Swordfish Interoperability Guide

Version: 1.2.8

ABSTRACT:

The Swordfish Interoperability Guide provides a reference of the Swordfish-specific extensions that have been made to the profile syntax defined in the Redfish Interoperabilty profile.

Working Draft

Table of Contents

List of Tables

USAGE

the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of Storage Networking Industry Association nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

> THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, 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. SNIA makes no warranty of any kind with regard to this publication, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The SNIA shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use.

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

Current Revision

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

Contact SNIA

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

FEEDBACK AND INTERPRETATIONS

Requests for interpretation, suggestions for improvement and addenda, or defect reports are welcome. They should be sent via the SNIA Feedback Portal at http://www.snia.org/feedback/ or by mail to SNIA, 5201 Great America Parkway, Suite 320, Santa Clara, CA 95054, USA.

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.

USAGE

1. Any text, diagram, chart, table or definition reproduced must be reproduced in its entirety with no alteration, and,

2. Any document, printed or electronic, in which material from this document (or any portion hereof) is reproduced must acknowledge the SNIA copyright on that material, and must credit the SNIA for granting permission for its reuse.

Other than as explicitly provided above, you may not make any commercial use of this document, or any portion thereof, or distribute this document to third parties. All rights not explicitly granted are expressly reserved to SNIA.

Permission to use this document for purposes other than those enumerated above may be requested by emailing tcmd@snia.org. Please include the identity of the requesting individual and/or company and a brief description of the purpose, nature, and scope of the requested use.

All code fragments, scripts, data tables, and sample code in this SNIA document are made available under the following license:

BSD 3-Clause Software License

Copyright (c) 2025, The Storage Networking Industry Association.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of The Storage Networking Industry Association (SNIA) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

> THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, 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 publication, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The SNIA shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use.

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

Current Revision

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

Contact SNIA

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

FEEDBACK AND INTERPRETATIONS

Requests for interpretation, suggestions for improvement and addenda, or 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.

Revision History

The evolution of this document is summarized in Table 1.

Table 1: Revision History

Date	Rev	Notes
21 May 2024	1.2.7	Release as SNIA Working Draft
13 August 2024	1.2.7	Release as SNIA Publication
		- Errata change in FeaturesRegistry: set Resources to type Resource.Resource
28 January 2025	1.2.8	Release as Working Draft
		- Update Event profile to remove references to licensing
		- Updated CreateConsistencyGroup Use Case to use `ConsistencyGroup` and
		move to MappingMasking section
		- Update NVMeNamespaceManagement profile
		- Features Registry: Fix type of Resources collection

About SNIA

SNIA is a not-for-profit global organization made up of corporations, universities, startups, and individuals. The members collaborate to develop and promote vendor-neutral architectures, standards,

and education for management, movement, and security for technologies related to handling and optimizing data. SNIA focuses on the transport, storage, acceleration, format, protection, and optimization of infrastructure for data. Learn more at www.snia.org.

Acknowledgements

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

Table 2: Contributors

Member	Representatives (* – prior employer)
Intel Corporation	Richelle Ahlvers

1 Introduction

1.1 Overview

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

In addition to the data model and API, Swordfish has developed a suite of supported functionality into profiles and grouped them in to supported features. These profiles form the basis for a robust and consistent assessment of an implementation’s conformance to the Swordfish specification, and its ability to reliably interoperate with other Swordfish-compliant tools and systems, while the features provide a common basis to advertise functionality in run-time, using high-level and easy to consume descriptions for both humans and automation.

As with many other elements in the Swordfish ecosystem, the Swordfish profiles definition leverage and extend the Redfish profile definition, defined in the RedfishInteroperabilityProfile schema.

This document:

- defines the nomenclature and usage of Swordfish Features and Profiles
- provides an overview of the RedfishInteroperabilityProfile schema and definition, and
- highlights the extensions Swordfish has made to the RedfishInteroperabilityProfile schema within the SwordfishInteroperabilityProfile schema
- defines the current set of features supported by Swordfish
- provides a detailed review of each underlying profile, including any constraints on its implementation or support.

1.2 Who should read this document?

This document will be most useful to anyone developing or deploying a Swordfish-based system. It provides a Swordfish developer with a clear definition of the profiles that have been built around the current Swordfish specification. The profile definitions provided by this document will also be useful to Swordfish client developers who are interested in developing a more complete undestanding of the profiles provided by Swordfish, and the implicaiton and requirements surrounding profile support offered by service providers.

1.3 Using this guide

This document is broken into two main sections:

- a description of features and profiles in Swordfish, including:
    - a detailed review of the underlying modeling of Swordfish Features and Profiles, and
    - a discussion of the added functionality provided by the SwordfishInteroperabilityProfile and its extension of the underlying Redfish functionality
- the details of the current set of features and profiles, including:
    - a summary of the current set of features supported by Swordfish, along with its required profile(s), and
    - a detailed review of each profile defined against the current Swordfish specification.

2 Features and Profiles

2.1 Introduction

Features and Profiles are paired constructs that serve related, but different, functions.

Features are high-level descriptions of functionality. An implementation advertises that it supports a given set of Features through a Features Registry (/redfish/v1/Registries). Features reflect functionality defined in profiles.

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

- Swordfish.Block.Provisioning
- Swordfish.File.Provisioning

Features (and the Feature Registry) are something that clients can query at runtime to determine what functionality Swordfish Implementations supports. Profiles are predominantly a tool for developers and test, used to ensure that a implementation is accurate, complete and interoperable.

2.2 Profile document definition

Like the Redfish interoperability profile, which it extends, the SwordfishInteroperability profile is specified in a JSON document. The JSON objects and properties contained in the SwordfishInteroperability profile are described in this document and the RedfishInteroperabilityProfile.v1_5_0.json from DMTF's Redfish schema repository at http://redfish.dmtf.org/profiles. The json-schema can be used to validate a profile document to ensure compatibility with automated conformance tools or utilities.

The JSON document structure is intended to align easily with JSON payloads retrieved from a Swordfish service implementations, and to allow for easy comparisons and conformance testing. Many of the properties defined within this structure have assumed default values that correspond with the most common use case, so that those properties can be omitted from the document for brevity.

As an aid to the reader, much of the material in the Redfish document is repeated here, and any changes or extensions are clearly identified.

> NB: The latest version of the Redfish Interoperability Profile includes substantial

> functionality that was developed after the creation of the Swordfish Interoperability Profile. As an organizational design tenet to keep Swordfish and Redfish in sync, this will be revisited on a periodic basis. Accordingly, Swordfish developers should take care to work from this document. The Swordfish Interoperability Profile v1_0_0 is based on the Redfish Interoperability Profile v1_5_0.

2.3 File name conventions

The document that describes a profile follows the Redfish schema file naming conventions from the Redfish Specification. The file name format for profiles shall be:

```
<ProfileName>.v<MajorVersion>_<MinorVersion>_<Errata>.json
```

For example, the file name of the BasicServer profile v1.2.0 is BasicServer.v1_2_0.json. The file name for a given profile shall include the profile name and version, and aatch those property values within the profile's JSON document.

2.4 Profile Structure

2.4.1 Basic profile properties

At the top level of the JSON document are the basic properties, which describe the profile, including authorship and contact information, versioning, and other profiles to include in order to build upon previous work. They are summarizded in Table 3.

Table 3: Basic Profile Summary

Property	Type	Description
SchemaDefinition	string	The JSON schema that defines this Redfish interoperability profile document and can be used to validate its contents.
ProfileName	string	The name of this profile.
ProfileVersion	string	The version of this Redfish profile. The version shall be represented using a .. format.
Purpose	string	A description of the purpose of this Redfish profile, such as its intended target audience, product segments, etc.
ContactInfo	string	An email address that can be

Property	Type	Description
		used to provide feedback about this Redfish profile.
OwningEntity	string	The name of the owning entity that defined this Redfish interoperability profile.
ContributedBy	string	The name of the original author or entity that contributed the content of this profile to the owning entity.
License	string	The license statement for this profile.
RequiredProfiles	object	A set of Redfish profiles that serve as a basis for this profile. The requirements set forth in these profiles are included in this profile.
Protocol	object	Requirements related to the Redfish protocol outside of the JSON resources. See the Protocol requirements clause.
Resources	object	The JSON resource requirements. See the Resource (schema) requirements clause.
Registries	object	The registry requirements. See the Registry-level requirements clause.

2.4.2 Required profiles

The following RequiredProfiles object structure, while shared with Redfish, differs in capability for Swordfish. In the Swordfish Interoperability Profile, the inter-dependence of profiles can be defined using Conditionals (see Swordfish Extensions to Interoperability Profile Schema), rather than the simple include model in Redfish.

The RequiredProfiles object contains object properties whose names match the name of the profile to be included. Each of these sub-objects contains the properties listed below in Table 4.

