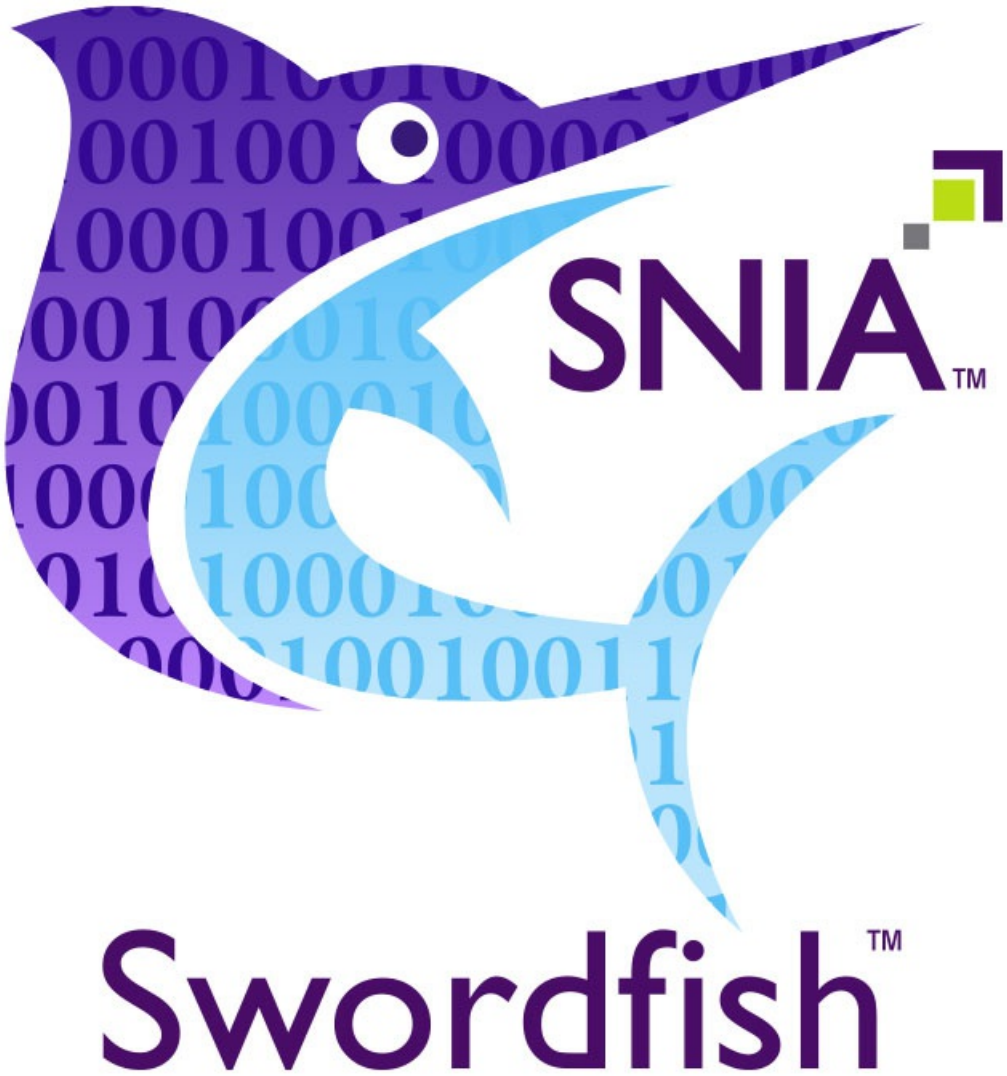

Swordfish Scalable Storage Management API Specification

Swordfish Work in Progress Notice

Version 1.0.2

Publication of this Working Draft for review and comment has been approved by the Scalable Storage Management Technical Work Group. This draft represents a 'best effort' attempt by the Scalable Storage Management Technical Work Group to reach preliminary consensus, and it may be updated, replaced, or made obsolete at any time. This document should not be used as reference material or cited as other than a 'work in progress.' Suggestions for revision should be directed to http://www.snia.org/feedback.

SNIA Work in Progress

Released: WIP November 1, 2016

USAGE

> INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

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 specification, 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/.

Revision History

Date	Revision	Notes
19 September 2016	1.0.0	Initial Release
12 October 2016	1.0.1	General clean up and formatting consistency
		Clarify use of StorageGroup
		Detail interactions between DefaultValue and Nullable in schema attributes
		Set default values for Boolean attributes
		Clarify appropriate EntitySet referencing
		Improve descriptions for many schema attributes
		Replace IsDefault with reference to ClassOfService in SZtoragePool and StorageService
		Align Location with Redfish model
		Change time values to conform to ISO 8601
		Collapse TargetEndpointGroup and InitiatorEndpointGroup into EndpointGroup
		Property and enumeration deleted from StorageReplicaInfo to remove redundancy with ReplicaSyncType
		Add DefaultClassOfService link to StoragePool and StorageVolume in lieu of ClassOfService.IsDefault
		Remove invalid measurement annotation from DataProtectionLoSCapabilities
		Moved schedule to Redfish

Date	Revision	Notes
1 November 2016	1.0.2	Spec and schema updates:
		Change multiple collections' types from collections (arrays) to ResourceCollections to conform to Redfish usage guidelines:
		- Capacity.CapacitySource ProvidingVolumes, ProvidingPools and ProvidingDrives
		- StorageService.StorageService StorageGroups, ClientEndpointGroups and ServerEndpointGroups
		Change multiple collections' types from collections (arrays) to ResourceCollections to conform to Redfish usage guidelines and move NavigationProperties from Links section:
		- EndpointGroup Endpoints
		- FileShare EthernetInterfaces
		- StoragePool ClassesOfService
		- StorageService ClassesOfService

Suggestion for changes or modifications to this document should be sent to the SNIA Scalable Storage Management (SSM) Technical Working Group at http://www.snia.org/feedback/.

Contact SNIA

SNIA Web Site

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 defect 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.

Table of Contents

Swordfish Scalable Storage Management API Specification 1
Swordfish Work in Progress Notice 1
Version 1.0.2 1
SNIA Work in Progress 2
Released: WIP November 1, 2016 2
USAGE 2
DISCLAIMER 3
Revision History 3
Contact SNIA 4
FEEDBACK AND INTERPRETATIONS 4
INTENDED AUDIENCE 4
VERSIONING POLICY 4
Table of Contents 6
1 Abstract 8
2 Scope 8
3 Normative References 8
3.1 Overview 8
3.2 Approved references 8
3.3 References under development 10
3.4 Other references 10
4 Terms and Definitions 10
4.1 Overview 10
4.2 Swordfish-specific 10
4.3 Reference to Redfish terms 11
4.4 Keywords (normative language terms) 12
5 Security 12
6 Swordfish Overview 12
6.1 Introduction 13
6.2 Relation to Redfish 13
6.3 Discovering Swordfish resources 13
6.4 Storage Services 14
6.5 The ClassOfService resource 15
6.6 The Endpoint resource 16
6.7 The Endpoint Collection resource 16
6.8 The Endpoint Group resource 16
6.9 The Endpoint Group Collection resource 16
6.10 The StorageGroup resource 16
6.11 The StoragePool resource 17
6.12 The Volume resource 17
6.13 The FileSystem resource 17
7 Data model and schema 17
7.1 Schema Introduction and Overview 17
7.2 Swordfish extensions to Redfish 18
7.3 Common schema attributes 18
7.4 Default values and NULLABLE attributes 19
7.5 Common schema annotations 19
7.6 Entity Sets 20
7.7 Addressing entities within a collection 20
7.8 Addressing members of a ResourceCollection 21
7.9 Schema repository 21
7.10 Referencing other schemas 21
8 Swordfish type definitions 21
8.1 Overview 21
8.2 ClassOfService 1.0.0 21
8.3 ClassOfServiceCollection 23

8.4 DataProtectionLoSCapabilities 1.0.0 24
8.5 DataSecurityLoSCapabilities 1.0.0 25
8.6 DataStorageLoSCapabilities 1.0.0 27
8.7 DriveCollection 28
8.8 EndpointCollection 28
8.9 EndpointGroup 1.0.0 29
8.10 FileShare 1.0.0 30
8.11 FileSystem 1.0.0 33
8.12 FileSystemCollection 46
8.13 HostedStorageServices 48
8.14 IOConnectivityLoSCapabilities 1.0.0 50
8.15 IOPerformanceLoSCapabilities 1.0.0 51
8.16 StorageGroup 1.0.0 53
8.17 StorageGroupCollection 62
8.18 StoragePool 1.0.0 64
8.19 StoragePoolCollection 68
8.20 StorageService 1.0.0 70
8.21 StorageServiceCollection 75
8.22 StorageSystemCollection 78
8.23 Volume 1.1.0 78
8.24 VolumeCollection 93

1 Abstract

The Swordfish Scalable Storage Management API ("Swordfish") uses RESTful interface semantics and a standardized data model to provide a scalable, customer-centric interface for managing storage and related data services.

