.20.4.1 SupportedAccessProtocols: Access protocols supported by this service option. NOTE: SMB+NFS* requires that SMB and at least one of NFSv3 or NFXv4 are also selected, (i.e. {'SMB', 'NFSv4', 'SMB+NFS*'}).

string	Description
AHCI	This value shall indicate conformance to the Intel Advanced Host Controller Interface (AHCI) Specification.
DisplayPort	This value shall indicate conformance to the VESA DisplayPort Specification.
DVI	This value shall indicate conformance to the Digital Display Working Group DVI-A, DVI-D, or DVI-I Specification.
Ethernet	This value shall indicate conformance to the IEEE 802.3 Ethernet specification.
FC	This value shall indicate conformance to the T11 Fibre Channel Physical and Signaling Interface Specification.
FCoE	This value shall indicate conformance to the T11 FC-BB-5 Specification.

string	Description
FCP	This value shall indicate conformance to the INCITS 481: Information Technology - Fibre Channel Protocol for SCSI.
FICON	This value shall indicate conformance to the ANSI FC-SB-3 Single-Byte Command Code Sets-3 Mapping Protocol for the Fibre Channel (FC) protocol. Fibre Connection (FICON) is the IBM-proprietary name for this protocol.
FTP	This value shall indicate conformance to the RFC114-defined File Transfer Protocol (FTP).
GenZ	This value shall indicate conformance to the Gen-Z Core Specification.
HDMI	This value shall indicate conformance to the HDMI Forum HDMI Specification.
HTTP	This value shall indicate conformance to the Hypertext Transport Protocol (HTTP) as defined by RFC3010 or RFC5661.
HTTPS	This value shall indicate conformance to the Hypertext Transfer Protocol Secure (HTTPS) as defined by RFC2068 or RFC2616, which uses Transport Layer Security (TLS) as defined by RFC5246 or RFC6176.
I2C	This value shall indicate conformance to the NXP Semiconductors I2C-bus Specification.
InfiniBand	This value shall indicate conformance to the InfiniBand Architecture Specification-defined InfiniBand protocol.

string	Description
iSCSI	This value shall indicate conformance to the IETF Internet Small Computer Systems Interface (iSCSI) Specification.
iWARP	This value shall indicate conformance to the RFC5042-defined Internet Wide Area RDMA Protocol (iWARP) that uses the transport layer mechanisms as defined by RFC5043 or RFC5044.
MultiProtocol	This value shall indicate conformance to multiple protocols.
NFSv3	This value shall indicate conformance to the RFC1813-defined Network File System (NFS) protocol.
NFSv4	
NVLink	This value shall indicate conformance to the NVIDIA NVLink protocol.
NVMe	This value shall indicate conformance to the Non-Volatile Memory Host Controller Interface Specification.
NVMeOverFabrics	This value shall indicate conformance to the NVM Express over Fabrics Specification.
OEM	This value shall indicate conformance to an OEM-specific architecture and the OEM section may include additional information.
PCIe	This value shall indicate conformance to the PCI-SIG PCI Express Base Specification.
RoCE	This value shall indicate conformance to the InfiniBand Architecture Specification-defined RDMA over Converged Ethernet Protocol.

string	Description
RoCEv2	This value shall indicate conformance to the InfiniBand Architecture Specification-defined RDMA over Converged Ethernet Protocol version 2.
SAS	This value shall indicate conformance to the T10 SAS Protocol Layer Specification.
SATA	This value shall indicate conformance to the Serial ATA International Organization Serial ATA Specification.
SFTP	This value shall indicate conformance to the RFC114-defined SSH File Transfer Protocol (SFTP) that uses Transport Layer Security (TLS) as defined by RFC5246 or RFC6176.
SMB	This value shall indicate conformance to the Server Message Block (SMB), or Common Internet File System (CIFS), protocol.
TCP	This value shall indicate conformance to the IETF-defined Transmission Control Protocol (TCP). For example, RFC7414 defines the roadmap of the TCP specification.
TFTP	This value shall indicate conformance to the IETF-defined Trivial File Transfer Protocol (TFTP). For example, RFC1350 defines the core TFTP version 2 specification.
UDP	This value shall indicate conformance to the IETF-defined User Datagram Protocol (UDP). For example, RFC768 defines the core UDP specification.

string	Description
UHCI	This value shall indicate conformance to the Intel Universal Host Controller Interface (UHCI) Specification, Enhanced Host Controller Interface Specification, or the Extensible Host Controller Interface Specification.
USB	This value shall indicate conformance to the USB Implementers Forum Universal Serial Bus Specification.
VGA	This value shall indicate conformance to the VESA SVGA Specification.

9.5.21 IOPerformanceLineOfService 1.1.1

9.5.21.1 Description   This structure may be used to define a service option related to IO performance.

9.5.21.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/IOPerfor
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/IOPerformanceLinesOfService/{*IOPerformance.*

9.5.21.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
AverageIO OperationLatenc yMicroseconds	integer(us)	• read-write(null)*	The value shall be the expected average IO latency in microseconds calculated over sample periods (see Sample PeriodSeconds).
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
IO OperationsPerSe condIsLimited	boolean	*re ad-write(null)*	If true, the system should not allow IOPS to exceed MaxIo OperationsPerSe condPerTerabyte * VolumeSize. Otherwise, the system shall not enforce a limit. The default value for this property is false.
IOWorkload {}	object		The value shall be a description of the expected workload. The workload provides the context in which the values of MaxIO OperationsPerSe condPerTerabyte and Average IOOperationLate ncyMicroseconds are expected to be achievable. For property details, see IOWorkload v1.0.0).

Property	Type	Attributes	Notes
MaxIOOp erationsPerSeco ndPerTerabyte	i nteger(1/s/TBy)	• read-write(null)*	The value shall be the amount of IOPS a volume of a given committed size in Terabytes can support. This IOPS density value is useful as a metric that is independent of capacity. Cost is a function of this value and the AverageI OOperationLaten cyMicroseconds.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
• *SamplePeriod**	string	re ad-write(null)	The value shall be an ISO 8601 duration specifying the sampling period over which average values are calculated.

9.5.22 IOPerformanceLoSCapabilities 1.3.0

9.5.22.1 Description   Each instance of IOPerformanceLoSCapabilities shall describe the capabilities of the system to support various IO performance service options.

9.5.22.2 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/IOPerformanceLoSCapabilities

9.5.22.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.1+)* {}	object		The Actions property shall contain the available actions for this resource.
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Identifier {}	object		The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).

Property	Type	Attributes	Notes
IOLimiti ngIsSupported	boolean	*re ad-write(null)*	If true, the system should limit IOPS to MaxIO OperationsPerSe condPerTerabyte * (Volume Size in Terabytes). Otherwise, the system shall not inforce a limit. The default value for this property is false.
Ma xSamplePeriod	string(s)	• read-write(null)*	The value shall be an ISO 8601 duration specifying the maximum sampling period over which average values are calculated.
Mi nSamplePeriod	string(s)	• read-write(null)*	The value shall be an ISO 8601 duration specifying the minimum sampling period over which average values are calculated.

Property	Type	Attributes	Notes
• *MinSupportedIo OperationLa- tenc yMicrosec- onds**	integer(us)	• read- write(null)*	The value shall be the minimum supported average IO latency in microseconds calculated over the SamplePeriod.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
Support edIOWorkloads [{ }]	array (object)	*(null)*	The value shall be a collection of supported workloads. For property details, see IOWorkload.
SupportedL inesOfService [{	array		The value shall be a collection supported IO performance service options.
@odata.id	string	*read-write*	Link to a IOPerforman ceLineOfService resource. See the Links section and the *IOPerformanc eLineOfService* schema for details.
}]			

9.5.23 LineOfService 1.1.0

9.5.23.1 Description   This service option is the abstract base class for other ClasssOfService and concrete lines of service.

9.5.23.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.24 LineOfServiceCollection

9.5.24.1 URIs   /redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/DataProt
/redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/DataSecurityLinesOfServi
/redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/DataStorageLinesOfServic
/redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/IOConnectivityLinesOfSer
/redfish/v1/StorageServices/{*StorageServiceId*}/ClassesOfService/{*ClassOfServiceId*}/IOPerformanceLinesOfSer
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService /redfish/v1/StorageServices/{*StorageServiceId*}/
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/DataSecurityLinesOfService
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/DataStorageLinesOfService
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/IOConnectivityLinesOfService
/redfish/v1/StorageServices/{*StorageServiceId*}/LinesOfService/IOPerformanceLinesOfService

9.5.24.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		The value of each member entry shall reference a LineOfService resource.
@odata.id	string	*read-only*	Link to a LineOfService resource. See the Links section and the *LineOfService* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.

Property	Type	Attributes	Notes
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.25 NVMeDomain 1.1.0

9.5.25.1 Description   Properties for the Domain.

9.5.25.2 URIs   /redfish/v1/NVMeDomains/*{NVMeDomainId}*

9.5.25.3 Properties

Property	Type	Attributes	Notes
Actions {}	object		This property shall contain the available actions for this resource.
AvailableF irmwareImages [{	array		A collection of available firmware images.
@odata.id	string	*read-only*	Link to a NV MeFirmwareImage resource. See the Links section and the *NVM eFirmwareImage* schema for details.
}]			
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
DomainMembers [{	array		The members of the domain.

Property	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Links {	object		This property shall contain links to resources that are related to but are not contained by or subordinate to this resource.
Asso ciatedDomains [{	array		This property shall contain an array of links to resources of type NVMeDomain that represent associated domains.

Property	Type	Attributes	Notes
@odata.id	string	*read-only*	Link to another NVMeDomain resource.
}]			
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.
}			
MaximumCa pacityPerEndura nceGroupBytes	integer(By)	*read-only(null)*	This property shall contain the maximum capacity per endurance group in bytes of this NVMe Domain.

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
Status {}	object		This property shall contain any status or health properties of the resource. For property details, see Status.

Property	Type	Attributes	Notes
TotalDomain CapacityBytes	integer(By)	*read-only(null)*	This property shall contain the total capacity in bytes of this NVMe Domain.
Un allocatedDomain CapacityBytes	integer(By)	*read-only(null)*	This property shall contain the total unallocated capacity in bytes of this NVMe Domain.

9.5.26 NVMeDomainCollection

9.5.26.1 URIs   /redfish/v1/NVMeDomains

9.5.26.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		The value of each member entry shall reference a NVMeDomain resource.

Property	Type	Attributes	Notes
@odata.id	string	*read-only*	Link to a NVMeDomain resource. See the Links section and the *NVMeDomain* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.

9.5.27 NVMeFirmwareImage 1.1.0

9.5.27.1 Description   NVMe Domain firmware image information.

9.5.27.2 URIs   /redfish/v1/NVMeDomains/*{DomainId}*/AvailableFirmwareImages/*{FirmwareImageId}*

9.5.27.3 Properties

Property	Type	Attributes	Notes
Actions {}	object		This property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Fi rmwareVersion	string	*r ead-only(null)*	This property shall contain the firmware version of the available NVMe firmware image.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
N VMeDeviceType	string(enum)	*read-only(null)*	This property shall specify the type of NVMe device for this NVMe firmware image. *For the possible property values, see NVMeDeviceType in Property details.*

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
Vendor	string	*r ead-only(null)*	This property shall include the name of the manufacturer or vendor associate with this NVMe firmware image.

9.5.27.4 Property details

9.5.27.4.1 NVMeDeviceType: This property shall specify the type of NVMe device for this NVMe firmware image.

string	Description
Drive	Specifies an device type of Drive, indicating a NVMe device that presents as an NVMe SSD device.

string	Description
FabricAttachArray	Specifies an NVMe device type of FabricAttachArray, indicating a NVMe device that presents an NVMe front-end that abstracts the back end storage, typically with multiple options for availability and protection.
JBOF	Specifies an device type of JBOF, indicating a NVMe device that presents as an NVMe smart enclosure for NVMe devices, typically NVMe Drives.