Table 4: Required Properties

Property	Type	Description
Repository	string	A URI providing the location of the repository that contains the JSON files to be included. The filenames of the JSON files contained in the repository are expected to follow the Redfish interoperability profile filename conventions. If absent, the repository location shall be the Redfish profile repository (`<http://redfish.dmtf.org/profiles>`).
MinVersion	string	The minimum version required by this Redfish profile. The version shall be represented using a `<major>.<minor>.<errata>` format, including an optional errata version. If this property is absent, the minimum value shall be 1.0.0.

```
"RequiredProfiles": {
  "SwordfishDiscovery": {
    "Repository": "https://redfish.dmtf.org/profiles/swordfish",
    "MinVersion": "1.1.4",
    "ReadRequirement": "None",
    "ConditionalRequirements": [{
      "Purpose": "Either SwordfishDiscovery or SwordfishCoSDiscovery must be
      supported.",
      "CompareProfile": "SwordfishCoSDiscovery",
      "CompareType": "Absent",
      "ReadRequirement": "Mandatory"
    }]
  },
  "SwordfishCoSDiscovery": {
    "Repository": "https://redfish.dmtf.org/profiles/swordfish",
    "MinVersion": "1.2.0",
    "ReadRequirement": "None",
    "ConditionalRequirements": [{
      "Purpose": "Either SwordfishDiscovery or SwordfishCoSDiscovery must be
      supported.",
      "CompareProfile": "SwordfishDiscovery",
      "CompareType": "Absent",
```

```
        "ReadRequirement": "Mandatory"
      }]
    }
```

2.4.3 Protocol requirements

The “Protocol” object in a Profile contains properties which describe the Redfish protocol version that is required for this profile.

Swordfish profiles tend to only instrument the MinVersion property of the protocol object, which defines the minimum required version of the Redfish specification and service root. There are also minimum requirements for Swordfish schema versions. They are defined by a profile’s version and included in the MinVersion of RequiredProfiles.

Property	Type	Description
MinVersion	string	Indicates the minimum version of the Redfish Specification protocol support required by this profile. This version shall be reported by the Redfish service in the ServiceRoot resource property RedfishVersion. The version shall be represented using a .. format, including an optional errata version. If this property is absent, the minimum value shall be 1.0.0.
Discovery	string	Indicates support requirements for the Redfish SSDP discovery protocol. If this property is absent, there is no requirement for SSDP. See the Requirement values clause.
HostInterface	string	Indicates support requirements for the Redfish host interface. If this property is absent, there is no requirement for a host interface. See the Requirement values clause.
ExpandQuery	string	Indicates support requirements for the $expand query parameter.

Property	Type	Description
		Additional $expand support requirements may be specified in the resource entry for the ProtocolFeaturesSupported object within ServiceRoot. If this property is absent, there is no requirement for support of the $expand query parameter. See the Requirement values clause.
FilterQuery	string	Indicates support requirements for the $filter query parameter. If this property is absent, there is no requirement for support of the $filter query parameter. See the Requirement values clause.
SelectQuery	string	Indicates support requirements for the $select query parameter. If this property is absent, there is no requirement for support of the $select query parameter. See the Requirement values clause.
OnlyQuery	string	Indicates support requirements for the only query parameter. If this property is absent, there is no requirement for support of the only query parameter. See the Requirement values clause.
ExcerptQuery	string	Indicates support requirements for the excerpt query parameter. If this property is absent, there is no requirement for support of the excerpt query parameter. See the Requirement values clause.

2.4.3.1 Example

```
{
    "Protocol": {
        "MinVersion": "1.8"
    }
}
```

2.4.4 Conditional requirements

The most flexible aspect of the Redfish profile definition is the ability to make resource or property-level requirements that are dependent on one or more conditional requirements within the resource and the parent resources in the resource tree.

Swordfish extends this capability even further to apply to profiles and registries.

The ConditionalRequirements array function specifies these conditional requirements, which add to any requirements also defined for the resource or property. Note that a condition cannot override or weaken a requirement already specified. For example, if a property requirement is marked as Mandatory, no conditional requirement could mark the property as None. Instead, the property would be specified with a None requirement, and with one or more ConditionalRequirements that would specify when the property requirement becomes Mandatory.

The following options are available for each conditional requirement:

Property	Type	Description
ReadRequirement	string	The requirement to apply to the resource or property if the condition is met.
WriteRequirement	string	Property-level write (HTTP PATCH or PUT) requirement for this property. See the Write requirement clause.
Purpose	string	Text describing the purpose of this conditional requirement.
SubordinateToResource	array	An ordered list, from top of hierarchy to bottom, of resources where this resource is linked as a subordinate resource. The conditional requirements listed for the resource apply only to instances which are subordinate to the listed parent resource list. See the Parent and subordinate resources clause.
Comparison	string	The condition used to compare the value of the property to Values. See the Comparison clause.
Values	array	The values used to perform a

Property	Type	Description
		comparison. Multiple values are only allowed for AnyOf or AllOf comparisons. If no Comparison property is present, the comparison is assumed to be an AnyOf comparison.
CompareProperty	string	The name or path to the property in this resource whose value is used to test this condition. If the value begins with a / character, the value shall represent an RFC6901-defined JSON Pointer, specifying an explicit path from the root level of the resource to a property within the resource. Otherwise, the property name will be evaluated at the current object level within the resource, and if it is not found, upper levels will be searched until the root level is reached.
CompareValues	array	Values of the CompareProperty used to test this condition.
CompareType	string	The condition used to compare the value of the property named by CompareProperty to the values of CompareValues. This property follows the same definition as the Comparison property. If the comparison is true, this conditional requirement applies.

2.4.4.1 Parent and subordinate resources

Because there can be several instances of a particular Redfish schema in the resource tree, the requirements placed on those resources may vary depending on their usage. Since the profile is schema-centric, the SubordinateToResource function allows a profile to specify requirements based a resource instance's placement in the resource tree.

SubordinateToResource allows specifying the schema (resource) path from parent resources to the resource to which the requirements apply. This property contains an array of schema names, in the top-

down order that they appear in the path to the required resource.

Example

```
"AllocatedPools": {
    "ReadRequirement": "None",
    "ConditionalRequirements": [{
        "Purpose": "Required when NVMePoolType = EnduranceGroup. If NVMSet, do not
    implement.",
        "CompareProperty": "/NVMeProperties/NVMePoolType",
        "CompareType": "Equal",
        "CompareValues": ["EnduranceGroup"],
        "Comparison": "Absent",
        "ReadRequirement": "Mandatory"
    }]
}
```

2.4.4.1.1 Compare property

A typical need for a conditional requirement is a dependency on the value of another property within the resource. This type of dependency can be used when several different product variations share a common schema definition. In that case, Redfish schemas normally define a type-specifying property with enumerations, for a variety of product categories, that can be used to differentiate profile requirements by product category.

To accomplish this, there are three Profile properties related to this function:

Property	Type	Description
CompareProperty	string	The name or path to the property in this resource whose value is used to test this condition. If the value begins with a / character, the value shall represent an RFC6901-defined JSON Pointer, specifying an explicit path from the root level of the resource to a property within the resource. Otherwise, the property name will be evaluated at the current object level within the resource, and if it is not found, upper levels will be searched until the root level is reached.
CompareType	string	The condition used to compare

Property	Type	Description
		the value of the property named by CompareProperty to the values of CompareValues. This property follows the same definition as the Comparison property. If the comparison is true, this conditional requirement applies. This property is required and shall be present for all uses of CompareProperty.
CompareValues	array	Values of the CompareProperty used to test this condition. This property shall be present for any value of CompareType except Present and Absent.

2.4.4.1.2 Examples

Simple dependencies can be expressed using the conditional requirement and a comparison. This example shows a CompareProperty condition applied to the Pepperoni property. If the PizzaType property is not equal to Cheese, then the Pepperoni property becomes both mandatory and must have a value of true.

```
{
    "Pepperoni": {
        "ReadRequirement": "Recommended",
        "ConditionalRequirements": [
            {
                "Purpose": "Pepperoni is required on all pizza types except Cheese.",
                "CompareProperty": "PizzaType",
                "CompareType": "NotEqual",
                "CompareValues": [
                    "Cheese"
                ],
                "ReadRequirement": "Mandatory",
                "Comparison": "Equal",
                "Values": [
                    true
                ]
            }
        ]
    }
}
```

This example shows a CompareProperty condition applied to the IndicatorLED property, which has a base Recommended requirement, but becomes Mandatory if the SystemType property has a value of Physical or Composed.

```
{
    "IndicatorLED": {
        "ReadRequirement": "Recommended",
        "ConditionalRequirements": [
            {
                "Purpose": "Physical and composed systems must have a writable LED",
                "CompareProperty": "SystemType",
                "CompareType": "AnyOf",
                "CompareValues": [
                    "Physical",
                    "Composed"
                ],
                "ReadRequirement": "Mandatory",
                "WriteRequirement": "Mandatory"
            }
        ]
    }
}
```

This example shows a CompareProperty condition applied to the SerialNumber property, which has a Conditional requirement, becoming Mandatory only in cases where the /Location/PartLocation/LocationType property (specified as a JSON Pointer per RFC6901) has a value that is not Embedded.

```
{
    "SerialNumber": {
        "ReadRequirement": "Conditional",
        "ConditionalRequirements": [
            {
                "Purpose": "SerialNumber is required on Memory resources whose
        LocationType is not Embedded.",
                "CompareProperty": "/Location/PartLocation/LocationType",
                "CompareType": "NotEqual",
                "CompareValues": [
                    "Embedded"
                ],
                "ReadRequirement": "Mandatory"
            }
        ]
    }
}
```

2.5 Swordfish Extensions to Interoperability Profile Schema

Swordfish extends the Redfish Interoperability profile schema in multiple areas, several of which are detailed below. A primary area of focus is around an expanded use of conditionals, allowing more expansive comparisons across large sets of resources and resource types. It also adds conditional support to profiles, as Swordfish's unique usage allows for the combinatorial effect for related features and profiles. Additionally, conditional requirements are extended to Actions, as functionality such as replication need only be instantiated in a portion of a broad-scale implementation.