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 Scope

The Swordfish specification extends the Redfish Scalable Platforms Management API Specification (DSP0266) from the DMTF. Thus a Swordfish conformant implementation is at the same time a Redfish conformant implementation.

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 specifies additional data models and behaviors for the management of storage systems and storage infrastructure. A Swordfish implementation must also conform to all requirements specified in the Redfish specifications.

3 Normative References

3.1 Overview

The following referenced documents 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

Table 1: Approved normative references

Tag	Title (Version)	Author	URL
ISO-8601	Data elements and interchange formats -- Information interchange -- Representation of dates and times -- Part 1: Basic rules	ISO/IEC	http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=70907
ISO-Direct	ISO/IEC Directives, Part 2 Principles and rules for the structure and drafting of ISO and IEC documents (Seventh Edition, 2016)	ISO/IEC	http://isotc.iso.org/livelink/livelink/ fetch/2000/2122/4230450/4230456/ISO_IEC_Directives *Part_2 Principles_and_rules_for_the structure_and_drafting_of_ISO_and_IEC* documents_-*2016*%287th_edition%29_-_PDF.pdf?nodeid=17667902&vernum=-2
Redfish	Redfish Scalable Platforms Management API Specification (v1.0.4)	DMTF	http://www.dmtf.org/sites/default/files/standards/documents/DSP0266_1.0.4.pdf
OData	Open Data Protocol (v. 4.0)	OASIS	https://www.oasis-open.org/standards#odatav4.0
RFC3986	Uniform Resource Identifier (URI): Generic Syntax (2005)	The Internet Society	http://www.rfc-base.org/txt/rfc-3986.txt
CSDL	Common Schema Definition Language (4.0)	OASIS	http://docs.oasis-open.org/odata/odata/v4.0/odata-v4.0-part3-csdl.html
ITIL	ITIL Glossary (2011)	ITIL	https://www.axelos.com/Corporate/media/ Files/Glossaries/ITIL_2011_Glossary_GB-v1-0.pdf

Tag	Title (Version)	Author	URL
Units	The Unified Code for Units of Measure (v2.0.1)	Regenstrief Institute, Inc. and the UCUM Organization	http://unitsofmeasure.org/trac
TLS	Transport Layer Security (TLS) Protocol Version 1.2	IETF	https://www.ietf.org/rfc/rfc5246.txt
SPC-4	SCSI Primary Commands - 4 (SPC-4) INCITS 513-2015	T10	http://www.techstreet.com/cgi-bin/joint.cgi/incits

3.3 References under development

None defined in this document.

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

4.2.1 Definitions

The following terms are used in this document.

Table 2: Swordfish terms

Term	Definition
Entity	An element in a model that represents resources. The element may be either a type declaration or a model instance representing an instance of the resource.
Entity Instance	A model element that represents the information and behaviors of a particular instance of an entity.
Entity Type	A model element that specifies the structure, information and behaviors of an entity.
Instance	See Entity Instance.
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.
Metamodel	A model that defines the semantics for the construction of a model.
Model	A set of entities and the relationships between them that define the semantics, behavior and state of that set.
Resource	A named item of interest. The item may be be a collection of other items. A resource may be assigned a URI that allows it to receive and process messages. A particular instance of a resource is represented in the model by an entity instance. The type of a resource is represented in the model by an entity type.
Schema	A formal language representation of a model that conforms to a metamodel.
Service Document	The term Service Document is used to refer to a particular resource that is directly accessed via the 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. See also OData Service Document
Swordfish service	A service that is a Redfish service and that implements Swordfish extensions to the Redfish model that conform to the requirements of this document.

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 here, as an aid to the reader.

Table 3: Redfish terms

Term	Definition
OData	The Open Data Protocol, as defined in OData-Protocol.
OData Service Document	The name for a resource that provides information about the Service Root. The Service Document provides a standard format for enumerating the resources exposed by the service that enables generic hypermedia-driven OData clients to navigate to the resources of the Redfish Service. See also Service Document
Redfish Schema	The CSDL defintion of Redfish resources.
Redfish service	An OData service that conforms to requirements of the Redfish specification.

Term	Definition
Redfish Service Entry Point	Also referred to as "Service Entry Point". An URI through which a particular instance of a Redfish Service is accessed. A Redfish Service may have more than one Service Entry Point
Request	A message from a Client to a Server. It consists of a request line (which includes the Operation), request headers, an empty line and an optional message body.
Service Root	The term Service Root is used to refer to a particular resource that is directly accessed via the Redfish 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 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 below.

Table 4: 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 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 version 1.2 or greater

6 Swordfish Overview

6.1 Introduction

Swordfish defines a scalable, REST-based interface for managing an ecosystem that includes storage systems, and that conforms to the requirements of Redfish.

6.2 Relation to Redfish

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 are present in a Swordfish model.

A complete Swordfish implementation must include the Redfish-defined `ServiceRoot`. It may also support the Redfish-defined `ServiceContainer`. Storage systems managed by the Swordfish storage service are located in the `ServiceRoot` (and `ServiceContainer`) via the `StorageSystems` resource collection. They are modeled using Redfish `ComputerSystems`. The physical infrastructure is modeled using Redfish `Chassis`.

Each Swordfish `StorageService` is located in the `ServiceRoot` (and `ServiceContainer`) via the `StorageServices` resource collection. All Swordfish defined instances are located through `StorageService` intances. A Swordfish management client may focus entirely on entities defined by the Swordfish schema.

The combined Redfish and Swordfish models defines information requirements and constraints on the values that are used as input or output of the operations supported by the Swordfish interface. The Swordfish interface relies on the operations specified by the OData REST protocol (#normative-references). Additional operations (known as Actions) are also defined by the model. The information content is defined by a schema specified using the Common Schema Definition Language (CSDL) (#normative-references) defined by the OData organization within OASIS (https://www.oasis-open.org/).

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.

6.3 Discovering Swordfish resources

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

- `/Redfish`, which contains one or more version properties for the integrated Swordfisn 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 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 navigation from it will be the same elements found via the `ServiceRoot`.

> Note: Swordfish does not define a separate service from a Redfish service. 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.

6.3.1 The ServiceRoot and ServiceContainer entities

A **GET** to the `/redfish/v1` will return the `ServiceRoot` entity. A **GET** to the `/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 principal properties utilized for Swordfish management. All other `ServiceRoot` and `ServiceContainer` properties are as defined by Redfish.

- `Systems`: A reference to a `ComputerSystemCollection` with members that are of type `ComputerSystem` that support general applications.
- `StorageSystems`: 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.
- `StorageServices`: A reference to a `StorageServiceCollection` with members that are of type `StorageService`.
- `Chassis`: A reference to a `ChassisCollection` with members that are of type `Chassis`.

6.3.1.1 The StorageSystems resource collection

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.

6.3.1.2 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.

6.3.1.3 The StorageServices resource collection

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

6.3.1.4 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.

6.4 Storage Services

6.4.1 The StorageService resource

A storage service is a service modeled as a `StorageService` The storage service is hosted on a storage system and exposes logical storage, resources and 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 supported by the storage service.

- `ClassesOfService`: A reference to a resource collection that specifies the set supported `ClassOfService` resources.
- `ClientEndpointGroups`: A reference to a resource collection that collects `ClientEndpointGroup` 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.
- `Endponts`: 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.
- `ServerEndpointGroups`: A reference to a resource collection that collects `ServerEndpointGroup` 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.

6.4.1.1 The StorageService Links section

This property specifies the storage system that hosts the storage service.

- `HostingSystem`: A reference to the `ComputerSystem` instance that hosts this `StorageService`.

The following properties each include a set of attributes that each describe a range of capabilities that the storage service can support for a particular kind of service.

- `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 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.

6.5 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 ITIL.)

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.

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

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.

6.6 The Endpoint resource

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

6.7 The Endpoint Collection resource

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

6.8 The Endpoint Group resource

The `EndpointGroup` is a resource that represents a set of `Endpoint` resources that have the same management characteristics and which will all have the same access state.

6.9 The Endpoint Group Collection resource

The `EndpointGroupCollection` is resource collection that references a set of `EndpointGroup` resources.

6.10 The StorageGroup resource

StorageGroups represent a set of volumes that are managed as a group with the same consistency requirements. The volumes of a storage

group are collectively exposed or hidden to a set of clients.

The set of volumes is specified by the `Volumes` attribute, which is a resource collection that references volumes.

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.

6.11 The StoragePool resource

The `StoragePool` resource represents a factory that has amount of storage capacity and has the ability to produce storage volumes or other storage pools conforming to one or more classes of service. Storage Devices the physical capacity used to create storage pools.

The following are the principal properties of `StoragePool` that locate resources provisioned or supported by the storage pool.

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

6.12 The Volume resource

This `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 represent block addressable capacity that is conformant to a `ClassOfService`.

6.13 The FileSystem resource

This `FileSystem` resource represents a file system. File systems represent file-addressable capacity that are conformant to a `ClassOfService`. Each `FileSystem` may contain a collection of `FileShares` that can be presented to hosts.

7 Data model and schema

7.1 Schema Introduction and Overview

The Swordfish model is an extension to Redfish, as illustrated in Figure 1. A Swordfish instance cannot be implemented solely from Swordfish schema; it must include Redfish schema.