9.5.28 SpareResourceSet 1.0.1

9.5.28.1 Description   The values define a set of spares of a particular type.

9.5.28.2 Properties

Property	Type	Attributes	Notes
Actions *(v1.0.1+)* {}	object		The Actions property shall contain the available actions for this resource.
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Links {	object		This structure shall contain references to resources that are not contained within this resource.
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
OnHandSpares** [{	array		The type of resources in the set.

Property	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			
Replace mentSpareSets [{	array		Other spare sets that can be utilized to replenish this spare set.
@odata.id	string	*read-only*	Link to another S pareResourceSet resource.
}]			
}			
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
O nHandLocation {}	object		The location where this set of spares is kept. For property details, see Location v1.5.0).
OnLine	boolean	*re ad-write(null)*	This set shall be available online.
• *ResourceType**	string	*re ad-write(null)*	The type of resources in the set.
Ti meToProvision	string	*re ad-write(null)*	Amount of time needed to make an on-hand resource available as a spare. Pattern: - ?P(+D)?(T(+H)?(+M)?(+(.+)?S)?)?

Property	Type	Attributes	Notes
Ti meToReplenish	string	*re ad-write(null)*	Amount of time to needed replenish consumed on-hand resources. Pattern: -?P(+D)?(T(+H)?(+M)?(+(.+)?S)?)?

9.5.29 StorageGroup 1.5.0

9.5.29.1 Description   The primary purposes of the collection shall be to govern access to the storage by clients or to add service requirements for the members of the collection. Access to the collected storage by a specified set of hosts shall be made available or unavailable atomically. Requirements specified by the class of service shall be satisfied by each collected element to which they apply. The storage group may contain: block, file, or object storage; local storage system access points through which the collection is made available; and hosts, or host access points to which the collection is made available.

9.5.29.2 URIs   /redfish/v1/Storage/{*StorageId*}/StorageGroups/{*StorageGroupId*} /redfish/v1/Storage/{*StorageId*}/Volumes/{*VolumeId*}/StorageGroups/{*StorageGroupId*} /redfish/v1/StorageServices/{*StorageServiceId*}/StorageGroups/{*StorageGroupId*} /red-fish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*}/StorageGroups/{*StorageGroupId*}

9.5.29.3 Properties

Property	Type	Attributes	Notes
AccessState	string(enum)	• read-write(null)*	The value of this property shall describe the access characteristics of this storage group. All associated logical units through all aggregated ports shall share this access state. *For the possible property values, see AccessState in Property details.*
Actions {	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
#StorageGroup. ExposeVolumes** {}	object		Exposes the storage of this group via the target endpoints named in the ServerEndpointGroups to the initiator endpoints named in the Client EndpointGroups. The property VolumesAreExposed shall be set to true when this action is completed. *For more information, see the Actions section below.*
#StorageGroup.HideVolumes {}	object		Hide the storage of this group from the initiator endpoints named in the Client EndpointGroups. The property VolumesAreExposed shall be set to false when this action is completed. *For more information, see the Actions section below.*
}			

Property	Type	Attributes	Notes
Authent icationMethod *(v1.2+)*	string(enum)	• read- write(null)*	The value of this property must be what kind of authentication that the endpoints in this StorageGroup understands. *For the possible property values, see Authe nticationMethod in Property details.*

Property	Type	Attributes	Notes
ChapInfo *(v1.2+)* [{	array		The value of this property must reflect the authentication used by this specific endpoint. If this endpoint represents an initiator, and AuthenticationMethod is CHAP or MutualCHAP, the Credentials fields CHAPUsername and CHAPSecret must be used. If this endpoint represents a target endpoint and AuthenticationMethod is MutualCHAP, then Mut ualCHAPUser-name and M utualCHAPSecret must be used.
CHAPPass-word** *(v1.3+)*	string	*re ad-write(null)*	The value of this property shall be the password when CHAP authentication is specified.

Property	Type	Attributes	Notes
CHAPUser (v1.3+)	string	re ad-write(null)	The value of this property shall be the username when CHAP authentication is specified.
Initiato rCHAPPassword (v1.2+)	string	re ad-write(null)	The value of this property shall be the shared secret for Mutual (2-way)CHAP authentication.
Init iatorCHAPUser (v1.2+)	string	re ad-write(null)	If present, this property is the initiator CHAP username for Mutual (2-way) authentication. For example, with an iSCSI scenario, use the initiator iQN.
Targe tCHAPPassword (v1.3+)	string	re ad-write(null)	The value of this property shall be the CHAP Secret for 2-way CHAP authentication.

Property	Type	Attributes	Notes
T argetCHAPUser (v1.2+)	string	re ad-write(null)	The value of this property shall be the Target CHAP Username for Mutual (2-way) CHAP authentication. For example, with an iSCSI scenario, use the target iQN.
T argetPassword (v1.2+, deprecated v1.3	string	re ad-write(null)	The value of this property shall be the CHAP Secret for 2-way CHAP authentication. *Deprecated in v1.3 and later. This property is deprecated in favor of Targe tCHAPPassword.*

}]

Property	Type	Attributes	Notes
ClientE ndpointGroups [{	array		An array of references to groups of client-side endpoints that may be used to make requests to the storage exposed by this StorageGroup. If null, the implementation may allow access to the storage via any client-side endpoint. If empty, the implementation shall not allow access to the storage via any client-side endpoint.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
DHChapInfo *(v1.3+)* [{	array		The value of this property must reflect the authentication used by this specific endpoint when the authentication type is specificed as DHCHAP. If this endpoint represents an initiator, and Authe nticationMethod is DHCHAP, the Credentials fields LocalD HCHAPAuthSecret and PeerD HCHAPAuthSecret must be used.
LocalDHC HAPAuthSecret *(v1.3+)*	string	*re ad-write(null)*	This property shall be the local DHCHAP auth secret for DHCHAP authentication.

Property	Type	Attributes	Notes
PeerDHC HAPAuthSecret *(v1.3+)*	string	*re ad-write(null)*	The value of this property shall be the peer DHCHAP auth secret for DHCHAP authentication.
}]			
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Identifier {}	object		The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).
Links {	object		This property shall contain links to other resources that are related to this resource.
Child StorageGroups [{	array		An array of references to StorageGroups are incorporated into this StorageGroup.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to another StorageGroup resource.
}]			
C lassOfService{	object		The ClassOfService that all storage in this StorageGroup conforms to. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-write*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
Parent StorageGroups [{	array		An array of references to StorageGroups that incorporate this StorageGroup.
@odata.id	string	*read-only*	Link to another StorageGroup resource.
}]			
}			
MappedVolumes *(v1.1+)* [{	array		An array of mapped volumes managed by this storage group.
Acc essCapability *(v1.4+)*	string(enum)	• read-write(null)*	Each entry shall specify the storage access capability for this mapped volume. *For the possible property values, see A ccessCapability in Property details.*
Logi calUnitNumber	string	*re ad-write(null)*	If present, the value is a SCSI Logical Unit Number for the Volume.
Volume {	object		The value shall reference a mapped Volume. See the *Volume* schema for details on this property.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a Volume resource. See the Links section and the *Volume* schema for details.
}			
}]			
Members AreConsistent	boolean	*re ad-write(null)*	The value of this property shall be set to true if all members are in a consistent state. The default value for this property is false.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
ReplicaInfo {	object		This property shall describe the replication relationship between this storage group and a corresponding source storage group. See the *Stor ageReplicaInfo* schema for details on this property.
@odata.id	string	*read-only*	Link to a ReplicaInfo resource. See the Links section and the *Stor ageReplicaInfo* schema for details.
}			

Property	Type	Attributes	Notes
R eplicaTargets *(v1.1.1+)* [{	array		The value shall reference the target replicas that are sourced by this replica.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

Property	Type	Attributes	Notes
ServerE ndpointGroups [{	array		An array of references to groups of server-side endpoints that may be used to make requests to the storage exposed by this storage group. If null, the implementation may allow access to the storage via any server-side endpoint. If empty, the implementation shall not allow access to the storage via any server-side endpoint.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

Property	Type	Attributes	Notes
Status {}	object		The property shall contain the status of the StorageGroup. For property details, see Status.
Volumes [{	array		An array of references to volumes managed by this storage group.
@odata.id	string	*read-write*	Link to a Volume resource. See the Links section and the *Volume* schema for details.
}]			
Volu mesAreExposed	boolean	*re ad-write(null)*	The value of this property shall be set to true if storage volumes are exposed to the paths defined by the client and server endpoints. The default value for this property is false.

9.5.29.4 Actions

9.5.29.4.1 ExposeVolumes   Description

Exposes the storage of this group via the target endpoints named in the ServerEnd-

pointGroups to the initiator endpoints named in the ClientEndpointGroups. The property VolumesAreExposed shall be set to true when this action is completed.

Action URI: {Base URI of target resource}/Actions/StorageGroup.ExposeVolumes

Action parameters

This action takes no parameters.

9.5.29.4.2 HideVolumes   Description

Hide the storage of this group from the initiator endpoints named in the ClientEndpointGroups. The property VolumesAreExposed shall be set to false when this action is completed.

Action URI: {Base URI of target resource}/Actions/StorageGroup.HideVolumes

Action parameters

This action takes no parameters.

9.5.29.5 Property details

9.5.29.5.1 AccessCapability: Each entry shall specify the storage access capability for this mapped volume.

string	Description
Read	Endpoints are allowed to perform reads from the specified resource.
ReadWrite	Endpoints are allowed to perform reads from and writes to the specified resource.

9.5.29.5.2 AccessState: The value of this property shall describe the access characteristics of this storage group. All associated logical units through all aggregated ports shall share this access state.

string	Description
NonOptimized	This value shall indicate each endpoint is in an active and non-optimized state.
Optimized	This value shall indicate each endpoint is in an active and optimized state.
Standby	This value shall indicate each endpoint is in a standby state.
Transitioning	This value shall indicate each endpoint is transitioning to a new state.
Unavailable	This value shall indicate each endpoint is in an unavailable state.

9.5.29.5.3 AuthenticationMethod: The value of this property must be what kind of authentication that the endpoints in this StorageGroup understands.

string	Description
CHAP	iSCSI Challenge Handshake Authentication Protocol (CHAP) authentication is used. For ChapInfo, the CHAPUser and CHAPPassword properties shall be used when type CHAP is selected.
DHCHAP	Diffie-Hellman Challenge Handshake Authentication Protocol (DHCHAP) is an authentication protocol used in Fibre Channel. When MutualCHAP is selected, DHChapInfo shall be used instead of CHAPInfo, and the LocalDHCHAPAuthSecret and PeerDHCHAPAuthSecret properties shall be used.

string	Description
MutualCHAP	iSCSI Mutual Challenge Handshake Authentication Protocol (CHAP) authentication is used. For ChapInfo, the InitiatorCHAPUser, InitiatorCHAPPassword, TargetCHAPUser, and TargetCHAPPassword properties shall be used when type MutualCHAP is selected.
None	

9.5.30 StorageGroupCollection

9.5.30.1 URIs   /redfish/v1/Storage/{*StorageId*}/StorageGroups /redfish/v1/Storage/{*StorageId*}/Volumes/{*Vol*
/redfish/v1/StorageServices/{*StorageServiceId*}/StorageGroups /redfish/v1/StorageServices/{*StorageServiceId*}/

9.5.30.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		The value of each member entry shall reference a StorageGroup resource.