2.5.1 PropertyProfile : PropertyRequirements

The SwordfishInteroperabilityProfile extends the Redfish model by extending comparisons and conditionals across resources. This allows for more complex dependencies in feature structure. For example, the underlying capacity requirements can be abstracted away from "must be disk-based" to allow for disk-based or memory-based capacity sources underlying storage systems.

Example

```
    "Volume": {
      "Purpose": "One of the following mechanisms is required to support reporting of
        volume capacity.  In order to claim support of the Capacity Management Feature,
        correspondingly 'write' (expand) on that property must be supported.",
      "PropertyRequirements": {
        "CapacityBytes": {
          "ReadRequirement": "Recommended",
          "ConditionalRequirements": [{
              "Purpose": "Either CapacityBytes or Capacity/#Data/#AllocatedBytes must
        be supported.",
              "CompareProperty": "/Capacity/Data/AllocatedBytes",
              "CompareType": "Absent",
              "WriteRequirement": "Mandatory"
          }]
        },
        "Capacity": {
          "PropertyRequirements": {
            "Data": {
              "PropertyRequirements": {
                "AllocatedBytes": {
                  "ConditionalRequirements": [{
                      "Purpose": "Either CapacityBytes or
        Capacity/#Data/#AllocatedBytes must be supported.",
                      "CompareProperty": "CapacityBytes",
                      "CompareType": "Absent",
                      "WriteRequirement": "Mandatory"
```

```
                }]
              }
            }
          }
        }
      }
    }
```

2.5.2 PropertyProfile : ReadRequirements

Swordfish has extended the ReadRequirements usage in an important way, adding the capability to specify a value of `DoNotImplement`, defined as "This property shall not be implemented in any instance of this resource". The Swordfish CTP test framework has also been extended to check for this usage, flagging usage with warnings.

This is used to both harden use cases, such as properties that should not be implemented for a use case, or to highlight deprecated properties in favor of alternatives.

Examples

```
  "EthernetInterface": {
    "ReadRequirement": "DoNotImplement",
    "Purpose": "This property is recommended as 'Do not implement' for this device
      and protocol type."
  },
  "PhysicalNetworkPortAssignment": {}
}
```

```
"NetworkPorts": {
  "ReadRequirement": "DoNotImplement",
  "Purpose": "Deprecated, do not implement."
},
```

2.5.3 ActionProfile : ConditionalRequirements

Swordfish also allows conditionals on actions. This enables flexibility again in large scale, complex configurations, where actions may be required in some like resources, but not all, of large configurations.

Example

```
"Volume": {
  "ActionRequirements": {
    "CreateReplicaTarget": {
      "ReadRequirement": "IfImplemented",
      "Purpose": "Ability to reset the system is a core requirement of most users.",
      "ConditionalRequirements": [{
        "Purpose": "Either CreateReplicaTarget or AssignReplicaTarget must be
       supported in order to advertise local replication support, when
       ReplicationEnabled is defined on the volume.",
        "CompareProperty": "ReplicationEnabled",
        "CompareType": "Absent",
        "ReadRequirement": "None"
      },
        {
        "Purpose": "Either CreateReplicaTarget or AssignReplicaTarget must be
       supported in order to advertise local replication support, when
       ReplicationEnabled is defined on the volume.",
        "CompareProperty": "AssignReplicaTarget",
        "CompareType": "Absent",
        "ReadRequirement": "Mandatory"
      }]
    },
    "AssignReplicaTarget": {
      "ReadRequirement": "IfImplemented",
      "Purpose": "Ability to reset the system is a core requirement of most users.",
      "ConditionalRequirements": [{
        "Purpose": "Either CreateReplicaTarget or AssignReplicaTarget must be
       supported in order to advertise local replication support, when
       ReplicationEnabled is defined on the volume.",
        "CompareProperty": "ReplicationEnabled",
        "CompareType": "Absent",
        "ReadRequirement": "None"
      },{
        "Purpose": "Either CreateReplicaTarget or AssignReplicaTarget must be
       supported in order to advertise local replication support.",
        "CompareProperty": "CreateReplicaTarget",
        "CompareType": "Absent",
        "ReadRequirement": "Mandatory"
      }]
    }
  }
}
```

2.5.4 Condition : Properties : CompareProfile

In the Swordfish Interoperability Profile, profiles support Conditionals through virtue of the addition of the CompareProfile property to `Condition`.

Example

```
"RequiredProfiles": {
  "SwordfishDiscovery": {
    "Repository": "https://redfish.dmtf.org/profiles/swordfish",
    "MinVersion": "1.1.4"
  },
  "SwordfishBlockProvisioning": {
    "Repository": "https://redfish.dmtf.org/profiles/swordfish",
    "MinVersion": "1.3.0"
  },
  "SwordfishFabricConnectivityRightsforEthernet": {
    "Repository": "https://redfish.dmtf.org/profiles/swordfish",
    "MinVersion": "1.0.2",
    "ConditionalRequirements": [{
      "Purpose": "Either SwordfishFabricConnectivityRightsforEthernet or
      SwordfishFabricConnectivityRightsforRDMA must be supported.",
      "CompareProfile": "SwordfishFabricConnectivityRightsforRDMA",
      "CompareType": "Absent",
      "ReadRequirement": "Mandatory"
    }]
  },
  "SwordfishFabricConnectivityRightsforRDMA": {
    "Repository": "https://redfish.dmtf.org/profiles/swordfish",
    "MinVersion": "1.0.2",
    "ConditionalRequirements": [{
      "Purpose": "Either SwordfishFabricConnectivityRightsforEthernet or
      SwordfishFabricConnectivityRightsforRDMA must be supported.",
      "CompareProfile": "SwordfishFabricConnectivityRightsforEthernet",
      "CompareType": "Absent",
      "ReadRequirement": "Mandatory"
    }]
  },
  "SwordfishFabricAccessRightsforEthernet": {
    "Repository": "https://redfish.dmtf.org/profiles/swordfish",
    "MinVersion": "1.0.2",
    "ReadRequirement": "None",
    "ConditionalRequirements": [{
      "Purpose": "Either SwordfishFabricAccessRightsforEthernet or
      SwordfishFabricAccessRights must be supported.",
      "CompareProfile": "SwordfishFabricAccessRights",
      "CompareType": "Absent",
      "ReadRequirement": "Mandatory"
    }]
  },
  "SwordfishFabricAccessRights": {
    "Repository": "https://redfish.dmtf.org/profiles/swordfish",
    "MinVersion": "1.0.2",
    "ReadRequirement": "None",
    "ConditionalRequirements": [{
      "Purpose": "Either SwordfishFabricAccessRightsforEthernet or
      SwordfishFabricAccessRights must be supported.",
```

```
        "CompareProfile": "SwordfishFabricAccessRightsforEthernet",
        "CompareType": "Absent",
        "ReadRequirement": "Mandatory"
      }]
    }
  },
```

3 Feature Overview

The features summarized in this document are taken from vesion 1.6.0 of the Swordfish Features Registry and are summarized in Table 5 .