!(updated_storage_model_diagram.png)

A complete Swordfish implementation includes a Redfish-defined Service Root, is instantiated upon a StorageSystem/ComputerSystem, and runs on a Redfish Chassis. At the same time, a storage client may focus entirely on the storage schema instantiations and never interact with the Redfish portion of an implementation.

Swordfish is defined in terms of schema extended from Redfish which are defined below. This section provides additional definition and context for these schema.

7.2 Swordfish extensions to Redfish

7.2.1 Overview

Redfish has added two properties to the ServiceRoot 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` - at least one entry in its `StorageServices.Members` property.

The second is `StorageServices`. This property references a collection of `StorageService` resources. It provides the client an efficient means to search across all `StorageService` resources, regardless of which `ComputerSystem` is supporting the service.

7.2.2 Swordfish and Redfish specific OEM or vendor extensions

The Swordfish and Redfish models are extended by subclassing the `OEM` `ComplexTypes` that are defined in the Swordfish and Redfish schemas.

7.2.3 OData specific OEM or vendor extensions

In addition to extending the Redfish model as described above. An OEM may extend the Redfish `ServiceContainer` by defining a new `EntityContainer` that extends the ServiceContainer found in the Redfish ServiceRoot_v1.xml file, (see OData EntityContainer).

> Note: This has the same semantics as subclassing in a typical object oriented environment.

An OEM extended implementation of the Swordfish service would access OEM extensions to `EntityContainer` via the service entry-point `/redfish/v1/odata`.

7.3 Common schema attributes

The following table lists common schema attributes used in the definition of Swordfish, for details see CSDL

Table 4: Schema attributes

Name	Applies to	Description
Abstract	ComplexType, EntityType	If true, the entity may not be instantiated

Name	Applies to	Description
BaseType	ComplexType, EntityType	Names an inherited element.
DefaultValue	Property	The value of a property if not explicitly set
Name	All	The name of the schema element
Nullable	NavigationProperty, Property	If false, the qualified property shall have a value. The default value is true. A navigation property whose Type attribute specifies a collection shall not specify Nullable=false, as the collection always exists, but may just be empty. *Note: Null is not itself a value, but is an indication of no value.*
Type	Property	The type of the element

7.4 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 5. 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 5: 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

7.5 Common schema annotations

The following table 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 5: Schema annotations

Name	Applies to	Description
AllowableValues	Parameter	The set of allowable values for a parameter
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.
RequiredIOnCreate	NavigationProperty, Property	If true, property is required on creation
Unit	Property	The unit of measure for the value.

7.6 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.

7.7 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.

7.8 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.

7.9 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.

7.10 Referencing other schemas

Swordfish directly reference the following Redfish schemas. - Chassis - ChassisCollection - ComputerSystem - ComputerSystemCollection - Drive - Endpoint - EthernetInterface - EventService - Location - RedfishExtensions - Redundancy - ResourceTask - Schedule - ServiceContainer - ServiceRoot

Other Redfish schema may be added by inference or directly to implementations. Examples are available in the Swordfish mockups.

8 Swordfish type definitions

8.1 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.

8.2 ClassOfService 1.0.0

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

ClassOfServiceVersion	string, null *read-write*	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).
Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.
LinesOfService {	object, null *read-write*	The value of this property shall define the required choices of utility or warranty.
DataProtectionLinesOfService [{}]	array *read-write*	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.
DataSecurityLinesOfService [{}]	array *read-write*	The value shall be a set of data security service options.
DataStorageLinesOfService [{}]	array *read-write*	The value shall be a set of data protection service options.
IOConnectivityLinesOfService [{}]	array *read-write*	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.
IOPerformanceLinesOfService [{}]	array *read-write*	The value shall be a set of IO performance service options.
}		

Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.3 ClassOfServiceCollection

This collection shall contain references to all ClassOfService resource instances sharing the same parent resource.

Description	null *read-write*	
Members [{	array *read-only*	The value of each member entry shall reference a ClassOfService resource.
ClassOfServiceVersion	string, null *read-write*	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).
Description	null *read-write*	
Id	*read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.
LinesOfService {}	object, null *read-write*	The value of this property shall define the required choices of utility or warranty.

Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}]		
Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.4 DataProtectionLoSCapabilities 1.0.0

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.

Description	null *read-write*	
Id	*read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.
Links {	object *read-only*	The value of this property shall contains links to other resources that are not contained in this resource.
Oem	*read-write*	This object represents the Oem property. All values for resources described by this schema shall comply to the requirements as described in the Redfish specification.

SupportedReplicaOptions [{}]	array *read-write*	The collection shall contain known and supported replica Classes of Service.
}		
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
SupportedDataProtectionLinesOfService [{}]	array *read-write*	The collection shall contain known and supported DataProtectionLinesOfService.
SupportedMinLifetime [{}]	array *read-write*	The value of each entry shall be an ISO 8601 duration that shall specify the minimum lifetime or the replica.
SupportedRecoveryGeographicObjectives [{}]	array *read-write*	The value of each entry shall specify a supported failure domain.
SupportedRecoveryPointObjectiveSeconds [{}]	array *read-write*	The value of each entry shall specify a supported time interval defining the maximum source information that may be lost on failure
SupportedRecoveryTimeObjectives [{}]	array *read-write*	The value of each entry shall specify a supported expectation for time to access an alternate replica.
SupportedReplicaTypes [{}]	array *read-write*	The value of each entry shall specify a supported replica type
SupportsIsolated	boolean, null *read-write*	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.

8.5 DataSecurityLoSCapabilities 1.0.0

This resource may be used to describe data security capabilities.

Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value identifies this resource. The value shall be unique within the managed ecosystem.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
SupportedAntivirusEngineProviders [{}]	array *read-write*	The entry values shall specify supported AntiVirus providers.
SupportedAntivirusScanPolicies [{}]	array *read-write*	The enumeration literal shall specify supported policies that trigger an AntiVirus scan.
SupportedChannelEncryptionStrengths [{}]	array *read-write*	The enumeration literal shall specify supported key sizes in a symmetric encryption algorithm (AES) for transport channel encryption.
SupportedDataSanitizationPolicies [{}]	array *read-write*	The enumeration literal shall specify supported data sanitization policies.
SupportedDataSecurityLinesOfService [{}]	array *read-write*	The collection shall contain supported DataSecurity service options.
SupportedHostAuthenticationTypes [{}]	array *read-write*	The enumeration literal shall specify supported authentication types for hosts (servers) or initiator endpoints.

SupportedMediaEncryptionStrengths [{}]	array *read-write*	The enumeration literal shall specify supported key sizes in a symmetric encryption algorithm (AES) for media encryption.
SupportedSecureChannelProtocols [{}]	array *read-write*	The enumeration literal shall specify supported protocols that provide encrypted communication.
SupportedUserAuthenticationTypes [{}]	array *read-write*	The enumeration literal shall specify supported authentication types for users (or programs).

8.6 DataStorageLoSCapabilities 1.0.0

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

Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
SupportedAccessCapabilities [{}]	array *read-write*	Each entry specifies a storage access capability.
SupportedDataStorageLinesOfService [{}]	array *read-write*	The collection shall contain known and supported DataStorageLinesOfService.

SupportedProvisioningPolicies [{}]	array *read-write*	This collection specifies supported storage allocation policies.
SupportedRecoveryTimeObjectives [{}]	array *read-write*	This collection specifies supported expectations for time to access the primary store after recovery.
SupportsSpaceEfficiency	boolean, null *read-write*	The value specifies whether storage compression or deduplication is supported. The default value for this property is false.

8.7 DriveCollection

An instance of this resource shall reference the set of Drive resources known in the scope of its use.

Description	null *read-write*	
Members [{}]	array *read-only*	The value of each entry of this property shall reference a Drive resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.8 EndpointCollection

An instance of this resource shall reference the set of Endpoint resources known in the scope of its use.

Description	null *read-write*	
Members [{}]	array *read-only*	The value of each member entry shall reference an Endpoint resource.
Name	 *read-write*	

Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.9 EndpointGroup 1.0.0

An EndpointGroup represents a collection of endpoints that are managed as a unit. By grouping together a collection of Endpoints, the EndpointGroup allows a collection of entities from differing sources or hosts to be manipulated uniformly and efficiently.

For any given EndpointGroup, all of its endpoints act exclusively as either server endpoints or client endpoints, as indicated by the value of the EndpointType property. Similarly, each Endpoint within a group has the same AccessState.

A server or client may define multiple EndpointGroup entities that access the same set of resources or functionality. A group may be designated as preferred, which signifies that access should be directed through its members in preference to the Endpoints listed in other EndpointGroups. If the value of EndpointType is Server, an EndpointGroup entity can be used to represent target port group as defined by SCSI. In that mode, the value of the TargetEndpointGroupIdentifier should correspond to the target port group number. (See clause "Device Identification VPD page" as defined in the SCSI Primary Commands specification.)