Property	Type	Attributes	Notes
@odata.id	string	*read-only*	Link to a StorageGroup resource. See the Links section and the *StorageGroup* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.

9.5.31 StoragePool 1.6.0

9.5.31.1 Description   A container of data storage capable of providing capacity conforming to one of its supported classes of service. The storage pool does not support IO to its data storage.

9.5.31.2 URIs   /redfish/v1/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySource*
/redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*} /redfish/v1/Storage/{*StorageId*}/StoragePools/{*S*
/redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourceId*}/ProvidingP
/redfish/v1/Storage/{*StorageId*}/Volumes/{*VolumeId*}/AllocatedPools/{*StoragePoolId*}
/redfish/v1/Storage/{*StorageId*}/Volumes/{*VolumeId*}/CapacitySources/{*CapacitySourceId*}/ProvidingPools/{*Sto*
/redfish/v1/StorageServices/{*StorageServiceId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*
/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*} /red-
fish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/AllocatedPools/{*AllocatedPoolId*}
/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourc*
/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*}/AllocatedPools/{*StoragePoolId*}
/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*}/CapacitySources/{*CapacitySourceId*}/Prov
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*Ca*
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/AllocatedPools/{*A*
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/

/redfish/v1/Systems/*{ComputerSystemId}*/Storage/*{StorageId}*/Volumes/*{VolumeId}*/AllocatedPools/*{StoragePo*
/redfish/v1/Systems/*{ComputerSystemId}*/Storage/*{StorageId}*/Volumes/*{VolumeId}*/CapacitySources/*{Capacity*

9.5.31.3 Properties

Property	Type	Attributes	Notes
Actions *(v1.3+)* {	object		The Actions property shall contain the available actions for this resource.
#StoragePool.AddDrives {}	object		This action shall be used to add a drive, or set of drives, to an underlying capacity source for the storage pool. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#StoragePool .RemoveDrives {}	object		This action shall be used to remove a drive from the StoragePool. This action is targeted at a graceful drive removal process, such as initiating a drive cleanup and data reallocation before drive removal from the pool. The implementation may impose restrictions on the number of drives removed simultaneously. *For more information, see the Actions section below.*
#Sto ragePool.SetCom pressionState {}	object		This action shall be used to set the compression state of the storage pool. This may be both a highly impactful, as well as a long running operation. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#StoragePool.SetDeduplicationState {}	object		This action shall be used to set the dedupe state of the storage pool. This may be both a highly impactful, as well as a long running operation. *For more information, see the Actions section below.*
#StoragePool.SetEncryptionState {}	object		This action shall be used to set the encryption state of the storage pool. This may be both a highly impactful, as well as a long running operation. *For more information, see the Actions section below.*
}			
A llocatedPools {	object		The value of this property shall contain a reference to the collection of storage pools allocated from this storage pool. Contains a link to a resource.

Property	Type	Attributes	Notes
@odata.id	string	*read-only*	Link to Collection of *StoragePool*. See the StoragePool schema for details.
}			
All ocatedVolumes {	object		The value of this property shall contain a reference to the collection of volumes allocated from this storage pool. Contains a link to a resource.
@odata.id	string	*read-only*	Link to Collection of *Volume*. See the Volume schema for details.
}			
B lockSizeBytes	integer(By)	*read-only(null)*	Maximum size in bytes of the blocks which form this Volume. If the block size is variable, then the maximum block size in bytes should be specified. If the block size is unknown or if a block concept is not valid (for example, with Memory), enter a 1.

Property	Type	Attributes	Notes
Capacity {}	object		The value of this property shall provide an information about the actual utilization of the capacity within this storage pool. For property details, see Capacity v1.0.0).
Ca pacitySources [{	array		Fully or partially consumed storage from a source resource. Each entry shall provide capacity allocation data from a named source resource.
@odata.id	string	*read-write*	Link to a CapacitySource resource. See the Links section and the *CapacitySource* schema for details.
}]			

Property	Type	Attributes	Notes
Cla ssesOfService {	object		This property shall contain references to all classes of service supported by this storage pool. Capacity allocated from this storage pool shall conform to one of the referenced classes of service. Contains a link to a resource.
@odata.id	string	*read-write*	Link to Collection of *LineOfService*. See the LineOfService schema for details.
}			
Compressed *(v1.3+, deprecated v1.6*	boolean	*re ad-write(null)*	This property shall contain a boolean indicator if the StoragePool is currently utilizing compression or not. *Deprecated in v1.6 and later. This property has been deprecated in favor of the IsCompressed and DefaultComp ressionBehavior properties.*

Property	Type	Attributes	Notes
Compr essionEnabled (v1.6+)	boolean	r ead-only(null)	The property shall indicate whether or not compression is enabled on the storage pool.
• *Deduplicated** (v1.3+, deprecated v1.6	boolean	re ad-write(null)	This property shall contain a boolean indicator if the StoragePool is currently utilizing deduplication or not. *Deprecated in v1.6 and later. This property has been deprecated in favor of the IsDeduplicated and Defaul tDedupeBehavior properties.*
Dedupli cationEnabled (v1.6+)	boolean	r ead-only(null)	The property shall indicate whether or not deduplication is enabled on the storage pool.

Property	Type	Attributes	Notes
DefaultC lassOfService *(v1.2+)* {	object		If present, this property shall reference the default class of service for entities allocated from this storage pool. If the C lassesOfService collection is not empty, then the value of this property shall be one of its entries. If not present, the default class of service of the containing StorageService entity shall be used. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-write*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			

Property	Type	Attributes	Notes
DefaultCompressionBehavior *(v1.6+)*	boolean	*re ad-write(null)*	If implemented, this property shall indicate the default dedupe behavior applied to the child resource (E.g., volume or storage pool) created out of the storage pool if the 'Compressed' property is not set on the create request.
DefaultDeduplicationBehavior *(v1.6+)*	boolean	*re ad-write(null)*	If implemented, this property shall indicate the default deduplication behavior applied to the child resource (E.g., volume or storage pool) created out of the storage pool if the 'Deduplicated' property is not set on the create request.

Property	Type	Attributes	Notes
DefaultEncry ptionBehavior *(v1.6+)*	boolean	*re ad-write(null)*	If implemented, this property shall indicate the default dedupe behavior applied to the child resource (E.g., volume or storage pool) created out of the storage pool if the 'Encrypted' property is not set on the create request.
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Encrypted *(v1.3+, deprecated v1.6*	boolean	*re ad-write(null)*	This property shall contain a boolean indicator if the StoragePool is currently utilizing encryption or not. *Deprecated in v1.6 and later. This property has been deprecated in favor of the IsEncrypted and DefaultEnc ryptionBehavior properties.*
Encr yptionEnabled *(v1.6+)*	boolean	*r ead-only(null)*	The property shall indicate whether or not encryption is enabled on the storage pool.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Identifier {}	object		The value identifies this resource. The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).
• *IOStatistics** (v1.2+) {}	object		The value shall represent IO statistics for this StoragePool. For property details, see IOStatistics.
Links {	object		The Links property, as described by the Redfish Specification, shall contain references to resources that are related to, but not contained by (subordinate to), this resource.

Property	Type	Attributes	Notes
Dedicat edSpareDrives *(v1.2+)* [{	array		The value of this property shall be a reference to the resources that this StoragePool is associated with and shall reference resources of type Drive. This property shall only contain references to Drive entities which are currently assigned as a dedicated spare and are able to support this StoragePool.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

Property	Type	Attributes	Notes
DefaultC lassOfService {	object		If present, this property shall reference the default class of service for entities allocated from this storage pool. If the C lassesOfService collection is not empty, then the value of this property shall be one of its entries. If not present, the default class of service of the containing StorageService entity shall be used. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-write*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.
OwningStorageResource *(v1.4+)* {	object		This shall be a pointer to the Storage resource that owns or contains this StoragePool.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			

Property	Type	Attributes	Notes
Spar eResourceSets *(v1.2+)* [{	array		Each referenced S pareResourceSet shall contain resources that may be utilized to replace the capacity provided by a failed resource having a compatible type.
@odata.id	string	*read-write*	Link to a S pareResourceSet resource. See the Links section and the *Sp areResourceSet* schema for details.
}]			
}			

Property	Type	Attributes	Notes
LowS paceWarningThre sholdPercents [ ]	array(%) (integer, null)	*read-write*	Each time the following value is less than one of the values in the array the LOW_SPACE_TH RESH- OLD_WARNING event shall be triggered: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(Consum edBytes))/SUM(A llocatedBytes).
MaxB lockSizeBytes *(v1.1.1+)*	integer(By)	*read-only(null)*	If present, the value is the maximum block size of an allocated resource. If the block size is unknown or if a block concept is not valid (for example, with Memory), this property shall be NULL.

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.
NVMeEn-duranceGroupProperties *(v1.4+)* {	object	*(null)*	This property shall contain properties to use when StoragePool is used to describe an NVMe Endurance Group.
EndGrpLifetime *(v1.4+)* {	object	*(null)*	This property shall contain any Endurance Group Lifetime properties.

Property	Type	Attributes	Notes
DataUnit-sRead (v1.4+)	integer	r ead-only(null)	The property shall contain the total number of data units read from this endurance group. This value does not include controller reads due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.

Property	Type	Attributes	Notes
Dat aUnitsWritten *(v1.4+)*	integer	r ead-only(null)	The property shall contain the total number of data units written from this endurance group. This value does not include controller writes due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.

Property	Type	Attributes	Notes
Endu ranceEstimate *(v1.4+)*	integer	*r ead-only(null)*	This property shall contain an estimate of the total number of data bytes that may be written to the Endurance Group over the lifetime of the Endurance Group assuming a write amplication of 1. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates endurance estimates are unsupported.
E rrorInformation LogEntryCount *(v1.4+)*	integer	*r ead-only(null)*	This property shall contain the number of error information log entries over the life of the controller for the endurance group.

Property	Type	Attributes	Notes
HostRea dCommandCount *(v1.4+)*	integer	*r ead-only(null)*	This property shall contain the number of read commands completed by all controllers in the NVM subsystem for the Endurance Group. For the NVM command set, the is the number of compare commands and read commands.
HostWrit eCommandCount *(v1.4+)*	integer	*r ead-only(null)*	This property shall contain the number of write commands completed by all controllers in the NVM subsystem for the Endurance Group. For the NVM command set, the is the number of compare commands and write commands.

Property	Type	Attributes	Notes
Med iaAndDataIntegr ityErrorCount (v1.4+)	integer	r ead-only(null)	This property shall contain the number of occurences where the controller detected an unrecovered data integrity error for the Endurance Group. Errors such as uncorrectable ECC, CRC checksum failure, or LBA tag mismatch are included in this field.

Property	Type	Attributes	Notes
Medi aUnitsWritten *(v1.4+)*	integer	r ead-only(null)	The property shall contain the total number of data units written from this endurance group. This value includes host and controller writes due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.

Property	Type	Attributes	Notes
PercentUsed (v1.4+)	integer	*r ead-only(null)*	This property shall contain a vendor-specific estimate of the percent life used for the endurance group based on the actual usage and the manufacturer prediction of NVM life. A value of 100 indicates that the estimated endurance of the NVM in the Endurance Group has been consumed, but may not indicate an NVM failure. According to the NVMe and JEDEC specs, the value is allowed to exceed 100. Percentages greater than 254 shall be represented as 255.
}			
P redictedMediaLi feLeftPercent (v1.4+)	number(%)	*read-only(null)*	This property shall contain an indicator of the percentage of life remaining in the drive's media.