Feature	Version	Profile	Summary
SNIA.Swordfish.Block.CapacityManagement	1.2.0	SwordfishBlockCapacityManagement.v1_2_0.json	Supports the Swordfish Block Capacity Management Feature.
SNIA.Swordfish.Block.CoSLocalReplication	1.3.0	SwordfishBlockCoSLocalReplication.v1_3_0.json	Supports the local replication for block storage Feature when using the Swordfish Class Of Service Feature.
SNIA.Swordfish.Block.CoSRemoteReplication	1.3.0	SwordfishBlockRemoteReplication.v1_3_0.json	Supports the remote replication for block storage Feature when using the Swordfish Class Of Service Feature.
SNIA.Swordfish.Block.LocalReplication	1.3.0	SwordfishBlockLocalReplication.v1_3_0.json	Supports the local replication for block storage Feature.
SNIA.Swordfish.Block.MappingMasking	1.3.0	SwordfishBlockMappingMasking.v1_3_0.json	Supports the Swordfish Block Mapping and Masking Feature.
SNIA.Swordfish.Block.Provisioning	1.3.0	SwordfishBlockProvisioning.v1_3_0.json	Supports the Block Provisioning Feature.
SNIA.Swordfish.Block.RemoteReplication	1.2.1	SwordfishBlockRemoteReplication.v1_2_1.json	Supports the remote replication for block storage Feature.
SNIA.Swordfish.CoSDiscovery	1.2.0	SwordfishCoSDiscovery.v1_2_0.json	Supports Discovery requirements when using the Swordfish Class Of Service Feature.
SNIA.Swordfish.Discovery	1.1.4	SwordfishDiscov	Supports discovery of resources

Feature	Version	Profile	Summary
		ery.v1_1_4.json	in a Swordfish system.
SNIA.Swordfish.EnergyStar	1.1.4	SwordfishEnergyStar.v1_1_4.json	Supports the Swordfish EnergyStar for Storage Feature.
SNIA.Swordfish.EventNotification	1.2.5	SwordfishEventNotification.v1_2_5.json	Supports the Swordfish Event Notification Feature.
SNIA.Swordfish.FabricAccessRights	1.0.2	SwordfishFabricAccessRights.v1_0_2.json	Supports the Swordfish Fabric Access Rights Feature.
SNIA.Swordfish.FabricAccessRightsforEthernet	1.0.2	SwordfishFabricAccessRightsforEthernet.v1_0_2.json	Supports the Swordfish Fabric Access Rights for Ethernet Feature.
SNIA.Swordfish.FabricConnectivityRights	1.0.2	SwordfishFabricConnectivityRights.v1_0_2.json	Supports the Swordfish Fabric Connectivity Rights Feature.
SNIA.Swordfish.FabricConnectivityRightsforEthernet	1.0.2	SwordfishFabricConnectivityRightsforEthernet.v1_0_2.json	Supports the Swordfish Fabric Connectivity Rights for Ethernet Feature.
SNIA.Swordfish.FabricConnectivityRightsforRDMA	1.0.2	SwordfishFabricConnectivityRightsforRDMA.v1_0_2.json	Supports the Swordfish Fabric Connectivity Rights for RDMA Feature.
SNIA.Swordfish.File.CapacityManagement	1.1.4	SwordfishFileCapacityManagement.v1_1_4.json	Supports the Swordfish File Capacity Management Feature.
SNIA.Swordfish.File.Provisioning	1.1.4	SwordfishFileProvisioning.v1_1_4.json	Supports the File Provisioning Feature.
SNIA.Swordfish.IOPerformance	1.2.0	SwordfishIOPerformance.v1_2_0.json	Supports the Swordfish IOPerformance Feature.
SNIA.Swordfish.ManagementController	1.0.2	SwordfishManagementController.v1_0_2.json	Supports the Swordfish Management Controller (e.g., embedded management controller) Feature.
SNIA.Swordfish.NVMeDrive	1.3.0	SwordfishNVMeDrive.v1_3_0.jso	Supports the Swordfish NVMe Drive Feature.

Feature	Version	Profile	Summary
		n	
SNIA.Swordfish.NVMeDriveAdvancedFeatures	1.2.0	SwordfishNVMeDriveAdvancedFeatures.v1_2_0.json	Supports Advanced Features functionality for NVMe Drives.
SNIA.Swordfish.NVMeDriveEthernetAttach	1.2.1	SwordfishNVMeEthernetAttach.v1_2_1.json	Supports Ethernet Attach functionality for NVMe Drives.
SNIA.Swordfish.NVMeEBOF	1.2.1	SwordfishNVMeEBOF.v1_2_1.json	Supports the NVMe EBOF Feature set.
SNIA.Swordfish.NVMeFrontEnd	1.3.0	SwordfishNVMeFrontEnd.v1_3_0.json	Supports requirements for Swordfish implementations with NVMe front-end interfaces.
SNIA.Swordfish.NVMeoF	1.2.0	SwordfishNVMeoF.v1_2_0.json	Supports the NVMe over Fabrics set.
SNIA.Swordfish.PCIeJBOF	1.2.1	SwordfishPCIeJBOF.v1_2_1.json	Supports the PCIe JBOF Feature set.

Table 5: Feature overview

4 Profiles

4.1 Overview

All profile entries, at the profile, resource, or property level, are “additive”. That is, each requirement can only apply more rigid requirements that override less rigid requirements.

Profiles define a minimum set of required functionality. Swordfish profiles can also specify functionality that should not be implemented.

However, profile requirements do not allow for exclusions of data. Implementations are able to provide more data in their resources than required by a profile, as an implementation likely addresses multiple use cases or profiles. This includes both standard properties and OEM extensions.

Swordfish profiles are generally defined using a building block approach.

Each profile review contains:

- Summary information, including version, purpose, owner, and supported features
- Protocol and registry information
- Additional profiles requirements
- A list of related resources, including:
    - operations permitted against them (e.g., read / write, CRUD). Reach and Write operations are labeled as mandatory (M), recommended (R), if-implemented (IfImp), or do-not-implement (DNI), and may include a conditional reqiurement (/C). CRUD requirements are labeled either true (T) or false (F).
    - any conditional requirements that constrain their implementation

4.2 SwordfishBlockCapacityManagement profile

4.2.1 Summary Information

- Version: 1.2.0
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish Block Capacity Management implementation requirements.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0

- Supported Features
    - SNIA.Swordfish.Block.CapacityManagement

4.2.2 Protocol Information

No protocol information defined in this profile.

4.2.3 Other Registry Information

4.2.4 Required profiles

The SwordfishBlockCapacityManagement profile depends upon the profiles listed in Table 6 .

Profile	Version	Conditions
SwordfishBlockProvisioning	1.3.0	

Table 6: Required profiles for SwordfishBlockCapacityManagement

4.2.5 Related resources

The SwordfishBlockCapacityManagement profile relies on the resources listed in Table 7

Resource	Purpose
VolumeCollection	
Volume	One of the following mechanisms is required to support reporting of volume capacity. In order to claim support of the Capacity Management Feature, correspondingly 'write' (expand) on that property must be supported.
CapacityBytes	Either CapacityBytes or Capacity/#Data/#AllocatedBytes must be supported.
Capacity	
Data	
AllocatedBytes	Either CapacityBytes or Capacity/#Data/#AllocatedBytes must be supported.

Table 7: Related resources for SwordfishBlockCapacityManagement

4.2.6 Resource Manipulation

The SwordfishBlockCapacityManagement profile requires the resource manipulations listed in Table 8

Resource	Read	Write	Create	Delete	Update
VolumeCollection	IfImp				T
Volume	M				
CapacityBytes	R	M/C			
Capacity	M				
Data	M				
AllocatedBytes	M	M/C			

Table 8: Resource manipulations for SwordfishBlockCapacityManagement

4.2.7 Conditional Requirements

4.2.7.1 CapacityBytes

Purpose

Either CapacityBytes or Capacity/#Data/#AllocatedBytes must be supported.

Test Condition / Requirement

if /Capacity/Data/AllocatedBytes is Absent, WriteRequirement is Mandatory

based on:

- CompareProperty: /Capacity/Data/AllocatedBytes
- CompareType: Absent
- WriteRequirement: Mandatory

4.2.7.2 AllocatedBytes

Purpose

Either CapacityBytes or Capacity/#Data/#AllocatedBytes must be supported.

Test Condition / Requirement

if CapacityBytes is Absent, WriteRequirement is Mandatory

based on:

- CompareProperty: CapacityBytes
- CompareType: Absent
- WriteRequirement: Mandatory

4.3 SwordfishBlockCoSLocalReplication profile

4.3.1 Summary Information

- Version: 1.3.0
- Contact Information: SNIA.org
- Purpose: Define local replication block requirements for service-based Swordfish conformant implementations.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.Block.CoSLocalReplication

4.3.2 Protocol Information

No protocol information defined in this profile.

4.3.3 Other Registry Information

4.3.4 Required profiles

The SwordfishBlockCoSLocalReplication profile depends upon the profiles listed in Table 9 .

Profile	Version	Conditions
SwordfishBlockProvisioning	1.3.0	
SwordfishCoSDiscovery	1.2.0	

Table 9: Required profiles for SwordfishBlockCoSLocalReplication

4.3.5 Related resources

The SwordfishBlockCoSLocalReplication profile relies on the resources listed in Table 10

Resource	Purpose
Volume	
ReplicaTargets	At least one volume must exist defined as a replica target.

Resource	Purpose
ReplicaInfo	At least one volume must exist defined as a replica.
ReplicaType	At least one volume must exist defined as a replica.
ReplicaFaultDomain	The ReplicaFaultDomain type must be set to Local. Must be set to [‘Local’]
StorageService	
Links	
DataProtectionLoSCapabilities	The Data Protection Line of Service must be supported. DataProtectionLoSCapabilities are a pre-requisite to the Line of Service.
DataProtectionLineOfServiceCollection	The Data Protection Line of Service must be supported.
Members	MinCount = 0
DataProtectionLineOfService	
ACTIONS	
CreateReplicas()	Ability to create an on-demand (ad-hoc) replica.