AccessState	null *read-write*	Access to all associated resources through all aggregated endpoints shall share this access state.
Description	null *read-write*	
Endpoints {	object, null *read-write*	The value of each entry shall reference an Endpoint resource.
Description	null *read-write*	
Members [{}]	array *read-only*	The value of each member entry shall reference an Endpoint resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.

}		
GroupType	null *read-write*	The group contains only endpoints of a given type Client/Initiator or Server/Target. If this endpoint group represents a SCSI target group, the value of GroupType shall be Server.
Id	 *read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.
Links {	object *read-only*	This structure shall contain references to resources that are not contained within this resource.
Oem	 *read-write*	This object represents the Oem property. All values for resources described by this schema shall comply to the requirements as described in the Redfish specification.
}		
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Preferred	boolean, null *read-write*	A value of True in this property shall indicate that access to the associated resource through the endpoints in this endpoint group is preferred over access through other endpoints. The default value for this property is false.
TargetEndpointGroupIdentifier	number, null *read-write*	If this endpoint group represents a SCSI target group, the value of this property shall contain a SCSI defined identifier for this group, which corresponds to the TARGET PORT GROUP field in the REPORT TARGET PORT GROUPS response and the TARGET PORT GROUP field in an INQUIRY VPD page 85 response, type 5h identifier. See the INCITS SAM-5 specification.

8.10 FileShare 1.0.0

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

CASupported	boolean, null *read-write*	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 FileSharingProtocols property includes SMB. The default value for this property is false.
DefaultAccessPrivileges [{}]	array *read-only*	The value of this property shall be an array containing entries for the default access privileges for the file share. Each entry shall specify a defaul access privilege. The types of default access can include Read, Write, and/or Execute.
Description	null *read-write*	
EthernetInterfaces	 *read-only*	The value shall be a link to an EthernetInterfaceCollection with members that provide access to the file share.
ExecuteSupport	boolean, null *read-only*	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, null *read-only*	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.
FileShareQuotaType	null *read-write*	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.
FileShareRemainingQuotaBytes	number, null (By) *read-write*	If present, the value of this property shall indicate the remaining number of bytes that may be consumed by this file share.
FileShareTotalQuotaBytes	number, null (By) *read-write*	If present, the value of this property shall indicate the maximum number of bytes that may be consumed by this file share.
FileSharingProtocols [{}]	array *read-only*	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.

Id	*read-write*	
Links {	object *read-only*	This property shall contain links to other resources that are related to this resource.
ClassOfService {}	object, null *read-write*	This value shall be a link to the ClassOfService for this file share.
FileSystem {}	object, null *read-write*	The value shall be a link to the file system containing the file share.
Oem	*read-write*	This object represents the Oem property. All values for resources described by this schema shall comply to the requirements as described in the Redfish specification.
}		
LowSpaceWarningThresholdPercents [{}]	array *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_THRESHOLD_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(ConsumedBytes))/SUM(AllocatedBytes)
Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.
RootAccess	boolean, null *read-only*	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	null *read-write*	This value of this property shall indicate the status of the file share.

WritePolicy	string, null *read-write*	The value of this property shall define how writes are replicated to the shared source. *See Property Details, below, for more information about this property.*

8.10.1 Property Details

8.10.1.1 WritePolicy:

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.

8.11 FileSystem 1.0.0

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

AccessCapabilities [{}]	array *read-write*	This property shall be an array containing entries for the supported IO access capabilities. Each entry shall specify a current storage access capability.
BlockSizeBytes	number, null (By) *read-only*	The value of this property shall be the block size of the file system in bytes.
Capacity {	object, null *read-write*	The value of this property shall be the capacity allocated to the file system in bytes.
Data {}	object, null *read-write*	The value shall be capacity information relating to provisioned user data.

IsThinProvisioned	boolean, null *read-write*	If the value is false, the capacity shall be fully allocated. The default value shall be false.
Metadata {}	object, null *read-write*	The value shall be capacity information relating to provisioned system (non-user accessible) data.
Snapshot {}	object, null *read-write*	The value shall be capacity information relating to provisioned snapshot or backup data.
}		
CapacitySources [{	array *read-write*	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.
ProvidedCapacity {}	object, null *read-write*	The value shall be the amount of space that has been provided from the ProvidingDrives, ProvidingVolumes or ProvidingPools.
ProvidedClassOfService {}	object, null *read-write*	The value shall reference the provided ClassOfService from the ProvidingDrives, ProvidingVolumes or ProvidingPools.
ProvidingDrives {}	object, null *read-write*	The value shall be a reference to a contributing drive or drives.
ProvidingPools {}	object, null *read-write*	The value shall be a reference to a contributing storage pool or storage pools.
ProvidingVolumes {}	object, null *read-write*	The value shall be a reference to a contributing volume or volumes.
}]		

CasePreserved	boolean, null *read-write*	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.
CaseSensitive	boolean, null *read-write*	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.
CharacterCodeSet [{}]	array *read-write*	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.
ClusterSizeBytes	number, null (By) *read-write*	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	null *read-write*	
ExportedShares [{	array *read-only*	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.
CASupported	boolean, null *read-write*	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 FileSharingProtocols property includes SMB. The default value for this property is false.
DefaultAccessPrivileges [{}]	array *read-only*	The value of this property shall be an array containing entries for the default access privileges for the file share. Each entry shall specify a defaul access privilege. The types of default access can include Read, Write, and/or Execute.
Description	null *read-write*	
EthernetInterfaces	 *read-only*	The value shall be a link to an EthernetInterfaceCollection with members that provide access to the file share.

ExecuteSupport	boolean, null *read-only*	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, null *read-only*	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.
FileShareQuotaType	null *read-write*	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.
FileShareRemainingQuotaBytes	number, null (By) *read-write*	If present, the value of this property shall indicate the remaining number of bytes that may be consumed by this file share.
FileShareTotalQuotaBytes	number, null (By) *read-write*	If present, the value of this property shall indicate the maximum number of bytes that may be consumed by this file share.
FileSharingProtocols [{}]	array *read-only*	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.
Id	 *read-write*	
Links {}	object *read-only*	This property shall contain links to other resources that are related to this resource.
LowSpaceWarningThresholdPercents [{}]	array *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_THRESHOLD_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(ConsumedBytes))/SUM(AllocatedBytes)

Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.
RootAccess	boolean, null *read-only*	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	null *read-write*	This value of this property shall indicate the status of the file share.
WritePolicy	string, null *read-write*	The value of this property shall define how writes are replicated to the shared source. *See Property Details, below, for more information about this property.*
}]		
Id	*read-write*	
Links {	object *read-only*	This property shall contain links to other resources that are related to this resource.
ClassOfService {}	object, null *read-write*	This value shall be a link to the ClassOfService for this file system.
Oem	*read-write*	This object represents the Oem property. All values for resources described by this schema shall comply to the requirements as described in the Redfish specification.
ReplicaCollection [{}]	array *read-only*	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.
}		

LowSpaceWarningThresholdPercents [{}]	array *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_THRESHOLD_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(ConsumedBytes))/SUM(AllocatedBytes)
MaxFileNameLengthBytes	number, null (By) *read-write*	If specified, this value shall specify the maximum length of a file name within the file system.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
RemainingCapacity {	object, null *read-write*	The value of this property shall be the remaining capacity allocated to the file system in bytes.
Data {}	object, null *read-write*	The value shall be capacity information relating to provisioned user data.
IsThinProvisioned	boolean, null *read-write*	If the value is false, the capacity shall be fully allocated. The default value shall be false.
Metadata {}	object, null *read-write*	The value shall be capacity information relating to provisioned system (non-user accessible) data.
Snapshot {}	object, null *read-write*	The value shall be capacity information relating to provisioned snapshot or backup data.

}		
ReplicaInfo {	object, null *read-write*	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.
ConsistencyEnabled	boolean, null *read-only*	If true, consistency shall be enabled across the source and its associated target replica(s). The default value for this property is false.
ConsistencyState	string, null *read-write*	The ConsistencyState enumeration literal shall indicate the current state of consistency. *See Property Details, below, for more information about this property.*
ConsistencyStatus	string, null *read-write*	The ConsistencyStatus enumeration literal shall specify the current status of consistency. Consistency may have been disabled or is experiencing an error condition. *See Property Details, below, for more information about this property.*
ConsistencyType	string, null *read-write*	The ConsistencyType enumeration literal shall indicate the consistency type used by the source and its associated target group. *See Property Details, below, for more information about this property.*
FailedCopyStopsHostIO	boolean, null *read-only*	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.
PercentSynced	number, null (%) *read-only*	Specifies the percent of the work completed to reach synchronization. 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	null *read-write*	The value shall reference the resource that is the source of this replica.
ReplicaPriority	string, null *read-write*	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. *See Property Details, below, for more information about this property.*