Property	Type	Attributes	Notes
}			
N VMeProperties (v1.6+) {	object	(null)	The property shall indicate the type of storage pool.
NVMePoolType** (v1.6+)	string(enum)	read-only(null)	This property shall indicate whether the StoragePool is used as an EnduranceGroup or an NVMSet. *For the possible property values, see NVMePoolType in Property details.*
}			
NVMe SetProperties (v1.4+) {	object	(null)	This property shall contain properties to use when StoragePool is used to describe an NVMe Set.
EnduranceGr oupIdentifier (v1.4+)	string	r ead-only(null)	This property shall contain a 16-bit hex value that contains the endurance group identifier. The endurance group identifier is unique within a subsystem. Reserved values include 0. Pattern: ^0[x X](([a-fA-F]

Property	Type	Attributes	Notes
OptimalW riteSizeBytes *(v1.4+)*	integer(By)	*read-only(null)*	This property shall contain the Optimal Write Size in Bytes for this NVMe Set.
Ra ndom4kReadTypic alNanoSeconds *(v1.4+)*	integer	*r ead-only(null)*	This property shall contain the typical time to complete a 4k read in 100 nano-second units when the NVM Set is in a Predictable Latency Mode Deterministic Window and there is 1 outstanding command per NVM Set.
SetIdentifier *(v1.4+)*	string	*r ead-only(null)*	This property shall contain a 16-bit hex value that contains the NVMe Set group identifier. The NVM Set identifier is unique within a subsystem. Reserved values include 0. Pattern: ^0[x X](([a-fA-F]
Unalloca ted- NVMNamespace CapacityBytes *(v1.4+)*	integer(By)	*read-only(null)*	This property shall contain the unallocated capacity of the NVMe Set in bytes.

Property	Type	Attributes	Notes
}			
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
PoolType *(v1.6+)* [ ]	array (string(enum))	*read-only(null)*	The property shall indicate the type of storage pool. *For the possible property values, see PoolType in Property details.*
Re coverableCapaci tySourceCount *(v1.2+)*	integer	*re ad-write(null)*	The value is the number of available capacity source resources currently available in the event that an equivalent capacity source resource fails.

Property	Type	Attributes	Notes
RemainingCa pacityPercent *(v1.1+)*	integer	*r ead-only(null)*	If present, this value shall return {[(SUM(AllocatedBytes) - S UM(ConsumedByte s)]/SUM(Alloca tedBytes)}*100 represented as an integer value.
Status {}	object		The property shall contain the status of the StoragePool. For property details, see Status.
S upportedProvisi oningPolicies *(v1.3+)* [ ]	array (string(enum))	• read-write(null)*	This collection shall specify all supported storage allocation policies for the Storage Pool. *For the possible property values, see SupportedProvi sioningPolicies in Property details.*
Suppo rtedRAIDTypes *(v1.3+)* [ ]	array (string(enum))	*read-only(null)*	This collection shall contain all the RAIDType values supported by the storage pool. *For the possible property values, see Sup portedRAIDTypes in Property details.*

9.5.31.4 Actions

9.5.31.4.1 AddDrives   Description

This action shall be used to add a drive, or set of drives, to an underlying capacity source for the storage pool.

Action URI: {Base URI of target resource}/Actions/StoragePool.AddDrives

Action parameters

Parameter Name	Type	Attributes	Notes
C apacitySource {	object	*optional*	This parameter shall contain the target capacity source for the drive(s). This property does not need to be specified if the storage pool only contains one capacity source, or if the implementation is capable of automatically selecting the appropriate capacity source. See the *CapacitySource* schema for details on this property.

Parameter Name	Type	Attributes	Notes
@odata.id	string	*read-only*	Link to a CapacitySource resource. See the Links section and the *CapacitySource* schema for details.
}			
Drives [{	array	*required*	This parameter shall contain the Uri to the existing drive or drives to be added to a capacity source of the storage pool. The implementation may impose restrictions on the number of drives added simultaneously.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

9.5.31.4.2 RemoveDrives   Description

This action shall be used to remove a drive from the StoragePool. This action is targeted at a graceful drive removal process, such as initiating a drive cleanup and data reallocation before drive removal from the pool. The implementation may impose restrictions on the number of drives removed simultaneously.

Action URI: {Base URI of target resource}/Actions/StoragePool.RemoveDrives

Action parameters

Parameter Name	Type	Attributes	Notes
Drives [{	array	*required*	This parameter shall contain the Uri to the drive or drives to be removed from the underlying capacity source.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

9.5.31.4.3 SetCompressionState   Description

This action shall be used to set the compression state of the storage pool. This may be both a highly impactful, as well as a long running operation.

Action URI: {Base URI of target resource}/Actions/StoragePool.SetCompressionState

Action parameters

Parameter Name	Type	Attributes	Notes
Enable	boolean	*required*	This property shall indicate the desired compression state of the storage pool.

9.5.31.4.4 SetDeduplicationState   Description

This action shall be used to set the dedupe state of the storage pool. This may be both a highly impactful, as well as a long running operation.

Action URI: {Base URI of target resource}/Actions/StoragePool.SetDeduplicationState

Action parameters

Parameter Name	Type	Attributes	Notes
Enable	boolean	*required*	This property shall indicate the desired deduplication state of the storage pool.

9.5.31.4.5 SetEncryptionState   Description

This action shall be used to set the encryption state of the storage pool. This may be both a highly impactful, as well as a long running operation.

Action URI: {Base URI of target resource}/Actions/StoragePool.SetEncryptionState

Action parameters

Parameter Name	Type	Attributes	Notes
Enable	boolean	*required*	This property shall indicate the desired encryption state of the storage pool.

9.5.31.5 Property details

9.5.31.5.1 NVMePoolType: This property shall indicate whether the StoragePool is used as an EnduranceGroup or an NVMSet.

string	Description
EnduranceGroup	This type shall be used to specify a pool of type EnduranceGroup, used by NVMe devices.
NVMSet	This type shall be used to specify a pool of type NVMSet, used by NVMe devices.

9.5.31.5.2 PoolType: The property shall indicate the type of storage pool.

string	Description
Block	This type shall be used to specify a pool of type block. This is used when the pool serves block storage.
File	This type shall be used to specify a pool of type file. This setting is used when the pool serves file storage.
Object	This type shall be used to specify a pool of type object.

string	Description
Pool	This type shall be used to specify a pool of type pool. This setting is used to indicate a 'pool of pools' hierarchy.

9.5.31.5.3 SupportedProvisioningPolicies: This collection shall specify all supported storage allocation policies for the Storage Pool.

string	Description
Fixed	This enumeration literal specifies storage shall be fully allocated.
Thin	This enumeration literal specifies storage may be over allocated.

9.5.31.5.4 SupportedRAIDTypes: This collection shall contain all the RAIDType values supported by the storage pool.

string	Description
None	A placement policy with no redundancy at the device level.
RAID0	A placement policy where consecutive logical blocks of data are uniformly distributed across a set of independent storage devices without offering any form of redundancy. This is commonly referred to as data striping. This form of RAID will encounter data loss with the failure of any storage device in the set.

string	Description
RAID00	A placement policy that creates a RAID 0 stripe set over two or more RAID 0 sets. This is commonly referred to as RAID 0+0. This form of data layout is not fault tolerant; if any storage device fails there will be data loss.
RAID01	A data placement policy that creates a mirrored device (RAID 1) over a set of striped devices (RAID 0). This is commonly referred to as RAID 0+1 or RAID 0/1. Data stored using this form of RAID is able to survive a single RAID 0 data set failure without data loss.
RAID1	A placement policy where each logical block of data is stored on more than one independent storage device. This is commonly referred to as mirroring. Data stored using this form of RAID is able to survive a single storage device failure without data loss.
RAID10	A placement policy that creates a striped device (RAID 0) over a set of mirrored devices (RAID 1). This is commonly referred to as RAID 1/0. Data stored using this form of RAID is able to survive storage device failures in each RAID 1 set without data loss.
RAID10E	A placement policy that uses a RAID 0 stripe set over two or more RAID 10 sets. This is commonly referred to as Enhanced RAID 10. Data stored using this form of RAID is able to survive a single device failure within each nested RAID 1 set without data loss.

string	Description
RAID10Triple	A placement policy that uses a striped device (RAID 0) over a set of triple mirrored devices (RAID 1Triple). This form of RAID can survive up to two failures in each triple mirror set without data loss.
RAID1E	A placement policy that uses a form of mirroring implemented over a set of independent storage devices where logical blocks are duplicated on a pair of independent storage devices so that data is uniformly distributed across the storage devices. This is commonly referred to as RAID 1 Enhanced. Data stored using this form of RAID is able to survive a single storage device failure without data loss.
RAID1Triple	A placement policy where each logical block of data is mirrored three times across a set of three independent storage devices. This is commonly referred to as three-way mirroring. This form of RAID can survive two device failures without data loss.

string	Description
RAID3	A placement policy using parity-based protection where logical bytes of data are uniformly distributed across a set of independent storage devices and where the parity is stored on a dedicated independent storage device. Data stored using this form of RAID is able to survive a single storage device failure without data loss. If the storage devices use rotating media, they are assumed to be rotationally synchronized, and the data stripe size should be no larger than the exported block size.
RAID4	A placement policy using parity-based protection where logical blocks of data are uniformly distributed across a set of independent storage devices and where the parity is stored on a dedicated independent storage device. Data stored using this form of RAID is able to survive a single storage device failure without data loss.
RAID5	A placement policy using parity-based protection for storing stripes of 'n' logical blocks of data and one logical block of parity across a set of 'n+1' independent storage devices where the parity and data blocks are interleaved across the storage devices. Data stored using this form of RAID is able to survive a single storage device failure without data loss.

string	Description
RAID50	A placement policy that uses a RAID 0 stripe set over two or more RAID 5 sets of independent storage devices. Data stored using this form of RAID is able to survive a single storage device failure within each RAID 5 set without data loss.
RAID6	A placement policy using parity-based protection for storing stripes of 'n' logical blocks of data and two logical blocks of independent parity across a set of 'n+2' independent storage devices where the parity and data blocks are interleaved across the storage devices. Data stored using this form of RAID is able to survive any two independent storage device failures without data loss.
RAID60	A placement policy that uses a RAID 0 stripe set over two or more RAID 6 sets of independent storage devices. Data stored using this form of RAID is able to survive two device failures within each RAID 6 set without data loss.
RAID6TP	A placement policy that uses parity-based protection for storing stripes of 'n' logical blocks of data and three logical blocks of independent parity across a set of 'n+3' independent storage devices where the parity and data blocks are interleaved across the storage devices. This is commonly referred to as Triple Parity RAID. Data stored using this form of RAID is able to survive any three independent storage device failures without data loss.

9.5.32 StoragePoolCollection

9.5.32.1 URIs   /redfish/v1/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySource*
/redfish/v1/Storage/{*StorageId*}/StoragePools /redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/Al
/redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourceId*}/ProvidingP
/redfish/v1/Storage/{*StorageId*}/Volumes/{*VolumeId*}/AllocatedPools   /red-
fish/v1/Storage/{*StorageId*}/Volumes/{*VolumeId*}/CapacitySources/{*CapacitySourceId*}/ProvidingPools
/redfish/v1/StorageServices/{*StorageServiceId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*
/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools /redfish/v1/StorageServices/{*StorageServiceId*}/S
/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourc*
/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*}/AllocatedPools
/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*}/CapacitySources/{*CapacitySourceId*}/Prov
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*Ca*
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools   /red-
fish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/AllocatedPools
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/Volumes/{*VolumeId*}/AllocatedPools
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/Volumes/{*VolumeId*}/CapacitySources/{*Capacity*

9.5.32.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		The value of each member entry shall reference a StoragePool resource.

Property	Type	Attributes	Notes
@odata.id	string	*read-only*	Link to a StoragePool resource. See the Links section and the *StoragePool* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.