Table 10: Related resources for SwordfishBlockCoSLocalReplication

4.3.6 Resource Manipulation

The SwordfishBlockCoSLocalReplication profile requires the resource manipulations listed in Table 11

Resource	Read	Write	Create	Delete	Update
Volume	M				
ReplicaTargets	M/C				
ReplicaInfo	M/C				
ReplicaType	M/C				
ReplicaFaultDomain	M/C				
StorageService	M				
Links	M				
DataProtectionLoSCapabilities	IfImp				

Resource	Read	Write	Create	Delete	Update
DataProtectionLineOfServiceCollection	IfImp		T	T	
Members	M				
DataProtectionLineOfService	M				
ACTIONS					
CreateReplicas	M				
ACTIONS					
CreateReplicas	M				

Table 11: Resource manipulations for SwordfishBlockCoSLocalReplication

4.3.7 Conditional Requirements

4.3.7.1 ReplicaTargets

Purpose

At least one volume must exist defined as a replica target.

Test Condition / Requirement

When subordinate to [‘StorageService’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘StorageService’]
- ReadRequirement: Mandatory

4.3.7.2 ReplicaInfo

Purpose

At least one volume must exist defined as a replica.

Test Condition / Requirement

When subordinate to [‘StorageService’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘StorageService’]

- ReadRequirement: Mandatory

4.3.7.3 ReplicaType

Purpose

At least one volume must exist defined as a replica.

Test Condition / Requirement

When subordinate to [‘StorageService’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘StorageService’]
- ReadRequirement: Mandatory

4.3.7.4 ReplicaFaultDomain

Test Condition / Requirement

When subordinate to [‘StorageService’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘StorageService’]
- ReadRequirement: Mandatory

4.3.7.5 DataProtectionLoSCapabilities

Purpose

The Data Protection Line of Service must be supported. DataProtectionLoSCapabilities are a pre-requisite to the Line of Service.

Test Condition / Requirement

When subordinate to [‘StorageService’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘StorageService’]

4.4 SwordfishBlockLocalReplication profile

4.4.1 Summary Information

- Version: 1.3.0
- Contact Information: SNIA.org
- Purpose: Define local replication block requirements for Swordfish conformant implementations.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.Block.LocalReplication

4.4.2 Protocol Information

No protocol information defined in this profile.

4.4.3 Other Registry Information

4.4.4 Required profiles

The SwordfishBlockLocalReplication profile depends upon the profiles listed in Table 12 .

Profile	Version	Conditions
SwordfishBlockProvisioning	1.3.0	

Table 12: Required profiles for SwordfishBlockLocalReplication

4.4.5 Related resources

The SwordfishBlockLocalReplication profile relies on the resources listed in Table 13

Resource	Purpose
Volume	
ACTIONS	
AssignReplicaTarget()	Ability to reset the system is a core requirement of most users.

Resource	Purpose
	Either CreateReplicaTarget or AssignReplicaTarget must be supported in order to advertise local replication support, when ReplicationEnabled is defined on the volume. Either CreateReplicaTarget or AssignReplicaTarget must be supported in order to advertise local replication support.
ReplicationEnabled	The ReplicationEnabled property shall be set on the volume to indicate that this volume is capable of supporting replication. When implemented and false, replication is supported and disabled. When implemented and true, replication is enabled. When replication is enabled, one of the two Actions, CreateReplicaTarget and AssignReplicaTarget, shall be implemented, regardless of whether the volume is currently in a replica relationship. At least one volume must have ReplicationEnabled to support Swordfish block local replication; ReplicaFaultDomain type must be set to local.
ReplicaTargets	
ReplicaInfo	
ReplicaType	
ReplicaFaultDomain	The ReplicaFaultDomain type must be set to Local.

Table 13: Related resources for SwordfishBlockLocalReplication

4.4.6 Resource Manipulation

The SwordfishBlockLocalReplication profile requires the resource manipulations listed in Table 14

Resource	Read	Write	Create	Delete	Update
Volume	M				
ACTIONS					
CreateReplicaTarget	M/C				
AssignReplicaTarget	M/C				
ReplicationEnabled	M/C				
ReplicaTargets	IfImp				
ReplicaInfo	IfImp				

Resource	Read	Write	Create	Delete	Update
ReplicaType	M				
ReplicaFaultDomain	M				

Table 14: Resource manipulations for SwordfishBlockLocalReplication

4.4.7 Conditional Requirements

4.4.7.1 ReplicationEnabled

Purpose

At least one volume must have ReplicationEnabled to support Swordfish block local replication; ReplicaFaultDomain type must be set to local.

Test Condition / Requirement

When subordinate to [‘Volume’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘Volume’]
- ReadRequirement: Mandatory

4.5 SwordfishBlockMappingMasking profile

4.5.1 Summary Information

- Version: 1.3.0
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish Block Mapping and Masking Feature details.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.Block.MappingMasking

4.5.2 Protocol Information

No protocol information defined in this profile.

4.5.3 Other Registry Information

4.5.4 Required profiles

The SwordfishBlockMappingMasking profile depends upon the profiles listed in Table 15 .

Profile	Version	Conditions
CoSDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.
SwordfishDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.

Table 15: Required profiles for SwordfishBlockMappingMasking

4.5.5 Related resources

The SwordfishBlockMappingMasking profile relies on the resources listed in Table 16

Resource	Purpose
ConnectionCollection	

Resource	Purpose
Members	MinCount = 1
EndpointCollection	
Members	MinCount = 1
EndpointGroupCollection	
Members	MinCount = 1
Connection	
ConnectionType	
VolumeInfo	
AccessCapabilities	
LUN	
Volume	
Links	
InitiatorEndpointGroups	
TargetEndpointGroups	

Table 16: Related resources for SwordfishBlockMappingMasking

4.5.6 Resource Manipulation

The SwordfishBlockMappingMasking profile requires the resource manipulations listed in Table 17

Resource	Read	Write	Create	Delete	Update
ConnectionCollection	S		T	T	T
Members	M				
EndpointCollection	IfImp		T	T	T
Members	M				
EndpointGroupCollection	IfImp		T	T	T
Members	M				
Connection	M				
ConnectionType	M				
VolumeInfo	M				
AccessCapabilities	M				
LUN	M				

Resource	Read	Write	Create	Delete	Update
Volume	M				
Links	M				
InitiatorEndpointGroups	M				
TargetEndpointGroups	M				

Table 17: Resource manipulations for SwordfishBlockMappingMasking

4.5.7 Conditional Requirements

4.5.7.1 ConnectionCollection

Test Condition / Requirement

When subordinate to [‘Fabric’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Fabric’]

4.6 SwordfishBlockProvisioning profile

4.6.1 Summary Information

- Version: 1.3.0
- Contact Information: SNIA.org
- Purpose: Defines Block Provisioning requirements for Swordfish implementations.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.Block.Provisioning

4.6.2 Protocol Information

No protocol information defined in this profile.

4.6.3 Other Registry Information

4.6.4 Required profiles

The SwordfishBlockProvisioning profile depends upon the profiles listed in Table 18 .

Profile	Version	Conditions
SwordfishCoSDiscovery	1.2.0	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.
SwordfishDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.

Table 18: Required profiles for SwordfishBlockProvisioning

4.6.5 Related resources

The SwordfishBlockProvisioning profile relies on the resources listed in Table 19

Resource	Purpose
VolumeCollection	

Resource	Purpose

Table 19: Related resources for SwordfishBlockProvisioning

4.6.6 Resource Manipulation

The SwordfishBlockProvisioning profile requires the resource manipulations listed in Table 20

Resource	Read	Write	Create	Delete	Update
VolumeCollection	IfImp		T	T	

Table 20: Resource manipulations for SwordfishBlockProvisioning

4.6.7 Conditional Requirements

4.6.7.1 VolumeCollection

Test Condition / Requirement

When subordinate to [‘Storage’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Storage’]

Test Condition / Requirement

When subordinate to [‘StoragePool’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘StoragePool’]

Test Condition / Requirement

When subordinate to [‘CapacitySource’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘CapacitySource’]

Test Condition / Requirement

When subordinate to [‘StorageService’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘StorageService’]

4.7 SwordfishBlockRemoteReplication profile

4.7.1 Summary Information

- Version: 1.2.1
- Contact Information: SNIA.org
- Purpose: Define remote replication block requirements for Swordfish conformant implementations.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.Block.RemoteReplication

4.7.2 Protocol Information

No protocol information defined in this profile.

4.7.3 Other Registry Information

4.7.4 Required profiles

The SwordfishBlockRemoteReplication profile depends upon the profiles listed in Table 21 .

Profile	Version	Conditions
SwordfishBlockProvisioning	1.2.2	

Table 21: Required profiles for SwordfishBlockRemoteReplication

4.7.5 Related resources

The SwordfishBlockRemoteReplication profile relies on the resources listed in Table 22

Resource	Purpose
Volume	
ACTIONS	
AssignReplicaTarget()	Ability to reset the system is a core requirement of most users.