ReplicaProgressStatus	string, null *read-write*	The ReplicaProgressStatus enumeration literal shall specify the status of the session with respect to Replication activity. *See Property Details, below, for more information about this property.*
ReplicaReadOnlyAccess	string, null *read-write*	The enumeration literal shall specify whether the source, the target, or both elements are read only to the host. *See Property Details, below, for more information about this property.*
ReplicaRecoveryMode	string, null *read-write*	The enumeration literal shall specify whether the copy operation continues after a broken link is restored. *See Property Details, below, for more information about this property.*
ReplicaRole	string, null *read-write*	The ReplicaRole enumeration literal shall represent the source or target role of this replica as known to the containing resource. *See Property Details, below, for more information about this property.*
ReplicaSkewBytes	number, null (By) *read-only*	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, ReplicaUpdateMode shall be switched to synchronous.
ReplicaState	string, null *read-write*	The ReplicaState enumeration literal shall specify the state of the relationship with respect to Replication activity. *See Property Details, below, for more information about this property.*
ReplicaType	string, null *read-write*	The ReplicaType enumeration literal shall describe the intended outcome of the replication. *See Property Details, below, for more information about this property.*
ReplicaUpdateMode	string, null *read-write*	The enumeration literal shall specify whether the target elements will be updated synchronously or asynchronously. *See Property Details, below, for more information about this property.*
RequestedReplicaState	string, null *read-write*	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. *See Property Details, below, for more information about this property.*

SyncMaintained	boolean, null *read-only*	If true, Synchronization shall be maintained. The default value for this property is false.
UndiscoveredElement	string, null *read-write*	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. *See Property Details, below, for more information about this property.*
WhenActivated	string, null (%) *read-only*	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.
WhenDeactivated	string, null (%) *read-only*	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.
WhenEstablished	string, null (%) *read-only*	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, null (%) *read-only*	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, null *read-only*	The value shall be an ISO 8601 conformant time of day that specifies when the elements were synchronized.
WhenSynchronized	string, null (%) *read-only*	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.
}		

8.11.1 Property Details

8.11.1.1 ConsistencyState:

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.

8.11.1.2 ConsistencyStatus:

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.

8.11.1.3 ConsistencyType:

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

8.11.1.4 ReplicaPriority:

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.

8.11.1.5 ReplicaProgressStatus:

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.
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.

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

8.11.1.6 ReplicaReadOnlyAccess:

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.

8.11.1.7 ReplicaRecoveryMode:

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.

8.11.1.8 ReplicaRole:

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

string	Description
Target	This enumeration literal shall indicate target element.

8.11.1.9 ReplicaState:

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.
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.
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.

8.11.1.10 ReplicaType:

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.

8.11.1.11 ReplicaUpdateMode:

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.

8.11.1.12 RequestedReplicaState:

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.
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.
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.

8.11.1.13 UndiscoveredElement:

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.

8.11.1.14 WritePolicy:

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.

8.12 FileSystemCollection

This resource shall contain a collection of references to FileSystem resource instances.

Description	null *read-write*	
Members [{	array *read-only*	This property shall contain references to the members of this FileSystem collection.
AccessCapabilities [{}]	array *read-write*	This property shall be an array containing entries for the supported IO access capabilities. Each entry shall specify a current storage access capability.
BlockSizeBytes	number, null (By) *read-only*	The value of this property shall be the block size of the file system in bytes.
Capacity {}	object, null *read-write*	The value of this property shall be the capacity allocated to the file system in bytes.
CapacitySources [{}]	array *read-write*	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.

CasePreserved	boolean, null *read-write*	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.
CaseSensitive	boolean, null *read-write*	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.
CharacterCodeSet [{}]	array *read-write*	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.
ClusterSizeBytes	number, null (By) *read-write*	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	null *read-write*	
ExportedShares [{}]	array *read-only*	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.
Id	 *read-write*	
Links {}	object *read-only*	This property shall contain links to other resources that are related to this resource.
LowSpaceWarningThresholdPercents [{}]	array *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_THRESHOLD_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(ConsumedBytes))/SUM(AllocatedBytes)

MaxFileNameLengthBytes	number, null (By) *read-write*	If specified, this value shall specify the maximum length of a file name within the file system.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
RemainingCapacity {}	object, null *read-write*	The value of this property shall be the remaining capacity allocated to the file system in bytes.
ReplicaInfo {}	object, null *read-write*	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.
}]		
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.13 HostedStorageServices

A Collection of Hosted Storage Service resource instances.

Description	null *read-write*	
Members [{	array *read-write*	The value of each member entry shall reference a StorageService resource.

Actions {}	object *read-only*	The Actions property shall contain the available actions for this resource.
ClassesOfService {}	object, null *read-write*	The value of each enty in the array shall reference a ClassOfService supported by this service.
ClientEndpointGroups	null *read-write*	The value of each entry in the array shall reference an EndpointGroup.
Description	null *read-write*	
Drives {}	object *read-write*	A collection that indicates all the drives managed by this storage service.
Endpoints {}	object, null *read-write*	The value of each enty in the array shall reference an Endpoint managed by this service.
FileSystems {}	object *read-write*	An array of references to FileSystems managed by this storage service.
Id	 *read-write*	
Identifier	null *read-write*	The value identifies this resource. The value shall be unique within the managed ecosystem.
Links {}	object *read-only*	Contains links to other resources that are related to this resource.
Name	 *read-write*	

Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Redundancy [{}]	array *read-only*	Redundancy information for the storage subsystem
ServerEndpointGroups	null *read-write*	The value of each entry in the array shall reference a EndpointGroup.
Status	null *read-write*	
StorageGroups {}	object, null *read-write*	The value of each enty in the array shall reference a StorageGroup.
StoragePools {}	object *read-write*	An array of references to StoragePools.
Volumes {}	object *read-write*	An array of references to Volumes managed by this storage service.
}]		
Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.14 IOConnectivityLoSCapabilities 1.0.0

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

Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value identifies this resource. The value shall be unique within the managed ecosystem.
MaxSupportedIOPS	number, null *read-write*	The value shall be the maximum IOPS that a connection can support.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
SupportedAccessProtocols [{}]	array *read-write*	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*'}).
SupportedIOConnectivityLinesOfService [{}]	array *read-write*	The collection shall contain known and supported IOConnectivityLinesOfService.

8.15 IOPerformanceLoSCapabilities 1.0.0

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

Description	null *read-write*	

IOLimitingIsSupported	boolean, null *read-write*	If true, the system should limit IOPS to MaxIOOperationsPerSecondPerTerabyte * (Volume Size in Terabytes). Otherwise, the system shall not inforce a limit. The default value for this property is false.
Id	 *read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.
MaxSamplePeriod	string, null (s) *read-write*	The value shall be an ISO 8601 duration specifying the maximum sampling period over which average values are calculated.
MinSamplePeriod	string, null (s) *read-write*	The value shall be an ISO 8601 duration specifying the minimum sampling period over which average values are calculated.
MinSupportedIoOperationLatencyMicroseconds	number, null (us) *read-write*	The value shall be the minimum supported average IO latency in microseconds calculated over the SamplePeriod
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
SupportedIOPerformanceLinesOfService [{}]	array *read-write*	The value shall be a collection supported IO performance service options.
SupportedIOWorkloads [{}]	array *read-write*	The value shall be a collection of supported workloads.

8.16 StorageGroup 1.0.0

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.

AccessState	null *read-write*	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.
Actions {	object *read-only*	The Actions property shall contain the available actions for this resource.
#StorageGroup.v1_0_0.ExposeVolumes {}	object *read-write*	Exposes the storage of this group via the target endpoints named in the ServerEndpointGroups to the initiator endpoints named in the ClientEndpointGroups. The property VolumesAreExposed shall be set to true when this action is completed.
#StorageGroup.v1_0_0.HideVolumes {}	object *read-write*	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.
Oem {}	object *read-write*	
}		
ClientEndpointGroups	null *read-write*	An array of references to ClientEndpointGroups that contain the Endpoints that may be used by clients to make requests to the storage exposed by this StorageGroup.
Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.

Links {	object *read-only*	This structure shall contain references to resources that are not contained within this resource.
ChildStorageGroups [{}]	array *read-write*	An array of references to StorageGroups are incorporated into this StorageGroup
ClassOfService {}	object, null *read-write*	The ClassOfService that all storage in this StorageGroup conforms to.
Oem	 *read-write*	This object represents the Oem property. All values for resources described by this schema shall comply to the requirements as described in the Redfish specification.
ParentStorageGroups [{}]	array *read-only*	An array of references to StorageGroups that incorporate this StorageGroup
}		
MembersAreConsistent	boolean, null *read-only*	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	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
ReplicaInfos [{	array *read-only*	This property shall describe the replication relationship between this storage group and a corresponding source and/or target storage group.
ConsistencyEnabled	boolean, null *read-only*	If true, consistency shall be enabled across the source and its associated target replica(s). The default value for this property is false.