9.5.33 StorageReplicaInfo 1.3.0

9.5.33.1 Description   This entity shall define the characteristics of a replica.

9.5.33.2 Properties

Property	Type	Attributes	Notes
Actions *(v1.2+)* {}	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.34 StorageService 1.5.0

9.5.34.1 Description   Collection of resources that the system can make available to one or more host systems. The collection can contain: block, file, or object storage; local system access points through which the collection is made available; hosts, or host access points to which the collection is made available.

9.5.34.2 URIs   /redfish/v1/StorageServices/*{StorageServiceId}* /redfish/v1/Systems/*{ComputerSystemId}*/Stor

9.5.34.3 Properties

Property	Type	Attributes	Notes
Actions {	object		The Actions property shall contain the available actions for this resource.

Property	Type	Attributes	Notes
#Sto rageService.Set EncryptionKey {} }	object		This defines the name of the custom action supported on this resource. *For more information, see the Actions section below.*
Cla ssesOfService {	object		The value of each entry in the array shall reference a ClassOfService supported by this service. Contains a link to a resource.
@odata.id }	string	*read-write*	Link to Collection of *LineOfService*. See the LineOfService schema for details.
ClientE ndpointGroups {}	object		The value of each entry in the array shall reference an EndpointGroup.
Cons istencyGroups *(v1.3+)* {	object		The value of each entry in the array shall reference a Co nsistencyGroup. Contains a link to a resource.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to Collection of *Con sistencyGroup*. See the C onsistencyGroup schema for details.
}			
D ataProtectionLo SCapabilities *(v1.2+)* {	object		The value shall reference the data protection capabilities of this service. See the *DataProtectionL oSCapabilities* schema for details on this property.
@odata.id	string	*read-write*	Link to a DataProtection LoSCapabilities resource. See the Links section and the *DataProtectionL oSCapabilities* schema for details.
}			
• *DataSecurityLo SCapabili- ties** (v1.2+)* {	object		The value shall reference the data security capabilities of this service. See the *DataSecurityL oSCapabilities* schema for details on this property.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a DataSecurity LoSCapabilities resource. See the Links section and the *DataSecurityL oSCapabilities* schema for details.
}			
DataStorageLo SCapabilities *(v1.2+)* {	object		The value shall reference the data storage capabilities of this service. See the *DataStorageL oSCapabilities* schema for details on this property.
@odata.id	string	*read-write*	Link to a DataStorage LoSCapabilities resource. See the Links section and the *DataStorageL oSCapabilities* schema for details.
}			

Property	Type	Attributes	Notes
DefaultC lassOfService *(v1.2+)* {	object		If present, this property shall reference the default class of service for entities allocated by this storage service. This default may be overridden by the Defaul tClassOfService property values within contained StoragePools. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-write*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.

Property	Type	Attributes	Notes
Drives {}	object		A collection that indicates all the drives managed by this storage service.
E ndpointGroups {}	object		The value of each entry in the array shall reference an EndpointGroup.
Endpoints {}	object		The value of each entry in the array shall reference an Endpoint managed by this service.
FileSystems {	object		An array of references to FileSystems managed by this storage service. Contains a link to a resource.
@odata.id	string	*read-write*	Link to Collection of *FileSystem*. See the FileSystem schema for details.
}			

Property	Type	Attributes	Notes
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specification-described requirements.
Identifier {}	object		The value identifies this resource. The value shall be unique within the managed ecosystem. For property details, see Identifier v1.12.0).
I OConnectivityLo SCapabilities *(v1.2+)* {	object		The value shall reference the IO connectivity capabilities of this service. See the *IOConnectivityL oSCapabilities* schema for details on this property.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a IOConnectivity LoSCapabilities resource. See the Links section and the *IOConnectivityL oSCapabilities* schema for details.
}			
IOPerformanceLo SCapabilities *(v1.2+)* {	object		The value shall reference the IO performance capabilities of this service. See the *IOPerformanceL oSCapabilities* schema for details on this property.
@odata.id	string	*read-write*	Link to a IOPerformance LoSCapabilities resource. See the Links section and the *IOPerformanceL oSCapabilities* schema for details.
}			

Property	Type	Attributes	Notes
• *IOStatistics** (v1.2+) {}	object		The value shall represent IO statistics for this StorageService. For property details, see IOStatistics.
L inesOfService (v1.4+) [{	array		The value of each entry shall reference a LineOfService collection defined for this service.
@odata.id	string	read-write	Link to Collection of LineOfService. See the LineOfService schema for details.
}]			
Links {	object		This property shall contain links to other resources that are related to this resource.
D ataProtectionLo SCapabilities {	object		The value shall reference the data protection capabilities of this service. See the DataProtectionL oSCapabilities schema for details on this property.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a DataProtection LoSCapabilities resource. See the Links section and the *DataProtectionLoSCapabilities* schema for details.
}			
DataSecurityLoSCapabilities** {	object		The value shall reference the data security capabilities of this service. See the *DataSecurityLoSCapabilities* schema for details on this property.
@odata.id	string	*read-write*	Link to a DataSecurity LoSCapabilities resource. See the Links section and the *DataSecurityLoSCapabilities* schema for details.
}			

Property	Type	Attributes	Notes
DataStorageLoSCapabilities {	object		The value shall reference the data storage capabilities of this service. See the *DataStorageLoSCapabilities* schema for details on this property.
@odata.id	string	*read-write*	Link to a DataStorage LoSCapabilities resource. See the Links section and the *DataStorageLoSCapabilities* schema for details.
}			
DefaultClassOfService {	object		If present, this property shall reference the default class of service for entities allocated by this storage service. This default may be overridden by the DefaultClassOfService property values within contained StoragePools. See the *ClassOfService* schema for details on this property.

Property	Type	Attributes	Notes
@odata.id	string	*read-write*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			
HostingSystem {}	object		The value shall reference the ComputerSystem or St orageController that hosts this service.
I OConnectivityLo SCapabilities {	object		The value shall reference the IO connectivity capabilities of this service. See the *IOConnectivityL oSCapabilities* schema for details on this property.
@odata.id	string	*read-write*	Link to a IOConnectivity LoSCapabilities resource. See the Links section and the *IOConnectivityL oSCapabilities* schema for details.
}			

Property	Type	Attributes	Notes
IOPerfor-manceLo SCapabilities {	object		The value shall reference the IO performance capabilities of this service. See the *IOPerformanceL oSCapabilities* schema for details on this property.
@odata.id	string	*read-write*	Link to a IOPerformance LoSCapabilities resource. See the Links section and the *IOPerformanceL oSCapabilities* schema for details.
}			
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
}			

Property	Type	Attributes	Notes
Name	string	*read-only required*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.
Redundancy [{	array		This collection shall contain the redundancy information for the storage subsystem.

Property	Type	Attributes	Notes
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			
ServerE ndpointGroups {}	object		The value of each entry in the array shall reference a EndpointGroup.
Spar eResourceSets *(v1.2+)* [{	array		Each contained S pareResourceSet shall contain resources that may be utilized to replace the capacity provided by a failed resource having a compatible type.
@odata.id	string	*read-write*	Link to a S pareResourceSet resource. See the Links section and the *Sp areResourceSet* schema for details.
}]			

Property	Type	Attributes	Notes
Status {}	object		The property shall contain the status of the StorageService. For property details, see Status.
StorageGroups {	object		The value of each entry in the array shall reference a StorageGroup. Contains a link to a resource.
@odata.id	string	*read-only*	Link to Collection of *StorageGroup*. See the StorageGroup schema for details.
}			
• *StoragePools** {	object		An array of references to StoragePools. Contains a link to a resource.
@odata.id	string	*read-only*	Link to Collection of *StoragePool*. See the StoragePool schema for details.
}			

Property	Type	Attributes	Notes
Stor ageSubsystems *(v1.0.1+)* {}	object		The value shall be a link to a collection of type St orageCollection having members that represent storage subsystems managed by this storage service.
Volumes {	object		An array of references to Volumes managed by this storage service. Contains a link to a resource.
@odata.id	string	*read-write*	Link to Collection of *Volume*. See the Volume schema for details.
}			

9.5.34.4 Actions

9.5.34.4.1 SetEncryptionKey   Description

This defines the name of the custom action supported on this resource.

Action URI: {Base URI of target resource}/Actions/StorageService.SetEncryptionKey

Action parameters

Parameter Name	Type	Attributes	Notes
EncryptionKey	string	*optional*	This defines the property name for the action.

9.5.34.5 Property details

9.5.34.5.1 idRef:

@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.

9.5.35 StorageServiceCollection

9.5.35.1 URIs   /redfish/v1/StorageServices /redfish/v1/Systems/*{ComputerSystemId}*/StorageServices

9.5.35.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		The value of each member entry shall reference a StorageService resource.
@odata.id	string	*read-only*	Link to a StorageService resource. See the Links section and the *StorageService* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.

Property	Type	Attributes	Notes
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

9.5.36 StorageSystemCollection

9.5.36.1 URIs   /redfish/v1/StorageSystems /redfish/v1/Systems

9.5.36.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		The value of each member entry shall reference a ComputerSystem resource that shall have a HostingRoles entry with a value of ' StorageServer'.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

Property	Type	Attributes	Notes
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specification-described requirements. For property details, see Oem.

9.5.37 Volume 1.6.2

9.5.37.1 Description   This resource shall be used to represent a volume, virtual disk, logical disk, LUN, or other logical storage for a Redfish implementation.

9.5.37.2 URIs   /redfish/v1/CompositionService/ResourceBlocks/{*ResourceBlockId*}/Storage/{*StorageId*}/Volu
/redfish/v1/CompositionService/ResourceBlocks/{*ResourceBlockId*}/Systems/{*ComputerSystemId*}/Storage/{*St*
/redfish/v1/ResourceBlocks/{*ResourceBlockId*}/Storage/{*StorageId*}/Volumes/{*VolumeId*}
/redfish/v1/ResourceBlocks/{*ResourceBlockId*}/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/Volumes/{*Vo*
/redfish/v1/Storage/{*StorageId*}/ConsistencyGroups/{*ConsistencyGroupId*}/Volumes/{*VolumeId*}
/redfish/v1/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*}/ProvidingVolu
/redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/AllocatedVolumes/{*VolumeId*}
/redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourceId*}/ProvidingV
/redfish/v1/Storage/{*StorageId*}/Volumes/{*VolumeId*} /redfish/v1/StorageServices/{*StorageServiceId*}/Consisten
/redfish/v1/StorageServices/{*StorageServiceId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*
/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/AllocatedVolumes/{*VolumeId*}
/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourc*
/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes/{*VolumeId*} /redfish/v1/StorageServices/{*StorageServi*
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/ConsistencyGroups/{*ConsistencyGroupId*}/Volum
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*Ca*
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/AllocatedVolumes

/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/
/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/Volumes/{*VolumeId*}