Resource	Purpose
	Either CreateReplicaTarget or AssignReplicaTarget must be supported in order to advertise remote replication support, when ReplicationEnabled is defined on the volume. Either CreateReplicaTarget or AssignReplicaTarget must be supported in order to advertise remote replication support.
ReplicationEnabled	The ReplicationEnabled property shall be set on the volume to indicate that this volume is capable of supporting replication. When implemented and false, replication is supported and disabled. When implemented and true, replication is enabled. When replication is enabled, one of the two Actions, CreateReplicaTarget and AssignReplicaTarget, shall be implemented, regardless of whether the volume is currently in a replica relationship. At least one volume must have ReplicationEnabled to support Swordfish block remote replication; ReplicaFaultDomain type must be set to Remote.
ReplicaTargets	
ReplicaInfo	
ReplicaType	
ReplicaFaultDomain	The ReplicaFaultDomain type must be set to Remote.

Table 22: Related resources for SwordfishBlockRemoteReplication

4.7.6 Resource Manipulation

The SwordfishBlockRemoteReplication profile requires the resource manipulations listed in Table 23

Resource	Read	Write	Create	Delete	Update
Volume	M				
ACTIONS					
CreateReplicaTarget	M/C				
AssignReplicaTarget	M/C				
ReplicationEnabled	M/C				
ReplicaTargets	IfImp				
ReplicaInfo	IfImp				

Resource	Read	Write	Create	Delete	Update
ReplicaType	M				
ReplicaFaultDomain	M				

Table 23: Resource manipulations for SwordfishBlockRemoteReplication

4.7.7 Conditional Requirements

4.7.7.1 ReplicationEnabled

Purpose

At least one volume must have ReplicationEnabled to support Swordfish block remote replication; ReplicaFaultDomain type must be set to Remote.

Test Condition / Requirement

When subordinate to ['Volume'], ReadRequirement is Mandatory

based on:

- SubordinatetoResource ['Volume']
- ReadRequirement: Mandatory

4.8 SwordfishCoSDiscovery profile

4.8.1 Summary Information

- Version: 1.2.0
- Contact Information: SNIA.org
- Purpose: Defines Class Of Service Discovery requirements for service-based Swordfish implementations.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.CoSDiscovery

4.8.2 Protocol Information

- Protocol version: 1.18.0

4.8.3 Other Registry Information

4.8.4 Required profiles

4.8.5 Related resources

The SwordfishCoSDiscovery profile relies on the resources listed in Table 24

Resource	Purpose
StorageServiceCollection	
Members	MinCount = 1
ServiceRoot	
StorageServices	
StorageService	
Identifier	
Status	
State	
Health	

Resource	Purpose
ClassesOfService	
Volumes	Volumes must exist unless FileSystems exists, in which case it is only Recommended.
StoragePools	
FileSystems	FileSystems must exist unless Volumes exists, in which case it is only Recommended.
Links	
DefaultClassOfService	When there is at least one class of service instantiated, the DefaultClassOfService property should be set.
DataProtectionLoSCapabilities	One or more Line of Service Capabilities must be defined.
DataSecurityLoSCapabilities	One or more Line of Service Capabilities must be defined.
DataStorageLoSCapabilities	One or more Line of Service Capabilities must be defined.
IOConnectivityLoSCapabilities	One or more Line of Service Capabilities must be defined.
IOPerformanceLoSCapabilities	One or more Line of Service Capabilities must be defined.

Table 24: Related resources for SwordfishCoSDiscovery

4.8.6 Resource Manipulation

The SwordfishCoSDiscovery profile requires the resource manipulations listed in Table 25

Resource	Read	Write	Create	Delete	Update
StorageServiceCollection	M				
Members	M				
ServiceRoot	M				
StorageServices	M				
StorageService	M				
Identifier	M				
Status	M				
State	M				
Health	M				
ClassesOfService	M				

Resource	Read	Write	Create	Delete	Update
Volumes	M/C				
StoragePools	M				
FileSystems	M/C				
Links	M				
DefaultClassOfService	R				
DataProtectionLoSCapabilities	M/C				
DataSecurityLoSCapabilities	M/C				
DataStorageLoSCapabilities	M/C				
IOConnectivityLoSCapabilities	M/C				
IOPerformanceLoSCapabilities	M/C				

Table 25: Resource manipulations for SwordfishCoSDiscovery

4.8.7 Conditional Requirements

4.8.7.1 Volumes

Purpose

Volumes must exist unless FileSystems exists, in which case it is only Recommended.

Test Condition / Requirement

if FileSystems is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: FileSystems
- CompareType: Absent
- ReadRequirement: Mandatory

4.8.7.2 FileSystems

Purpose

FileSystems must exist unless Volumes exists, in which case it is only Recommended.

Test Condition / Requirement

if Volumes is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: Volumes
- CompareType: Absent
- ReadRequirement: Mandatory

4.8.7.3 DataProtectionLoSCapabilities

Purpose

One or more Line of Service Capabilities must be defined.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.8.7.4 DataSecurityLoSCapabilities

Purpose

One or more Line of Service Capabilities must be defined.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.8.7.5 DataStorageLoSCapabilities

Purpose

One or more Line of Service Capabilities must be defined.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.8.7.6 IOConnectivityLoSCapabilities

Purpose

One or more Line of Service Capabilities must be defined.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.8.7.7 IOPerformanceLoSCapabilities

Purpose

One or more Line of Service Capabilities must be defined.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.9 SwordfishDiscovery profile

4.9.1 Summary Information

- Version: 1.1.4
- Contact Information: SNIA.org
- Purpose: Define requirements for discovery of resources and relationships that are common to all Swordfish conformant implementations.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.Discovery

summary supplement

4.9.2 Protocol Information

- Protocol version: 1.18.0

4.9.3 Other Registry Information

4.9.4 Required profiles

required profiles supplement

4.9.5 Related resources

related resources supplement

The SwordfishDiscovery profile relies on the resources listed in Table 26

Resource	Purpose
ServiceRoot	
Chassis	If Drives are used as capacity sources, these must be in Chassis/{ChassisId}/Drives.
Storage	
Systems	If Storage is attached to a ComputerSystem rather than

Resource	Purpose
	instantiated as a Standalone instance.
StorageSystems	If Storage is attached to a ComputerSystem rather than instantiated as a Standalone instance.
StorageCollection	
Members	MinCount = 1
VolumeCollection	
Chassis	
Drives	If Drives provide the capacity source for the related Storage object, Drives are required to be in at least one Chassis representation. As physical configurations can be represented by multiple Chassis instances, not all Chassis instances must include a Drives collection.
Storage	
Volumes	
StoragePools	StoragePools must be implemented in primary Storage object instances. For Storage objects modeling embedded or back-end devices (e.g., NVMe drives), StoragePools are not required to be modeled.
Drives	There are two requirements for Drives. The first is encapsulated in a custom test, as it cannot be well specified in the profile language. 1: The implementation must choose at least one selected media container (appropriate for the system's selected media type) and instantiate a collection for it. This may be instantiated as a CapacitySources structure to feed a StoragePool, FileSystem, or Volume. When drives are used, the Drives Collection in either Storage or, when implementing the ClassOfService Feature, StorageServices shall be implemented. 2: When drives are used, the Drives Collection in Chassis shall be implemented.
ComputerSystem	
HostingRoles	Swordfish implementations must specify a hosting role type of StorageServer. Must be set to ['StorageServer']

Table 26: Related resources for SwordfishDiscovery

4.9.6 Resource Manipulation

resource manipulation supplement

The SwordfishDiscovery profile requires the resource manipulations listed in Table 27

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Chassis	IfImp				
Storage	M				
Systems	IfImp				
StorageSystems	IfImp				
StorageCollection	M				
Members	M				
VolumeCollection	M				
Chassis	M				
Drives	IfImp/C				
Storage	M				
Volumes	M				
StoragePools	IfImp				
Drives	M/C				
ComputerSystem	IfImp				
HostingRoles	M				

Table 27: Resource manipulations for SwordfishDiscovery

4.9.7 Conditional Requirements

conditional requirements supplement

4.9.7.1 Drives

Purpose

If Drives provide the capacity source for the related Storage object, Drives are required to be in at least one Chassis representation. As physical configurations can be represented by multiple Chassis instances, not all Chassis instances must include a Drives collection.

Test Condition / Requirement

When subordinate to [‘Chassis’], ReadRequirement is IfImplemented

based on:

- SubordinatetoResource [‘Chassis’]
- ReadRequirement: IfImplemented

4.9.7.2 Drives

Purpose

There are two requirements for Drives. The first is encapsulated in a custom test, as it cannot be well specified in the profile language. 1: The implementation must choose at least one selected media container (appropriate for the system’s selected media type) and instantiate a collection for it. This may be instantiated as a CapacitySources structure to feed a StoragePool, FileSystem, or Volume. When drives are used, the Drives Collection in either Storage or, when implementing the ClassOfService Feature, StorageServices shall be implemented. 2: When drives are used, the Drives Collection in Chassis shall be implemented.

Test Condition / Requirement

if /redfish/v1/Chassis/{ChassisId}/Drives is Present, ReadRequirement is Mandatory

based on:

- CompareProperty: /redfish/v1/Chassis/{ChassisId}/Drives
- CompareType: Present
- ReadRequirement: Mandatory

4.10 SwordfishEnergyStar profile

4.10.1 Summary Information

- Version: 1.1.4
- Contact Information: SNIA.org
- Purpose: Defines Requirements for Swordfish to meet EnergyStar reporting for storage.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.EnergyStar

4.10.2 Protocol Information

No protocol information defined in this profile.