ConsistencyState	string, null *read-write*	The ConsistencyState enumeration literal shall indicate the current state of consistency. *See Property Details, below, for more information about this property.*
ConsistencyStatus	string, null *read-write*	The ConsistencyStatus enumeration literal shall specify the current status of consistency. Consistency may have been disabled or is experiencing an error condition. *See Property Details, below, for more information about this property.*
ConsistencyType	string, null *read-write*	The ConsistencyType enumeration literal shall indicate the consistency type used by the source and its associated target group. *See Property Details, below, for more information about this property.*
FailedCopyStopsHostIO	boolean, null *read-only*	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.
PercentSynced	number, null (%) *read-only*	Specifies the percent of the work completed to reach synchronization. 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	null *read-write*	The value shall reference the resource that is the source of this replica.
ReplicaPriority	string, null *read-write*	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. *See Property Details, below, for more information about this property.*
ReplicaProgressStatus	string, null *read-write*	The ReplicaProgressStatus enumeration literal shall specify the status of the session with respect to Replication activity. *See Property Details, below, for more information about this property.*
ReplicaReadOnlyAccess	string, null *read-write*	The enumeration literal shall specify whether the source, the target, or both elements are read only to the host. *See Property Details, below, for more information about this property.*

ReplicaRecoveryMode	string, null *read-write*	The enumeration literal shall specify whether the copy operation continues after a broken link is restored. *See Property Details, below, for more information about this property.*
ReplicaRole	string, null *read-write*	The ReplicaRole enumeration literal shall represent the source or target role of this replica as known to the containing resource. *See Property Details, below, for more information about this property.*
ReplicaSkewBytes	number, null (By) *read-only*	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, ReplicaUpdateMode shall be switched to synchronous.
ReplicaState	string, null *read-write*	The ReplicaState enumeration literal shall specify the state of the relationship with respect to Replication activity. *See Property Details, below, for more information about this property.*
ReplicaType	string, null *read-write*	The ReplicaType enumeration literal shall describe the intended outcome of the replication. *See Property Details, below, for more information about this property.*
ReplicaUpdateMode	string, null *read-write*	The enumeration literal shall specify whether the target elements will be updated synchronously or asynchronously. *See Property Details, below, for more information about this property.*
RequestedReplicaState	string, null *read-write*	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. *See Property Details, below, for more information about this property.*
SyncMaintained	boolean, null *read-only*	If true, Synchronization shall be maintained. The default value for this property is false.
UndiscoveredElement	string, null *read-write*	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. *See Property Details, below, for more information about this property.*

WhenActivated	string, null (%) *read-only*	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.
WhenDeactivated	string, null (%) *read-only*	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.
WhenEstablished	string, null (%) *read-only*	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, null (%) *read-only*	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, null *read-only*	The value shall be an ISO 8601 conformant time of day that specifies when the elements were synchronized.
WhenSynchronized	string, null (%) *read-only*	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.
}]		
ServerEndpointGroups	null *read-write*	An array of references to ServerEndpointGroups that contain the Endpoints that may be used by the storage service to receive requests from clients for storage exposed by this StorageGroup.
Status	null *read-write*	

Volumes {	object, null *read-write*	An array of references to Volumes managed by this StorageGroup.
Description	null *read-write*	
Members [{}]	array *read-write*	The value of each member entry shall reference a Volume resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
VolumesAreExposed	boolean, null *read-only*	The value of this property shall be set to true if storage volumes are exposed to the initiator endpoints. The default value for this property is false.

8.16.1 Property Details

8.16.1.1 ConsistencyState:

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.

8.16.1.2 ConsistencyStatus:

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.

8.16.1.3 ConsistencyType:

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

8.16.1.4 ReplicaPriority:

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.

8.16.1.5 ReplicaProgressStatus:

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.
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.
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.

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

8.16.1.6 ReplicaReadOnlyAccess:

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.

8.16.1.7 ReplicaRecoveryMode:

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.

8.16.1.8 ReplicaRole:

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

8.16.1.9 ReplicaState:

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.
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.

8.16.1.10 ReplicaType:

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.

8.16.1.11 ReplicaUpdateMode:

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.

8.16.1.12 RequestedReplicaState:

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.

string	Description
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.
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.
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.

8.16.1.13 UndiscoveredElement:

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.

8.17 StorageGroupCollection

This collection shall contain references to all StorageGroup resource instances sharing the same parent resource.

Description	null *read-write*	

Members [{	array *read-only*	The value of each member entry shall reference a StorageGroup resource.
AccessState	null *read-write*	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.
Actions {}	object *read-only*	The Actions property shall contain the available actions for this resource.
ClientEndpointGroups	null *read-write*	An array of references to ClientEndpointGroups that contain the Endpoints that may be used by clients to make requests to the storage exposed by this StorageGroup.
Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.
Links {}	object *read-only*	This structure shall contain references to resources that are not contained within this resource.
MembersAreConsistent	boolean, null *read-only*	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	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.

ReplicaInfos [{}]	array *read-only*	This property shall describe the replication relationship between this storage group and a corresponding source and/or target storage group.
ServerEndpointGroups	null *read-write*	An array of references to ServerEndpointGroups that contain the Endpoints that may be used by the storage service to receive requests from clients for storage exposed by this StorageGroup.
Status	null *read-write*	
Volumes {}	object, null *read-write*	An array of references to Volumes managed by this StorageGroup.
VolumesAreExposed	boolean, null *read-only*	The value of this property shall be set to true if storage volumes are exposed to the initiator endpoints. The default value for this property is false.
}]		
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.18 StoragePool 1.0.0

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.

AllocatedPools {	object, null *read-write*	The value of this property shall contain a reference to the collection of storage pools allocated from this storage pool.
Description	null *read-write*	

Members [{}]	array *read-only*	The value of each member entry shall reference a StoragePool resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
AllocatedVolumes {	object, null *read-write*	The value of this property shall contain a reference to the collection of volumes allocated from this storage pool.
Description	null *read-write*	
Members [{}]	array *read-write*	The value of each member entry shall reference a Volume resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
BlockSizeBytes	number, null (By) *read-only*	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.
Capacity {	object, null *read-write*	The value of this property shall provide an information about the actual utilization of the capacity within this storage pool.

Data {}	object, null *read-write*	The value shall be capacity information relating to provisioned user data.
IsThinProvisioned	boolean, null *read-write*	If the value is false, the capacity shall be fully allocated. The default value shall be false.
Metadata {}	object, null *read-write*	The value shall be capacity information relating to provisioned system (non-user accessible) data.
Snapshot {}	object, null *read-write*	The value shall be capacity information relating to provisioned snapshot or backup data.
}		
CapacitySources [{	array *read-only*	Fully or partially consumed storage from a source resource. Each entry shall provide capacity allocation data from a named source resource.
ProvidedCapacity {}	object, null *read-write*	The value shall be the amount of space that has been provided from the ProvidingDrives, ProvidingVolumes or ProvidingPools.
ProvidedClassOfService {}	object, null *read-write*	The value shall reference the provided ClassOfService from the ProvidingDrives, ProvidingVolumes or ProvidingPools.
ProvidingDrives {}	object, null *read-write*	The value shall be a reference to a contributing drive or drives.
ProvidingPools {}	object, null *read-write*	The value shall be a reference to a contributing storage pool or storage pools.

ProvidingVolumes {}	object, null *read-write*	The value shall be a reference to a contributing volume or volumes.
}]		
ClassesOfService {	object, null *read-write*	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.
Description	null *read-write*	
Members [{}]	array *read-only*	The value of each member entry shall reference a ClassOfService resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value identifies this resource. The value shall be unique within the managed ecosystem.
Links {	object *read-only*	This structure shall contain references to resources that are not contained within this resource.

DefaultClassOfService {}	object, null *read-write*	If present, this property shall reference the default class of service for entities allocated from this storage pool. If not present, the default class of service of the containing StorageService entity shall be used.
Oem	 *read-write*	This object represents the Oem property. All values for resources described by this schema shall comply to the requirements as described in the Redfish specification.
}		
LowSpaceWarningThresholdPercents [{}]	array *read-write*	Each time the following value is less than one of the values in the array the LOW_SPACE_THRESHOLD_WARNING event shall be triggered: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(ConsumedBytes))/SUM(AllocatedBytes).
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Status	null *read-write*	

8.19 StoragePoolCollection

This collection shall contain references to all StoragePool resource instances sharing the same parent resource.

Description	null *read-write*	
Members [{	array *read-only*	The value of each member entry shall reference a StoragePool resource.
AllocatedPools {}	object, null *read-write*	The value of this property shall contain a reference to the collection of storage pools allocated from this storage pool.

AllocatedVolumes {}	object, null *read-write*	The value of this property shall contain a reference to the collection of volumes allocated from this storage pool.
BlockSizeBytes	number, null (By) *read-only*	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.
Capacity {}	object, null *read-write*	The value of this property shall provide an information about the actual utilization of the capacity within this storage pool.
CapacitySources [{}]	array *read-only*	Fully or partially consumed storage from a source resource. Each entry shall provide capacity allocation data from a named source resource.
ClassesOfService {}	object, null *read-write*	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.
Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value identifies this resource. The value shall be unique within the managed ecosystem.
Links {}	object *read-only*	This structure shall contain references to resources that are not contained within this resource.
LowSpaceWarningThresholdPercents [{}]	array *read-write*	Each time the following value is less than one of the values in the array the LOW_SPACE_THRESHOLD_WARNING event shall be triggered: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(ConsumedBytes))/SUM(AllocatedBytes).

Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Status	null *read-write*	
}]		
Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.20 StorageService 1.0.0

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.

Actions {	object *read-only*	The Actions property shall contain the available actions for this resource.
#StorageService.SetEncryptionKey {}	object *read-write*	This defines the name of the custom action supported on this resource.
Oem {}	object *read-write*	
}		
ClassesOfService {	object, null *read-write*	The value of each enty in the array shall reference a ClassOfService supported by this service.

Description	null *read-write*	
Members [{}]	array *read-only*	The value of each member entry shall reference a ClassOfService resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
ClientEndpointGroups	null *read-write*	The value of each entry in the array shall reference an EndpointGroup.
Description	null *read-write*	
Drives {	object *read-write*	A collection that indicates all the drives managed by this storage service.
Description	null *read-write*	
Members [{}]	array *read-only*	The value of each entry of this property shall reference a Drive resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		

Endpoints {	object, null *read-write*	The value of each enty in the array shall reference an Endpoint managed by this service.
Description	null *read-write*	
Members [{}]	array *read-only*	The value of each member entry shall reference an Endpoint resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
FileSystems {	object *read-write*	An array of references to FileSystems managed by this storage service.
Description	null *read-write*	
Members [{}]	array *read-only*	This property shall contain references to the members of this FileSystem collection.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
Id	 *read-write*	

Identifier	null *read-write*	The value identifies this resource. The value shall be unique within the managed ecosystem.
Links {	object *read-only*	Contains links to other resources that are related to this resource.
DataProtectionLoSCapabilities {}	object, null *read-write*	The value shall reference the data protection capabilities of this service.
DataSecurityLoSCapabilities {}	object, null *read-write*	The value shall reference the data security capabilities of this service.
DataStorageLoSCapabilities {}	object, null *read-write*	The value shall reference the data storage capabilities of this service.
DefaultClassOfService {}	object, null *read-write*	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.
HostingSystem	null *read-write*	The value shall reference the ComputerSystem that hosts this service.
IOConnectivityLoSCapabilities {}	object, null *read-write*	The value shall reference the IO connectivity capabilities of this service.
IOPerformanceLoSCapabilities {}	object, null *read-write*	The value shall reference the IO performance capabilities of this service.
Oem	 *read-write*	This object represents the Oem property. All values for resources described by this schema shall comply to the requirements as described in the Redfish specification.
}		

Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Redundancy [{}]	array *read-only*	Redundancy information for the storage subsystem
ServerEndpointGroups	null *read-write*	The value of each entry in the array shall reference a EndpointGroup.
Status	null *read-write*	
StorageGroups {	object, null *read-write*	The value of each enty in the array shall reference a StorageGroup.
Description	null *read-write*	
Members [{}]	array *read-only*	The value of each member entry shall reference a StorageGroup resource.
Name	*read-write*	
Oem	*read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
StoragePools {	object *read-write*	An array of references to StoragePools.

Description	null *read-write*	
Members [{}]	array *read-only*	The value of each member entry shall reference a StoragePool resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		
Volumes {	object *read-write*	An array of references to Volumes managed by this storage service.
Description	null *read-write*	
Members [{}]	array *read-write*	The value of each member entry shall reference a Volume resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
}		

8.21 StorageServiceCollection

An instance of this resource shall reference the set of StorageService resources known in the scope of its use.

Description	null *read-write*	
Members [{	array *read-only*	The value of each member entry shall reference a StorageService resource.
Actions {}	object *read-only*	The Actions property shall contain the available actions for this resource.
ClassesOfService {}	object, null *read-write*	The value of each enty in the array shall reference a ClassOfService supported by this service.
ClientEndpointGroups	null *read-write*	The value of each entry in the array shall reference an EndpointGroup.
Description	null *read-write*	
Drives {}	object *read-write*	A collection that indicates all the drives managed by this storage service.
Endpoints {}	object, null *read-write*	The value of each enty in the array shall reference an Endpoint managed by this service.
FileSystems {}	object *read-write*	An array of references to FileSystems managed by this storage service.
Id	 *read-write*	
Identifier	null *read-write*	The value identifies this resource. The value shall be unique within the managed ecosystem.

Links {}	object *read-only*	Contains links to other resources that are related to this resource.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Redundancy [{}]	array *read-only*	Redundancy information for the storage subsystem
ServerEndpointGroups	null *read-write*	The value of each entry in the array shall reference a EndpointGroup.
Status	null *read-write*	
StorageGroups {}	object, null *read-write*	The value of each enty in the array shall reference a StorageGroup.
StoragePools {}	object *read-write*	An array of references to StoragePools.
Volumes {}	object *read-write*	An array of references to Volumes managed by this storage service.
}]		
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.22 StorageSystemCollection

An instance of this resource shall reference the set of ComputerSystem resources known in the scope of its use and and that has a HostingRoles entry with a value of 'StorageServer'.

Description	null *read-write*	
Members [{}]	array *read-only*	The value of each member entry shall reference a ComputerSystem resource that shall have a HostingRoles entry with a value of 'StorageServer'.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.23 Volume 1.1.0

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

AccessCapabilities *(v1.1+)* [{}]	array *read-write*	Each entry shall specify a current storage access capability.
Actions {	object *read-only*	The Actions property shall contain the available actions for this resource.
#Volume.Initialize {}	object *read-write*	This defines the name of the custom action supported on this resource.
Oem {}	object *read-write*	
}		

AllocatedPools *(v1.1+)* [{	array *read-only*	The value of this property shall contain references to all storage pools allocated from this volume.
AllocatedPools {}	object, null *read-write*	The value of this property shall contain a reference to the collection of storage pools allocated from this storage pool.
AllocatedVolumes {}	object, null *read-write*	The value of this property shall contain a reference to the collection of volumes allocated from this storage pool.
BlockSizeBytes	number, null (By) *read-only*	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.
Capacity {}	object, null *read-write*	The value of this property shall provide an information about the actual utilization of the capacity within this storage pool.
CapacitySources [{}]	array *read-only*	Fully or partially consumed storage from a source resource. Each entry shall provide capacity allocation data from a named source resource.
ClassesOfService {}	object, null *read-write*	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.
Description	null *read-write*	
Id	 *read-write*	
Identifier	null *read-write*	The value identifies this resource. The value shall be unique within the managed ecosystem.

Links {}	object *read-only*	This structure shall contain references to resources that are not contained within this resource.
LowSpaceWarningThresholdPercents [{}]	array *read-write*	Each time the following value is less than one of the values in the array the LOW_SPACE_THRESHOLD_WARNING event shall be triggered: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(ConsumedBytes))/SUM(AllocatedBytes).
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Status	null *read-write*	
}]		
BlockSizeBytes	number, null (By) *read-only*	This property shall contain size of the smallest addressable unit of the associated volume.
Capacity *(v1.1+)* {	object, null *read-write*	Information about the utilization of capacity allocated to this storage volume.
Data {}	object, null *read-write*	The value shall be capacity information relating to provisioned user data.
IsThinProvisioned	boolean, null *read-write*	If the value is false, the capacity shall be fully allocated. The default value shall be false.

Metadata {}	object, null *read-write*	The value shall be capacity information relating to provisioned system (non-user accessible) data.
Snapshot {}	object, null *read-write*	The value shall be capacity information relating to provisioned snapshot or backup data.
}		
CapacityBytes	number, null (By) *read-only*	This property shall contain the size in bytes of the associated volume.
CapacitySources *(v1.1+)* [{	array *read-write*	Fully or partially consumed storage from a source resource. Each entry provides capacity allocation information from a named source resource.
ProvidedCapacity {}	object, null *read-write*	The value shall be the amount of space that has been provided from the ProvidingDrives, ProvidingVolumes or ProvidingPools.
ProvidedClassOfService {}	object, null *read-write*	The value shall reference the provided ClassOfService from the ProvidingDrives, ProvidingVolumes or ProvidingPools.
ProvidingDrives {}	object, null *read-write*	The value shall be a reference to a contributing drive or drives.
ProvidingPools {}	object, null *read-write*	The value shall be a reference to a contributing storage pool or storage pools.
ProvidingVolumes {}	object, null *read-write*	The value shall be a reference to a contributing volume or volumes.
}]		

Description	null *read-write*	
Encrypted	boolean, null *read-write*	This property shall contain a boolean indicator if the Volume is currently utilizing encryption or not.
EncryptionTypes [{}]	array *read-write*	This property shall contain the types of encryption used by this Volume.
Id	 *read-write*	
Identifiers [{}]	array *read-only*	This property shall contain a list of all known durable names for the associated volume.
Links {	object *read-only*	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.
ClassOfService {}	object, null *read-write*	This property shall contain a reference to the ClassOfService that this storage volume conforms to.
Drives [{}]	array *read-only*	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.
Oem	 *read-write*	This object represents the Oem property. All values for resources described by this schema shall comply to the requirements as described in the Redfish specification.
}		
LowSpaceWarningThresholdPercents *(v1.1+)* [{}]	array *read-write*	Each time the following value is less than one of the values in the array the LOW_SPACE_THRESHOLD_WARNING event shall be triggered: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(ConsumedBytes))/SUM(AllocatedBytes)