9.5.37.3 Properties

Property	Type	Attributes	Notes
Acces sCapabilities *(v1.1+)* [ ]	array (string(enum))	• read-write(null)*	Each entry shall specify a current storage access capability. *For the possible property values, see Acc essCapabilities in Property details.*
Actions {	object		The Actions property shall contain the available actions for this resource.
#Volume.Assign ReplicaTarget** *(v1.4+)* {}	object		This action shall be used to establish a replication relationship by assigning an existing volume to serve as a target replica for an existing source volume. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#Volume.Cha ngeRAIDLayout *(v1.5+)* {}	object		This action shall request the system to change the RAID layout of the volume. Depending on the combination of the submitted parameters, this could be changing the RAID type, changing the span count, changing the number of drives used by the volume, or another configuration change supported by the system. Note that usage of this action while online may potentially cause data loss if the available capacity is reduced. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#Volume.Che ckConsistency {}	object		This defines the name of the custom action supported on this resource. *For more information, see the Actions section below.*
#Volume.Create ReplicaTarget** *(v1.4+)* {}	object		This action shall be used to create a new volume resource to provide expanded data protection through a replica relationship with the specified source volume. *For more information, see the Actions section below.*
#Volum e.ForceEnable *(v1.5+)* {}	object		This action shall request the system to force the volume to enabled state regardless of data loss scenarios. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#Volume.Initialize *(v1.5+)* {}	object		This defines the name of the custom action supported on this resource. If InitializeMethod is not specified in the request body, but the property InitializeMethod is specified, the property InitializeMethod value should be used. If neither is specified, the InitializeMethod should be Foreground. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#Volu me.RemoveReplic aRelationship *(v1.4+)* {}	object		This action shall be used to disable data synchronization between a source and target volume, remove the replication relationship, and optionally delete the target volume. *For more information, see the Actions section below.*
#Volume.Resu meReplication *(v1.4+)* {}	object		This action shall be used to resume the active data synchronization between a source and target volume, without otherwise altering the replication relationship. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
#Volume.ReverseReplicationRelationship (v1.4+) {}	object		This action shall be used to reverse the replication relationship between a source and target volume. *For more information, see the Actions section below.*
#Volume.SplitReplication (v1.4+) {}	object		This action shall be used to split the replication relationship and suspend data synchronization between a source and target volume. *For more information, see the Actions section below.*
#Volume.SuspendReplication (v1.4+) {}	object		This action shall be used to suspend active data synchronization between a source and target volume, without otherwise altering the replication relationship. *For more information, see the Actions section below.*

Property	Type	Attributes	Notes
}			
A llocatedPools *(v1.1+)* {	object		The value of this property shall contain references to all storage pools allocated from this volume. Contains a link to a resource.
@odata.id	string	*read-only*	Link to Collection of *StoragePool*. See the StoragePool schema for details.
}			
B lockSizeBytes	integer(By)	*read-only(null)*	This property shall contain size of the smallest addressable unit of the associated volume.
Capacity *(v1.1+)* {}	object		Information about the utilization of capacity allocated to this storage volume. For property details, see Capacity v1.0.0).
CapacityBytes	integer(By)	• read-write(null)*	This property shall contain the size in bytes of the associated volume.

Property	Type	Attributes	Notes
Ca pacitySources *(v1.1+)* [{	array		Fully or partially consumed storage from a source resource. Each entry provides capacity allocation information from a named source resource.
@odata.id	string	*read-write*	Link to a CapacitySource resource. See the Links section and the *CapacitySource* schema for details.
}]			
Compressed *(v1.4+)*	boolean	*re ad-write(null)*	This property shall contain a boolean indicator if the Volume is currently utilizing compression or not.
• *Deduplicated** *(v1.4+)*	boolean	*re ad-write(null)*	This property shall contain a boolean indicator if the Volume is currently utilizing deduplication or not.

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
DisplayName *(v1.4+)*	string	*re ad-write(null)*	This property shall contain a us er-configurable string to name the volume.
Encrypted	boolean	*re ad-write(null)*	This property shall contain a boolean indicator if the Volume is currently utilizing encryption or not.
En cryptionTypes [ ]	array (string(enum))	*read-write*	This property shall contain the types of encryption used by this Volume. *For the possible property values, see EncryptionTypes in Property details.*

Property	Type	Attributes	Notes
Id	string	*read-only required*	This property represents an identifier for the resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Identifiers [{ }]	array (object)		This property shall contain a list of all known durable names for the associated volume. For property details, see Identifier v1.12.0).

Property	Type	Attributes	Notes
Ini tializeMethod (v1.6+)	string(enum)	*read-only(null)*	This property shall indicate the initialization method used for this volume. If I nitializeMethod is not specified, the I nitializeMethod should be Foreground. This value reflects the most recently used Initialization Method, and may be changed using the Initialize Action. *For the possible property values, see I nitializeMethod in Property details.*
IOPe rfModeEnabled (v1.5+)	boolean	*re ad-write(null)*	This property shall indicate whether IO performance mode is enabled for the volume.
• *IOStatistics** (v1.2+) {}	object		The value shall represent IO statistics for this volume. For property details, see IOStatistics.

Property	Type	Attributes	Notes
Links {	object		The Links property, as described by the Redfish Specification, shall contain references to resources that are related to, but not contained by (subordinate to), this resource.
Cac heDataVolumes *(v1.6+)* [{	array		This shall be a pointer to the cache data volumes this volume serves as a cache volume. The corresponding VolumeUsage property shall be set to CacheOnly when this property is used.
@odata.id	string	*read-only*	Link to another Volume resource.
}]			

Property	Type	Attributes	Notes
Cach eVolumeSource *(v1.6+)* {	object	*(null)*	This shall be a pointer to the cache volume source for this volume. The corresponding VolumeUsage property shall be set to Data when this property is used.
@odata.id	string	*read-only*	Link to another Volume resource.
}			
C lassOfService *(v1.1+)* {	object		This property shall contain a reference to the ClassOfService that this storage volume conforms to. See the *ClassOfService* schema for details on this property.
@odata.id	string	*read-only*	Link to a ClassOfService resource. See the Links section and the *ClassOfService* schema for details.
}			

Property	Type	Attributes	Notes
Cl ientEndpoints *(v1.4+)* [{	array		The value of this property shall be references to the client Endpoints this volume is associated with.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			
Cons istencyGroups *(v1.4+)* [{	array		The value of this property shall be references to the Co nsistencyGroups this volume is associated with.
@odata.id	string	*read-only*	Link to a C onsistencyGroup resource. See the Links section and the *Co nsistencyGroup* schema for details.
}]			

Property	Type	Attributes	Notes
Dedicat edSpareDrives *(v1.2+)* [{	array		The value of this property shall be a reference to the resources that this volume is associated with and shall reference resources of type Drive. This property shall only contain references to Drive entities which are currently assigned as a dedicated spare and are able to support this Volume.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

Property	Type	Attributes	Notes
Drives [{	array		The value of this property shall be a reference to the resources that this volume is associated with and shall reference resources of type Drive. This property shall only contain references to Drive entities which are currently members of the Volume, not hot spare Drives which are not currently a member of the volume.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			

Property	Type	Attributes	Notes
Jo urnalingMedia *(v1.5+)* {	object	*(null)*	This shall be a pointer to the journaling media used for this Volume to address the write hole issue. Valid when WriteHoleP rotectionPolicy property is set to 'Journaling'.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
OwningSt orageResource *(v1.5+)* {	object		This shall be a pointer to the Storage resource that owns or contains this volume.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}			
OwningS torageService *(v1.4+)* {	object		This shall be a pointer to the StorageService that owns or contains this volume. See the *StorageService* schema for details on this property.
@odata.id	string	*read-only*	Link to a StorageService resource. See the Links section and the *StorageService* schema for details.
}			

Property	Type	Attributes	Notes
Se rverEndpoints *(v1.4+)* [{	array		The value of this property shall be references to the server Endpoints this volume is associated with.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			
Spar eResourceSets *(v1.3+)* [{	array		Each referenced S pareResourceSet shall contain resources that may be utilized to replace the capacity provided by a failed resource having a compatible type.
@odata.id	string	*read-write*	Link to a S pareResourceSet resource. See the Links section and the *Sp areResourceSet* schema for details.
}]			

Property	Type	Attributes	Notes
StorageGroups *(v1.4+)* [{	array		The value of this property shall be references to the StorageGroups this volume is associated with.
@odata.id	string	*read-only*	Link to a StorageGroup resource. See the Links section and the *StorageGroup* schema for details.
}] }			
Logi calUnitNumber *(v1.4+)*	integer	*r ead-only(null)*	This property shall contain host-visible Lo gicalUnitNumber assigned to this Volume. This property shall only be used when in a single connect configuration and no StorageGroup configuration is used.

Property	Type	Attributes	Notes
LowS paceWarningThre sholdPercents (v1.1+) [ ]	array(%) (integer, null)	*read-write*	Each time the following value is less than one of the values in the array the LOW_SPACE_TH RESH- OLD_WARNING event shall be triggered: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(Consum edBytes))/SUM(A llocatedBytes).
• *Manufacturer*** (v1.1+)	string	*r ead-only(null)*	This property shall contain a value that represents the manufacturer or implementer of the storage volume.
MaxB lockSizeBytes (v1.1+)	integer(By)	*read-only(null)*	This property shall contain size of the largest addressable unit of this storage volume.

Property	Type	Attributes	Notes
M ediaSpanCount (v1.4+)	integer	r ead-only(null)	This property shall indicate the number of media elements used per span in the secondary RAID for a hierarchical RAID type.
Model (v1.1+)	string	r ead-only(null)	The value is assigned by the manufacturer and shall represents a specific storage volume implementation.
Name	string	read-only required	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.

Property	Type	Attributes	Notes
NVMeNamesp aceProperties *(v1.5+)* {	object	*(null)*	This property shall contain properties to use when Volume is used to describe an NVMe Namespace.
For mattedLBASize *(v1.5+)*	string	*r ead-only(null)*	This property shall contain the LBA data size and metadata size combination that the namespace has been formatted with. This is a 4-bit data structure.
IsShareable *(v1.5+)*	boolean	*re ad-write(null)*	This property shall indicate whether the namespace is shareable.
Metad ataTransferredA tEndOfDataLBA *(v1.5+)*	boolean	*r ead-only(null)*	This property shall indicate whether or not the metadata is transferred at the end of the LBA creating an extended data LBA.
Name spaceFeatures *(v1.5+)* {	object	*(null)*	This property shall contain a set of Namespace Features.

Property	Type	Attributes	Notes
S upportsAtomicTr ansactionSize *(v1.5+)*	boolean	*r ead-only(null)*	This property shall indicate whether or not the NVM fields for Namespace preferred write granularity (NPWG), write alignment (NPWA), deallocate granularity (NPDG), deallocate alignment (NPDA) and optimal write size (NOWS) are defined for this namespace and should be used by the host for I/O optimization.
SupportsD eallocatedOrUnw rittenLBError *(v1.5+)*	boolean	*r ead-only(null)*	This property shall indicate that the controller supports deallocated or unwritten logical block error for this namespace. .

Property	Type	Attributes	Notes
Support-sIOPer formanceHints *(v1.5+)*	boolean	*r ead-only(null)*	This property shall indicate that the Namespace Atomic Write Unit Normal (NAWUN), Namespace Atomic Write Unit Power Fail (NAWUPF), and Namespace Atomic Compare and Write Unit (NACWU) fields are defined for this namespace and should be used by the host for this namespace instead of the c ontroller-level properties AWUN, AWUPF, and ACWU.
Suppo rtsNGUIDReuse *(v1.5+)*	boolean	*r ead-only(null)*	This property shall indicate that the namespace supports the use of an NGUID (namespace globally unique identifier) value.