4.10.3 Other Registry Information

4.10.4 Required profiles

The SwordfishEnergyStar profile depends upon the profiles listed in Table 28 .

Profile	Version	Conditions
CoSDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.
SwordfishDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.

Table 28: Required profiles for SwordfishEnergyStar

4.10.5 Related resources

The SwordfishEnergyStar profile relies on the resources listed in Table 29

Resource	Purpose
Chassis	To meet EnergyStar compliance, if the data is not reported or

Resource	Purpose
	aggregated at the system level, it shall be reported for each shelf or enclosure. (Note that data can be aggregated external to the system by the client when reported by subcomponents.) These properties should be collected in the same request. The system shall be able to support client queries of these properties at an interval of at least every 10 seconds (per ENERGY STAR Data Center Storage Version 1.1 Updated Program Requirements – April 1, 2019).
Power	
PowerSupply	
PowerInputWatts	
Thermal	Collect the inlet temperature for the power supply.
ReadingCelsius	
PhysicalContext	Shall be equal to ['Intake']

Table 29: Related resources for SwordfishEnergyStar

4.10.6 Resource Manipulation

The SwordfishEnergyStar profile requires the resource manipulations listed in Table 30

Resource	Read	Write	Create	Delete	Update
Chassis	IfImp				
Power	M				
PowerSupply	M				
PowerInputWatts	M				
Thermal	M				
ReadingCelsius	M				
PhysicalContext	M				

Table 30: Resource manipulations for SwordfishEnergyStar

4.10.7 Conditional Requirements

None defined.

4.11 SwordfishEventNotification profile

4.11.1 Summary Information

- Version: 1.3.0
- Contact Information: SNIA.org
- Purpose: Defines Swordfish Event Notification requirements.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.7.0
- Supported Features
    - SNIA.Swordfish.EventNotification

4.11.2 Protocol Information

- Protocol version: 1.18.0

4.11.3 Other Registry Information

4.11.3.1 Registry Name:Base

Registry details:

- Minimum version: 1.4.0
- Messages
    - **ResourceTypeIncompatible:** The ResourceTypeIncompatible event shall be sent when the requested resource type of the operation does not match that for the operation. This shall apply to all instantiated resources.

4.11.3.2 Registry Name:ResourceEvent

Registry details:

- Minimum version: 1.2.0
- Messages
    - **ResourceCreated:** The ResourceCreated event shall be sent on creation of a resource to indicate that the resource has been successfully created. This shall apply to all instantiated resources of type 'Collection'. The event shall be sent on any expansion of the collection, whether done by the user or by the system.
    - **ResourceRemoved:** The ResourceRemoved event shall be sent on removal (deletion) of a resource to indicate that the resource has been successfully removed. This applies to

all resources of type 'Collection'. The event shall be sent on any removal of resources from the collection, whether done by the user or by the system.

- **ResourceErrorsDetected:** The ResourceRemoved event shall be sent on detection of errors on a resource. As there are no standard schema properties that reflect error / threshold information in this release, these may be mapped to OEM properties or device-specific events, until standard properties are defined.
- **ResourceErrorsCorrected:** The ResourceRemoved event shall be sent on detection of corrected errors on a resource. As there are no standard schema properties that reflect error / threshold information in this release, these may be mapped to OEM properties or device-specific events, until standard properties are defined.
- **ResourceErrorThresholdExceeded:** The ResourceRemoved event shall be sent when the threshold for detected errors has been exceeded. As there are no standard schema properties that reflect error / threshold information in this release, these may be mapped to OEM properties or device-specific events, until standard properties are defined.
- **ResourceErrorThresholdCleared:** The ResourceRemoved event shall be sent when the threshold for detected errors has been cleared. As there are no standard schema properties that reflect error / threshold information in this release, these may be mapped to OEM properties or device-specific events, until standard properties are defined.
- **ResourceWarningThresholdExceeded:** The ResourceRemoved event shall be sent when the warning threshold for a resource has been exceeded. As there are no standard schema properties that reflect error / threshold information in this release, these may be mapped to OEM properties or device-specific events, until standard properties are defined.
- **ResourceWarningThresholdCleared:** The ResourceRemoved event shall be sent on when the resources has cleared a warning threshold. As there are no standard schema properties that reflect error / threshold information in this release, these may be mapped to OEM properties or device-specific events, until standard properties are defined.
- **ResourceStatusChangedOK:** The ResourceStatusChangedOK event shall be sent when the Status.State has changed and the Status.Health of the resource is OK. The new and old values of the State are reported in the Event message. This shall apply to all instantiated resources that implements Status.State.
- **ResourceStatusChangedWarning:** The ResourceStatusChangedWarning event shall be sent when the Status.State has changed and the Status.Health of the resource is Warning. The value of the State is reported in the Event message. This shall apply to all instantiated resources that implements Status.State.
- **ResourceStatusChangedCritical:** The ResourceStatusChangedCritical event shall be sent when the Status.State has changed and the Status.Health of the resource is Critical. The value of the State is reported in the Event message. This shall apply to all instantiated resources that implements Status.State.
- **URIForResourceChanged:** The URIForResourceChanged event shall be sent when the URI for the model resource has changed. Examples for this would be physical component replacement or redistribution.
- **ResourceChanged:** The ResourceChanged event shall be sent when one or more

properties of a resource have changed. This shall not be used whenever there is another event message for that specific change, such as only the state has changed. If multiple resource properties have changed, the event notification should be grouped together whenever possible. This should not be sent for changes related to fine-grained volatile values (eg., capacity, power, temperature). In the case of threshold crossings, other mechanisms should be employed, such as logging or threshold triggers.

- **ResourceVersionIncompatible:** The ResourceVersionIncompatible event shall be sent when an incompatible version of software has been detected. This corresponds to schema that have a 'Version' or 'Revision' property, generally presenting either firmware or software version information. As there are no schema properties that specifically reflect compatibility, this may be mapped to OEM properties or device-specific events, until standard properties are defined.
- **ResourceSelfTestFailed:** The ResourceSelfTestFailed event shall be sent when a self-test has failed. As there are no schema properties that reflect self-test controls, these may be mapped to OEM properties or device-specific events, until standard properties are defined.
- **ResourceSelfTestCompleted:** The ResourceSelfTestCompleted event shall be sent when a self-test has completed. As there are no schema properties that reflect self-test controls, these may be mapped to OEM properties or device-specific events, until standard properties are defined.

4.11.4 Required profiles

The SwordfishEventNotification profile depends upon the profiles listed in Table 31 .

Profile	Version	Conditions
CoSDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.
SwordfishDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.

Table 31: Required profiles for SwordfishEventNotification

4.11.5 Related resources

The SwordfishEventNotification profile relies on the resources listed in Table 32

Resource	Purpose
EventService	

Table 32: Related resources for SwordfishEventNotification

4.11.6 Resource Manipulation

The SwordfishEventNotification profile requires the resource manipulations listed in Table 33

Resource	Read	Write	Create	Delete	Update
EventService	M				

Table 33: Resource manipulations for SwordfishEventNotification

4.11.7 Conditional Requirements

None defined.

4.12 SwordfishFabricAccessRights profile

4.12.1 Summary Information

- Version: 1.0.2
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish Access Rights implementation requirements using Fabrics.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.FabricAccessRights

4.12.2 Protocol Information

- Protocol version: 1.18.0

4.12.3 Other Registry Information

4.12.4 Required profiles

The SwordfishFabricAccessRights profile depends upon the profiles listed in Table 34 .

Profile	Version	Conditions
SwordfishDiscovery	1.1.4	

Table 34: Required profiles for SwordfishFabricAccessRights

4.12.5 Related resources

The SwordfishFabricAccessRights profile relies on the resources listed in Table 35

Resource	Purpose
ServiceRoot	
Fabrics	
FabricCollection	
Members	MinCount = 1

Resource	Purpose
ConnectionCollection	
EndpointCollection	
Fabric	
Connections	Required when Switches not defined for the fabric (switches are required for connectivity rights management).
Description	
EndpointGroups	
Endpoints	
FabricType	
Id	
Name	
Status	
State	
Connection	
ConnectionType	
Description	
Id	
Links	
InitiatorEndpointGroups	Either InitiatorEndpointGroups or InitiatorEndpoints must be supported.
InitiatorEndpoints	Either InitiatorEndpoints or InitiatorEndpointGroups must be supported.
TargetEndpointGroups	Either TargetEndpointGroups or TargetEndpoints must be supported.
TargetEndpoints	Either TargetEndpoints or TargetEndpointGroups must be supported.
Name	
VolumeInfo	
AccessCapabilities	
Volume	
LUN	Dependent on protocol. Required for SCSI, FC connect.
Endpoint	
ConnectedEntities	

Resource	Purpose
EntityLink	
EntityRole	
EntityType	
EndpointProtocol	
Identifiers	
DurableName	
DurableNameFormat	
Links	
Connections	
EndpointGroup	
AccessState	This property is recommended as 'Do not implement'. This property has been deprecated.
Endpoints	This property is recommended as 'Do not implement'. This property has been deprecated.
GroupType	
Id	
Links	
Endpoints	
Connections	
Preferred	This property is recommended as 'Do not implement'. This property has been deprecated.