Manufacturer *(v1.1+)*	string, null *read-only*	This property shall contain a value that represents the manufacturer or implementer of the storage volume.
MaxBlockSizeBytes *(v1.1+)*	number, null (By) *read-only*	This property shall contain size of the largest addressable unit of this storage volume.
Model *(v1.1+)*	string, null *read-only*	The value is assigned by the manufacturer and shall represents a specific storage volume implementation.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Operations [{	array *read-only*	This property shall contain a list of all currently running on the Volume.
AssociatedTask	 *read-only*	A reference to the task associated with the operation if any.
OperationName	string, null *read-only*	The name of the operation.
PercentageComplete	number, null *read-only*	The percentage of the operation that has been completed.
}]		

OptimumIOSizeBytes	number, null (By) *read-only*	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.
ReplicaInfos *(v1.1+)* [{	array *read-only*	This property shall describe the replica relationship between this storage volume and a corresponding source and/or target volume.
ConsistencyEnabled	boolean, null *read-only*	If true, consistency shall be enabled across the source and its associated target replica(s). The default value for this property is false.
ConsistencyState	string, null *read-write*	The ConsistencyState enumeration literal shall indicate the current state of consistency. *See Property Details, below, for more information about this property.*
ConsistencyStatus	string, null *read-write*	The ConsistencyStatus enumeration literal shall specify the current status of consistency. Consistency may have been disabled or is experiencing an error condition. *See Property Details, below, for more information about this property.*
ConsistencyType	string, null *read-write*	The ConsistencyType enumeration literal shall indicate the consistency type used by the source and its associated target group. *See Property Details, below, for more information about this property.*
FailedCopyStopsHostIO	boolean, null *read-only*	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.
PercentSynced	number, null (%) *read-only*	Specifies the percent of the work completed to reach synchronization. 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	null *read-write*	The value shall reference the resource that is the source of this replica.

ReplicaPriority	string, null *read-write*	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. *See Property Details, below, for more information about this property.*
ReplicaProgressStatus	string, null *read-write*	The ReplicaProgressStatus enumeration literal shall specify the status of the session with respect to Replication activity. *See Property Details, below, for more information about this property.*
ReplicaReadOnlyAccess	string, null *read-write*	The enumeration literal shall specify whether the source, the target, or both elements are read only to the host. *See Property Details, below, for more information about this property.*
ReplicaRecoveryMode	string, null *read-write*	The enumeration literal shall specify whether the copy operation continues after a broken link is restored. *See Property Details, below, for more information about this property.*
ReplicaRole	string, null *read-write*	The ReplicaRole enumeration literal shall represent the source or target role of this replica as known to the containing resource. *See Property Details, below, for more information about this property.*
ReplicaSkewBytes	number, null (By) *read-only*	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, ReplicaUpdateMode shall be switched to synchronous.
ReplicaState	string, null *read-write*	The ReplicaState enumeration literal shall specify the state of the relationship with respect to Replication activity. *See Property Details, below, for more information about this property.*
ReplicaType	string, null *read-write*	The ReplicaType enumeration literal shall describe the intended outcome of the replication. *See Property Details, below, for more information about this property.*
ReplicaUpdateMode	string, null *read-write*	The enumeration literal shall specify whether the target elements will be updated synchronously or asynchronously. *See Property Details, below, for more information about this property.*

RequestedReplicaState	string, null *read-write*	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. *See Property Details, below, for more information about this property.*
SyncMaintained	boolean, null *read-only*	If true, Synchronization shall be maintained. The default value for this property is false.
UndiscoveredElement	string, null *read-write*	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. *See Property Details, below, for more information about this property.*
WhenActivated	string, null (%) *read-only*	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.
WhenDeactivated	string, null (%) *read-only*	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.
WhenEstablished	string, null (%) *read-only*	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, null (%) *read-only*	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, null *read-only*	The value shall be an ISO 8601 conformant time of day that specifies when the elements were synchronized.

WhenSynchronized	string, null (%) *read-only*	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.
}]		
Status	*read-write*	
StorageGroups *(v1.1+)* [{	array *read-only*	The value of this property shall contain references to all storage groups that include this volume.
AccessState	null *read-write*	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.
Actions {}	object *read-only*	The Actions property shall contain the available actions for this resource.
ClientEndpointGroups	null *read-write*	An array of references to ClientEndpointGroups that contain the Endpoints that may be used by clients to make requests to the storage exposed by this StorageGroup.
Description	null *read-write*	
Id	*read-write*	
Identifier	null *read-write*	The value shall be unique within the managed ecosystem.
Links {}	object *read-only*	This structure shall contain references to resources that are not contained within this resource.

MembersAreConsistent	boolean, null *read-only*	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	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
ReplicaInfos [{}]	array *read-only*	This property shall describe the replication relationship between this storage group and a corresponding source and/or target storage group.
ServerEndpointGroups	null *read-write*	An array of references to ServerEndpointGroups that contain the Endpoints that may be used by the storage service to receive requests from clients for storage exposed by this StorageGroup.
Status	null *read-write*	
Volumes {}	object, null *read-write*	An array of references to Volumes managed by this StorageGroup.
VolumesAreExposed	boolean, null *read-only*	The value of this property shall be set to true if storage volumes are exposed to the initiator endpoints. The default value for this property is false.
}]		
VolumeType	string, null *read-write*	This property shall contain the type of the associated Volume. *See Property Details, below, for more information about this property.*

8.23.1 Property Details

8.23.1.1 ConsistencyState:

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.

8.23.1.2 ConsistencyStatus:

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.

8.23.1.3 ConsistencyType:

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

8.23.1.4 ReplicaPriority:

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.

8.23.1.5 ReplicaProgressStatus:

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.
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.

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

8.23.1.6 ReplicaReadOnlyAccess:

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.

8.23.1.7 ReplicaRecoveryMode:

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.

8.23.1.8 ReplicaRole:

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

8.23.1.9 ReplicaState:

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.
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.
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.

8.23.1.10 ReplicaType:

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.

8.23.1.11 ReplicaUpdateMode:

string	Description
Active	This enumeration literal shall indicate Active-Active (i.e. bidirectional) synchronous updates.

string	Description
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.

8.23.1.12 RequestedReplicaState:

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.
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.
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.

8.23.1.13 UndiscoveredElement:

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

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

8.23.1.14 VolumeType:

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

8.24 VolumeCollection

This collection shall contain references to all Volume resource instances sharing the same parent resource.

Description	null *read-write*	
Members [{	array *read-write*	The value of each member entry shall reference a Volume resource.
AccessCapabilities [{}]	array *read-write*	Each entry shall specify a current storage access capability.
Actions {}	object *read-only*	The Actions property shall contain the available actions for this resource.
AllocatedPools [{}]	array *read-only*	The value of this property shall contain references to all storage pools allocated from this volume.
BlockSizeBytes	number, null (By) *read-only*	This property shall contain size of the smallest addressable unit of the associated volume.

Capacity {}	object, null *read-write*	Information about the utilization of capacity allocated to this storage volume.
CapacityBytes	number, null (By) *read-only*	This property shall contain the size in bytes of the associated volume.
CapacitySources [{}]	array *read-write*	Fully or partially consumed storage from a source resource. Each entry provides capacity allocation information from a named source resource.
Description	null *read-write*	
Encrypted	boolean, null *read-write*	This property shall contain a boolean indicator if the Volume is currently utilizing encryption or not.
EncryptionTypes [{}]	array *read-write*	This property shall contain the types of encryption used by this Volume.
Id	 *read-write*	
Identifiers [{}]	array *read-only*	This property shall contain a list of all known durable names for the associated volume.
Links {}	object *read-only*	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.
LowSpaceWarningThresholdPercents [{}]	array *read-write*	Each time the following value is less than one of the values in the array the LOW_SPACE_THRESHOLD_WARNING event shall be triggered: Across all CapacitySources entries, percent = (SUM(AllocatedBytes) - SUM(ConsumedBytes))/SUM(AllocatedBytes)

Manufacturer	string, null *read-only*	This property shall contain a value that represents the manufacturer or implementer of the storage volume.
MaxBlockSizeBytes	number, null (By) *read-only*	This property shall contain size of the largest addressable unit of this storage volume.
Model	string, null *read-only*	The value is assigned by the manufacturer and shall represents a specific storage volume implementation.
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.
Operations [{}]	array *read-only*	This property shall contain a list of all currently running on the Volume.
OptimumIOSizeBytes	number, null (By) *read-only*	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.
ReplicaInfos [{}]	array *read-only*	This property shall describe the replica relationship between this storage volume and a corresponding source and/or target volume.
Status	 *read-write*	
StorageGroups [{}]	array *read-only*	The value of this property shall contain references to all storage groups that include this volume.

VolumeType	string, null *read-write*	This property shall contain the type of the associated Volume. *See Property Details, below, for more information about this property.*
}]		
Name	 *read-write*	
Oem	 *read-write*	The value of this string shall be of the format for the reserved word *Oem*.

8.24.1 Property Details

8.24.1.1 VolumeType:

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