Property	Type	Attributes	Notes
SupportsThinProvisioning (v1.5+)	boolean	*read-only(null)*	This property shall indicate whether or not the NVMe Namespace supports thin provisioning. Specifically, the namespace capacity reported may be less than the namespace size.
}			
NamespaceId (v1.5+)	string	*read-only(null)*	This property shall contain the NVMe Namespace Identifier for this namespace. This property shall be a hex value. Namespace identifiers are not durable and do not have meaning outside the scope of the NVMe subsystem. NSID 0x0, 0xFFFFFFFF, 0xFFFFFFFE are special purpose values. Pattern: ^0[x X](([a-fA-F]

Property	Type	Attributes	Notes
Num berLBAFormats *(v1.5+)*	integer(By)	*read-only(null)*	This property shall contain the number of LBA data size and metadata size combinations supported by this namespace. The value of this property is between 0 and 16. LBA formats with an index set beyond this value will not be supported.
NVMeVersion *(v1.5+)*	string	*r ead-only(null)*	This property shall contain the version of the NVMe Base Specification supported.
}			
Oem {}	object		This property shall contain the OEM extensions. All values for properties that this object contains shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Property	Type	Attributes	Notes
Operations [{	array		This property shall contain a list of all currently running on the Volume.
AssociatedFeaturesRegistry {	object		This resource shall be used to represent a Feature registry for a Redfish implementation. See the *FeaturesRegistry* schema for details on this property.
@odata.id	string	*read-only*	Link to a FeaturesRegistry resource. See the Links section and the *FeaturesRegistry* schema for details.
}			
OperationName	string	*r ead-only(null)*	The name of the operation.
PercentageComplete	integer	*r ead-only(null)*	The percentage of the operation that has been completed.
}]			

Property	Type	Attributes	Notes
Optim umIOSizeBytes	integer(By)	*read-only(null)*	This property shall contain the optimum IO size to use when performing IO on this volume. For logical disks, this is the stripe size. For physical disks, this describes the physical sector size.
Provi sioningPolicy *(v1.4+)*	string(enum)	• read-write(null)*	This property shall specify the volume's supported storage allocation policy. *For the possible property values, see Pro visioningPolicy in Property details.*
RAIDType *(v1.3.1+)*	string(enum)	*read-only(null)*	This property shall contain the RAID type of the associated Volume. *For the possible property values, see RAIDType in Property details.*

Property	Type	Attributes	Notes
Re adCachePolicy *(v1.4+)*	string(enum)	• read-write(null)*	This property shall contain a boolean indicator of the read cache policy for the Volume. *For the possible property values, see ReadCachePolicy in Property details.*
Re coverableCapaci tySourceCount *(v1.3+)*	integer	*re ad-write(null)*	The value is the number of available capacity source resources currently available in the event that an equivalent capacity source resource fails.
RemainingCa pacityPercent *(v1.2+)*	integer	*r ead-only(null)*	If present, this value shall return {[(SUM(AllocatedBytes) - S UM(ConsumedByte s)]/SUM(Alloca tedBytes)}*100 represented as an integer value.

Property	Type	Attributes	Notes
ReplicaInfo *(v1.1+)* {}	object		This property shall describe the replica relationship between this storage volume and a corresponding source volume. For property details, see ReplicaInfo v1.3.0).
R eplicaTargets *(v1.3+)* [{	array		The value shall reference the target replicas that are sourced by this replica.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			
Status {}	object		The property shall contain the status of the Volume. For property details, see Status.

Property	Type	Attributes	Notes
StorageGroups *(v1.1+)* {	object		The value of this property shall contain references to all storage groups that include this volume. Contains a link to a resource.
@odata.id	string	*read-only*	Link to Collection of *StorageGroup*. See the StorageGroup schema for details.
}			
S tripSizeBytes *(v1.4+)*	integer(By)	• read-write(null)*	The number of consecutively addressed virtual disk blocks (bytes) mapped to consecutively addressed blocks on a single member extent of a disk array. Synonym for stripe depth and chunk size.
VolumeType	string(enum)	*read-only(null)*	This property shall contain the type of the associated Volume. *For the possible property values, see VolumeType in Property details.*

Property	Type	Attributes	Notes
VolumeUsage (v1.4+)	string(enum)	read-only(null)	This property shall contain the volume usage type for the Volume. *For the possible property values, see VolumeUsage in Property details.*
Wri teCachePolicy (v1.4+)	string(enum)	• read-write(null)*	This property shall contain a boolean indicator of the write cache policy for the Volume. *For the possible property values, see W riteCachePolicy in Property details.*
Wr iteCacheState (v1.4+)	string(enum)	read-only(null)	This property shall contain the WriteCacheState policy setting for the Volume. *For the possible property values, see WriteCacheState in Property details.*

Property	Type	Attributes	Notes
WriteHolePro tectionPolicy *(v1.4+)*	string(enum)	*read-write*	This property specifies the policy that is enabled to address the write hole issue on the RAID volume. If no policy is enabled at the moment, this property shall be set to 'Off'. *For the possible property values, see WriteHoleP rotectionPolicy in Property details.*

9.5.37.4 Actions

9.5.37.4.1 AssignReplicaTarget *(v1.4+)*   Description

This action shall be used to establish a replication relationship by assigning an existing volume to serve as a target replica for an existing source volume.

Action URI: {Base URI of target resource}/Actions/Volume.AssignReplicaTarget

Action parameters

Parameter Name	Type	Attributes	Notes
ReplicaType	string(enum)	*required*	This parameter shall contain the type of replica relationship to be created (e.g., Clone, Mirror, Snap). *For the possible property values, see ReplicaType in Property details.*
Repl icaUpdateMode	string(enum)	*required*	This parameter shall specify the replica update mode. *For the possible property values, see Re plicaUpdateMode in Property details.*
TargetVolume**	string	*required*	This parameter shall contain the Uri to the existing target volume.

9.5.37.4.2 ChangeRAIDLayout *(v1.5+)*   Description

This action shall request the system to change the RAID layout of the volume. Depending on the combination of the submitted parameters, this could be changing the RAID type, changing the span count, changing the number of drives used by the volume, or another configuration change supported by the system. Note that usage of this action while online may potentially cause data loss if the available capacity is reduced.

Action URI: {Base URI of target resource}/Actions/Volume.ChangeRAIDLayout

Action parameters

Parameter Name	Type	Attributes	Notes
Drives [{	array	*optional*	This parameter shall contain an array of the drives to be used by the volume.
@odata.id	string(URI)	*read-only*	The value of this property shall be the unique identifier for the resource and it shall be of the form defined in the Redfish specification.
}]			
M ediaSpanCount	integer	*optional*	This parameter shall contain the requested number of media elements used per span in the secondary RAID for a hierarchical RAID type.
RAIDType	string(enum)	*optional*	This parameter shall contain the requested RAID type for the volume. *For the possible property values, see RAIDType in Property details.*

Parameter Name	Type	Attributes	Notes
S tripSizeBytes	integer	*optional*	This parameter shall contain the number of blocks (bytes) requested for the strip size.

9.5.37.4.3 CheckConsistency   Description

This defines the name of the custom action supported on this resource.

Action URI: {Base URI of target resource}/Actions/Volume.CheckConsistency

Action parameters

This action takes no parameters.

9.5.37.4.4 CreateReplicaTarget *(v1.4+)*   Description

This action shall be used to create a new volume resource to provide expanded data protection through a replica relationship with the specified source volume.

Action URI: {Base URI of target resource}/Actions/Volume.CreateReplicaTarget

Action parameters

Parameter Name	Type	Attributes	Notes
ReplicaType	string(enum)	*required*	This parameter shall contain the type of replica relationship to be created (e.g., Clone, Mirror, Snap). *For the possible property values, see ReplicaType in Property details.*

Parameter Name	Type	Attributes	Notes
Repl icaUpdateMode	string(enum)	*required*	This parameter shall specify the replica update mode. *For the possible property values, see Re plicaUpdateMode in Property details.*
Targ etStoragePool	string	*required*	This parameter shall contain the Uri to the existing StoragePool in which to create the target volume.
VolumeName	string	*optional*	This parameter shall contain the Name for the target volume.

9.5.37.4.5 ForceEnable *(v1.5+)*   Description

This action shall request the system to force the volume to enabled state regardless of data loss scenarios.

Action URI: {Base URI of target resource}/Actions/Volume.ForceEnable

Action parameters

This action takes no parameters.

9.5.37.4.6 Initialize *(v1.5+)*   Description

This defines the name of the custom action supported on this resource. If Initial-izeMethod is not specified in the request body, but the property InitializeMethod is specified, the property InitializeMethod value should be used. If neither is specified, the InitializeMethod should be Foreground.

Action URI: {Base URI of target resource}/Actions/Volume.Initialize

Action parameters

Parameter Name	Type	Attributes	Notes
Ini tializeMethod	string(enum)	*optional*	This defines the property name for the action. *For the possible property values, see I nitializeMethod in Property details.*
I nitializeType	string(enum)	*optional*	This defines the property name for the action. *For the possible property values, see InitializeType in Property details.*

9.5.37.4.7 RemoveReplicaRelationship *(v1.4+)*   Description

This action shall be used to disable data synchronization between a source and target volume, remove the replication relationship, and optionally delete the target volume.

Action URI: {Base URI of target resource}/Actions/Volume.RemoveReplicaRelationship

Action parameters

Parameter Name	Type	Attributes	Notes
Delet eTargetVolume	boolean	*optional*	This parameter shall indicate whether or not to delete the target volume as part of the operation. If not defined, the system should use its default behavior.
TargetVolume**	string	*required*	This parameter shall contain the Uri to the existing target volume.

9.5.37.4.8 ResumeReplication *(v1.4+)*   Description

This action shall be used to resume the active data synchronization between a source and target volume, without otherwise altering the replication relationship.

Action URI: {Base URI of target resource}/Actions/Volume.ResumeReplication

Action parameters

Parameter Name	Type	Attributes	Notes
TargetVolume**	string	*required*	This parameter shall contain the Uri to the existing target volume.

9.5.37.4.9 ReverseReplicationRelationship *(v1.4+)*   Description

This action shall be used to reverse the replication relationship between a source and target volume.

Action URI: {Base URI of target resource}/Actions/Volume.ReverseReplicationRelationship

Action parameters

Parameter Name	Type	Attributes	Notes
TargetVolume**	string	*required*	This parameter shall contain the Uri to the existing target volume.

9.5.37.4.10 SplitReplication *(v1.4+)*   Description

This action shall be used to split the replication relationship and suspend data synchronization between a source and target volume.

Action URI: {Base URI of target resource}/Actions/Volume.SplitReplication

Action parameters

Parameter Name	Type	Attributes	Notes
TargetVolume**	string	*required*	This parameter shall contain the Uri to the existing target volume.

9.5.37.4.11 SuspendReplication *(v1.4+)*   Description

This action shall be used to suspend active data synchronization between a source and target volume, without otherwise altering the replication relationship.

Action URI: {Base URI of target resource}/Actions/Volume.SuspendReplication

Action parameters

Parameter Name	Type	Attributes	Notes
TargetVolume**	string	*required*	This parameter shall contain the Uri to the existing target volume.

9.5.37.5 Property details

9.5.37.5.1 AccessCapabilities: Each entry shall specify a current storage access capability.

string	Description
Append	This enumeration literal shall indicate that the storage may be written only to append.
Execute	This value shall indicate that Execute access is allowed by the file share.
Read	This enumeration literal shall indicate that the storage may be read.
Streaming	This enumeration literal shall indicate that the storage may be read sequentially.
Write	This enumeration literal shall indicate that the storage may be written multiple times.
WriteOnce	This enumeration literal shall indicate that the storage may be written only once.

9.5.37.5.2 EncryptionTypes: This property shall contain the types of encryption used by this Volume.

string	Description
ControllerAssisted	The volume is being encrypted by the storage controller entity.
NativeDriveEncryption	The volume is utilizing the native drive encryption capabilities of the drive hardware.

string	Description
SoftwareAssisted	The volume is being encrypted by software running on the system or the operating system.

9.5.37.5.3 InitializeMethod: This defines the property name for the action.

string	Description
Background	The volume will be available for use immediately, with data erasure and preparation to happen as background tasks.
Foreground	Data erasure and preparation tasks will complete before the volume is presented as available for use.
Skip	The volume will be available for use immediately, with no preparation.

9.5.37.5.4 InitializeType: This defines the property name for the action.

string	Description
Fast	The volume is prepared for use quickly, typically by erasing just the beginning and end of the space so that partitioning can be performed.
Slow	The volume is prepared for use slowly, typically by completely erasing the volume.

9.5.37.5.5 ProvisioningPolicy: This property shall specify the volume's supported storage allocation policy.

string	Description
Fixed	This enumeration literal specifies storage shall be fully allocated.
Thin	This enumeration literal specifies storage may be over allocated.

9.5.37.5.6 RAIDType: This parameter shall contain the requested RAID type for the volume.

string	Description
None *(v1.4.2+)*	A placement policy with no redundancy at the device level.
RAID0	A placement policy where consecutive logical blocks of data are uniformly distributed across a set of independent storage devices without offering any form of redundancy. This is commonly referred to as data striping. This form of RAID will encounter data loss with the failure of any storage device in the set.
RAID00	A placement policy that creates a RAID 0 stripe set over two or more RAID 0 sets. This is commonly referred to as RAID 0+0. This form of data layout is not fault tolerant; if any storage device fails there will be data loss.
RAID01	A data placement policy that creates a mirrored device (RAID 1) over a set of striped devices (RAID 0). This is commonly referred to as RAID 0+1 or RAID 0/1. Data stored using this form of RAID is able to survive a single RAID 0 data set failure without data loss.

string	Description
RAID1	A placement policy where each logical block of data is stored on more than one independent storage device. This is commonly referred to as mirroring. Data stored using this form of RAID is able to survive a single storage device failure without data loss.
RAID10	A placement policy that creates a striped device (RAID 0) over a set of mirrored devices (RAID 1). This is commonly referred to as RAID 1/0. Data stored using this form of RAID is able to survive storage device failures in each RAID 1 set without data loss.
RAID10E	A placement policy that uses a RAID 0 stripe set over two or more RAID 10 sets. This is commonly referred to as Enhanced RAID 10. Data stored using this form of RAID is able to survive a single device failure within each nested RAID 1 set without data loss.
RAID10Triple	A placement policy that uses a striped device (RAID 0) over a set of triple mirrored devices (RAID 1Triple). This form of RAID can survive up to two failures in each triple mirror set without data loss.

string	Description
RAID1E	A placement policy that uses a form of mirroring implemented over a set of independent storage devices where logical blocks are duplicated on a pair of independent storage devices so that data is uniformly distributed across the storage devices. This is commonly referred to as RAID 1 Enhanced. Data stored using this form of RAID is able to survive a single storage device failure without data loss.
RAID1Triple	A placement policy where each logical block of data is mirrored three times across a set of three independent storage devices. This is commonly referred to as three-way mirroring. This form of RAID can survive two device failures without data loss.
RAID3	A placement policy using parity-based protection where logical bytes of data are uniformly distributed across a set of independent storage devices and where the parity is stored on a dedicated independent storage device. Data stored using this form of RAID is able to survive a single storage device failure without data loss. If the storage devices use rotating media, they are assumed to be rotationally synchronized, and the data stripe size should be no larger than the exported block size.

string	Description
RAID4	A placement policy using parity-based protection where logical blocks of data are uniformly distributed across a set of independent storage devices and where the parity is stored on a dedicated independent storage device. Data stored using this form of RAID is able to survive a single storage device failure without data loss.
RAID5	A placement policy using parity-based protection for storing stripes of 'n' logical blocks of data and one logical block of parity across a set of 'n+1' independent storage devices where the parity and data blocks are interleaved across the storage devices. Data stored using this form of RAID is able to survive a single storage device failure without data loss.
RAID50	A placement policy that uses a RAID 0 stripe set over two or more RAID 5 sets of independent storage devices. Data stored using this form of RAID is able to survive a single storage device failure within each RAID 5 set without data loss.
RAID6	A placement policy using parity-based protection for storing stripes of 'n' logical blocks of data and two logical blocks of independent parity across a set of 'n+2' independent storage devices where the parity and data blocks are interleaved across the storage devices. Data stored using this form of RAID is able to survive any two independent storage device failures without data loss.

string	Description
RAID60	A placement policy that uses a RAID 0 stripe set over two or more RAID 6 sets of independent storage devices. Data stored using this form of RAID is able to survive two device failures within each RAID 6 set without data loss.
RAID6TP	A placement policy that uses parity-based protection for storing stripes of 'n' logical blocks of data and three logical blocks of independent parity across a set of 'n+3' independent storage devices where the parity and data blocks are interleaved across the storage devices. This is commonly referred to as Triple Parity RAID. Data stored using this form of RAID is able to survive any three independent storage device failures without data loss.

9.5.37.5.7 ReadCachePolicy: This property shall contain a boolean indicator of the read cache policy for the Volume.

string	Description
AdaptiveReadAhead	A caching technique in which the controller dynamically determines whether to pre-fetch data anticipating future read requests, based on previous cache hit ratio.
Off	The read cache is disabled.
ReadAhead	A caching technique in which the controller pre-fetches data anticipating future read requests.

9.5.37.5.8 ReplicaType: This parameter shall contain the type of replica relationship to be created (e.g., Clone, Mirror, Snap).

string	Description
Clone	This enumeration literal shall indicate that replication shall create a point in time, full copy the source.
Mirror	This enumeration literal shall indicate that replication shall create and maintain a copy of the source.
Snapshot	This enumeration literal shall indicate that replication shall create a point in time, virtual copy of the source.
TokenizedClone	This enumeration literal shall indicate that replication shall create a token based clone.

9.5.37.5.9 ReplicaUpdateMode: This parameter shall specify the replica update mode.

string	Description
Active	This enumeration literal shall indicate Active-Active (i.e. bidirectional) synchronous updates.
Adaptive	This enumeration literal shall indicate that an implementation may switch between synchronous and asynchronous modes.
Asynchronous	This enumeration literal shall indicate Asynchronous updates.
Synchronous	This enumeration literal shall indicate Synchronous updates.

9.5.37.5.10 VolumeType: This property shall contain the type of the associated Volume.

string	Description
Mirrored	The volume is a mirrored device.
NonRedundant	The volume is a non-redundant storage device.
RawDevice	The volume is a raw physical device without any RAID or other virtualization applied.
SpannedMirrors	The volume is a spanned set of mirrored devices.
SpannedStripesWithParity	The volume is a spanned set of devices which uses parity to retain redundant information.
StripedWithParity	The volume is a device which uses parity to retain redundant information.

9.5.37.5.11 VolumeUsage: This property shall contain the volume usage type for the Volume.

string	Description
CacheOnly	The volume shall be allocated for use as a non-consumable cache only volume.
Data	The volume shall be allocated for use as a consumable data volume.
ReplicationReserve	The volume shall be allocated for use as a non-consumable reserved volume for replication use.
SystemData	The volume shall be allocated for use as a consumable data volume reserved for system use.

string	Description
SystemReserve	The volume shall be allocated for use as a non-consumable system reserved volume.

9.5.37.5.12 WriteCachePolicy: This property shall contain a boolean indicator of the write cache policy for the Volume.

string	Description
Off *(v1.4.1+)*	Indicates that the write cache shall be disabled.
ProtectedWriteBack	A caching technique in which the completion of a write request is signaled as soon as the data is in cache, and actual writing to non-volatile media is guaranteed to occur at a later time.
UnprotectedWriteBack	A caching technique in which the completion of a write request is signaled as soon as the data is in cache; actual writing to non-volatile media is not guaranteed to occur at a later time.
WriteThrough	A caching technique in which the completion of a write request is not signaled until data is safely stored on non-volatile media.

9.5.37.5.13 WriteCacheState: This property shall contain the WriteCacheState policy setting for the Volume.

string	Description
Degraded	Indicates an issue with the cache state in which the cache space is diminished or disabled due to a failure or an outside influence such as a discharged battery.
Protected	Indicates that the cache state type in use generally protects write requests on non-volatile media.
Unprotected	Indicates that the cache state type in use generally does not protect write requests on non-volatile media.

9.5.37.5.14 WriteHoleProtectionPolicy: This property specifies the policy that is enabled to address the write hole issue on the RAID volume. If no policy is enabled at the moment, this property shall be set to 'Off'.

string	Description
DistributedLog	The policy that distributes additional log (e.q. checksum of the parity) among the volume's capacity sources to address write hole issue. Additional data is used to detect data corruption on the volume.
Journaling	The policy that uses separate block device for write-ahead logging to address write hole issue. All write operations on the RAID volume are first logged on dedicated journaling device that is not part of the volume.
Oem	The policy that is Oem specific. The mechanism details are unknown unless provided separately by the Oem.

string	Description
Off	The support for addressing the write hole issue is disabled. The volume is not performing any additional activities to close the RAID write hole.

9.5.38 VolumeCollection

9.5.38.1 URIs   /redfish/v1/CompositionService/ResourceBlocks/{*ResourceBlockId*}/Storage/{*StorageId*}/Volu

/redfish/v1/CompositionService/ResourceBlocks/{*ResourceBlockId*}/Systems/{*ComputerSystemId*}/Storage/{*St*

/redfish/v1/ResourceBlocks/{*ResourceBlockId*}/Storage/{*StorageId*}/Volumes   /red-

fish/v1/ResourceBlocks/{*ResourceBlockId*}/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/Volumes

/redfish/v1/Storage/{*StorageId*}/ConsistencyGroups/{*ConsistencyGroupId*}/Volumes

/redfish/v1/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*}/ProvidingVolu

/redfish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/AllocatedVolumes /red-

fish/v1/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourceId*}/ProvidingVolu

/redfish/v1/Storage/{*StorageId*}/Volumes /redfish/v1/StorageServices/{*StorageServiceId*}/ConsistencyGroups/{*C*

/redfish/v1/StorageServices/{*StorageServiceId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*CapacitySourceId*}

/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/AllocatedVolumes

/redfish/v1/StorageServices/{*StorageServiceId*}/StoragePools/{*StoragePoolId*}/CapacitySources/{*CapacitySourc*

/redfish/v1/StorageServices/{*StorageServiceId*}/Volumes /redfish/v1/StorageServices/{*StorageServiceId*}/Volum

/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/ConsistencyGroups/{*ConsistencyGroupId*}/Volum

/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/FileSystems/{*FileSystemId*}/CapacitySources/{*Ca*

/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/AllocatedVolumes

/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/StoragePools/{*StoragePoolId*}/CapacitySources/

/redfish/v1/Systems/{*ComputerSystemId*}/Storage/{*StorageId*}/Volumes

9.5.38.2 Properties

Property	Type	Attributes	Notes
Description	string	*r ead-only(null)*	This object represents the description of this resource. The resource values shall comply with the Redfish Specific ation-described requirements.
Members [{	array		The value of each member entry shall reference a Volume resource.
@odata.id	string	*read-only*	Link to a Volume resource. See the Links section and the *Volume* schema for details.
}]			
Members@o data.nextLink	string(URI)	*read-only*	The value of this property shall be a URI to a resource, with the same @odata.type, containing the next set of partial members.

Property	Type	Attributes	Notes
Name	string	*read-only*	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specific ation-described requirements. This string value shall be of the 'Name' reserved word format.
Oem {}	object		This property shall contain the OEM extensions. All values for properties contained in this object shall conform to the Redfish Specific ation-described requirements. For property details, see Oem.

Annex A: Bibliography

A.1 Overview

The following referenced documents provide important support for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

A.2 Informational references

The informational references are summarized in Table A.1.

Tag	Title (Version)	Author	URL
Pro files	Swordfish Profile Bundle Working Draft	SNIA	<https://www.snia.org/forums/smi/swordfish>
TLS	TLS Specification for Storage Systems	SNIA	<https://www.snia.org/tech_activities/standards/curr_standards/tls>

Table A.1: Informational References