Table 35: Related resources for SwordfishFabricAccessRights

4.12.6 Resource Manipulation

The SwordfishFabricAccessRights profile requires the resource manipulations listed in Table 36

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Fabrics	M				
FabricCollection	M				
Members	M				
ConnectionCollection	M				

Resource	Read	Write	Create	Delete	Update
EndpointCollection	M				
Fabric	M				
Connections	M/C				
Description	M				
EndpointGroups	R				
Endpoints	M				
FabricType	M				
Id	M				
Name	M				
Status	M				
State	M				
Connection	M				
ConnectionType	M				
Description	M				
Id	M				
Links	M				
InitiatorEndpointGroups	M/C				
InitiatorEndpoints	M/C				
TargetEndpointGroups	M/C				
TargetEndpoints	M/C				
Name	M				
VolumeInfo	M				
AccessCapabilities	M				
Volume	M				
LUN	R				
Endpoint	M				
ConnectedEntities	M				
EntityLink	R				
EntityRole	M				
EntityType	M				
EndpointProtocol	M				
Identifiers	M				

Resource	Read	Write	Create	Delete	Update
DurableName	M				
DurableNameFormat	M				
Links	M				
Connections	R				
EndpointGroup	R				
AccessState	DNI				
Endpoints	DNI				
GroupType	M				
Id	M				
Links	M				
Endpoints	M				
Connections	M				
Preferred	DNI				

Table 36: Resource manipulations for SwordfishFabricAccessRights

4.12.7 Conditional Requirements

4.12.7.1 Connections

Purpose

Required when Switches not defined for the fabric (switches are required for connectivity rights management).

Test Condition / Requirement

if Switches is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: Switches
- CompareType: Absent
- ReadRequirement: Mandatory

4.12.7.2 InitiatorEndpointGroups

Purpose

Either InitiatorEndpointGroups or InitiatorEndpoints must be supported.

Test Condition / Requirement

if InitiatorEndpoints is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: InitiatorEndpoints
- CompareType: Absent
- ReadRequirement: Mandatory

4.12.7.3 InitiatorEndpoints

Purpose

Either InitiatorEndpoints or InitiatorEndpointGroups must be supported.

Test Condition / Requirement

if InitiatorEndpointGroups is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: InitiatorEndpointGroups
- CompareType: Absent
- ReadRequirement: Mandatory

4.12.7.4 TargetEndpointGroups

Purpose

Either TargetEndpointGroups or TargetEndpoints must be supported.

Test Condition / Requirement

if TargetEndpoints is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: TargetEndpoints
- CompareType: Absent
- ReadRequirement: Mandatory

4.12.7.5 TargetEndpoints

Purpose

Either TargetEndpoints or TargetEndpointGroups must be supported.

Test Condition / Requirement

if TargetEndpointGroups is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: TargetEndpointGroups
- CompareType: Absent
- ReadRequirement: Mandatory

4.13 SwordfishFabricAccessRightsforEthernet profile

4.13.1 Summary Information

- Version: 1.0.2
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish Access Rights requirements using Fabrics, for implementations using Ethernet.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.FabricAccessRightsforEthernet

4.13.2 Protocol Information

- Protocol version: 1.18.0

4.13.3 Other Registry Information

4.13.4 Required profiles

The SwordfishFabricAccessRightsforEthernet profile depends upon the profiles listed in Table 37 .

Profile	Version	Conditions
SwordfishFabricAccessRights	1.0.2	

Table 37: Required profiles for SwordfishFabricAccessRightsforEthernet

4.13.5 Related resources

The SwordfishFabricAccessRightsforEthernet profile relies on the resources listed in Table 38

Resource	Purpose
Endpoint	
IPTransportDetails	

Resource	Purpose
TransportProtocol	
IPv4Address	Either IPv4Address or IPv6Address must be supported.
IPv6Address	Either IPv4Address or IPv6Address must be supported.
Port	

Table 38: Related resources for SwordfishFabricAccessRightsforEthernet

4.13.6 Resource Manipulation

The SwordfishFabricAccessRightsforEthernet profile requires the resource manipulations listed in Table 39

Resource	Read	Write	Create	Delete	Update
Endpoint	M/C				
IPTransportDetails	M				
TransportProtocol	M				
IPv4Address	M/C				
IPv6Address	M/C				
Port	M				

Table 39: Resource manipulations for SwordfishFabricAccessRightsforEthernet

4.13.7 Conditional Requirements

4.13.7.1 Endpoint

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘EndpointCollection’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘Fabric’, ‘EndpointCollection’]
- ReadRequirement: Mandatory

4.13.7.2 IPv4Address

Purpose

Either IPv4Address or IPv6Address must be supported.

Test Condition / Requirement

if IPv6Address is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: IPv6Address
- CompareType: Absent
- ReadRequirement: Mandatory

4.13.7.3 IPv6Address

Purpose

Either IPv4Address or IPv6Address must be supported.

Test Condition / Requirement

if IPv4Address is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: IPv4Address
- CompareType: Absent
- ReadRequirement: Mandatory

4.14 SwordfishFabricConnectivityRights profile

4.14.1 Summary Information

- Version: 1.0.2
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish Connectivity Rights implementation requirements using Fabrics.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.FabricConnectivityRights

4.14.2 Protocol Information

- Protocol version: 1.18.0

4.14.3 Other Registry Information

4.14.4 Required profiles

The SwordfishFabricConnectivityRights profile depends upon the profiles listed in Table 40 .

Profile	Version	Conditions
SwordfishDiscovery	1.1.4	

Table 40: Required profiles for SwordfishFabricConnectivityRights

4.14.5 Related resources

The SwordfishFabricConnectivityRights profile relies on the resources listed in Table 41

Resource	Purpose
ServiceRoot	
Fabrics	
FabricCollection	
Members	MinCount = 1

Resource	Purpose
SwitchCollection	
ZoneCollection	
EndpointCollection	
Fabric	
Description	
EndpointGroups	
Endpoints	
FabricType	
Id	
Name	
MaxZones	
Status	
Conditions	Reports Conditions when HealthRollup is not 'OK'.
HealthRollup	
Health	
State	
Switches	Required when Connections not defined for the fabric (connections are required only for access rights management).
Zones	Required when Connections not defined for the fabric (connections are required only for access rights management).
Endpoint	
ConnectedEntities	
EntityLink	
EntityRole	
EntityType	
EndpointProtocol	
Identifiers	
DurableName	
DurableNameFormat	
Links	
Ports	
Zones	
EndpointGroup	

Resource	Purpose
GroupType	
Id	
Links	
Endpoints	
Connections	
Port	
CurrentSpeedGbps	
Id	
Links	
AssociatedEndpoints	
MaxSpeedGbps	
Name	
PortId	
PortProtocol	
Status	
State	
Switch	
Description	
FirmwareVersion	
Id	
IndicatorLED	Do not implement IndicatorLED
IsManaged	
Links	
Chassis	
Location	
Manufacturer	
Model	
Name	
PartNumber	
Ports	
SerialNumber	
SupportedProtocols	

Resource	Purpose
SwitchType	
Zone	
Description	
ExternalAccessibility	
Id	
Links	
Endpoints	
InvolvedSwitches	
Name	
Status	
State	
ZoneType	

Table 41: Related resources for SwordfishFabricConnectivityRights

4.14.6 Resource Manipulation

The SwordfishFabricConnectivityRights profile requires the resource manipulations listed in Table 42

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Fabrics	M				
FabricCollection	M				
Members	M				
SwitchCollection	M				
ZoneCollection	M				
EndpointCollection	M				
Fabric	M				
Description	M				
EndpointGroups	R				
Endpoints	M				
FabricType	M				
Id	M				

Resource	Read	Write	Create	Delete	Update
Name	M				
MaxZones	R				
Status	M				
Conditions	M/C				
HealthRollup	R				
Health	M				
State	M				
Switches	M/C				
Zones	M/C				
Endpoint	M				
ConnectedEntities	M				
EntityLink	R				
EntityRole	M				
EntityType	M				
EndpointProtocol	M				
Identifiers	M				
DurableName	M				
DurableNameFormat	M				
Links	M				
Ports	R				
Zones	R				
EndpointGroup	IfImp				
GroupType	M				
Id	M				
Links	M				
Endpoints	M				
Connections	M				
Port	M				
CurrentSpeedGbps	M				
Id	M				
Links	M				
AssociatedEndpoints	M				

Resource	Read	Write	Create	Delete	Update
MaxSpeedGbps	M				
Name	M				
PortId	M				
PortProtocol	M				
Status	M				
State	M				
Switch	M				
Description	M				
FirmwareVersion	M				
Id	M				
IndicatorLED	DNI				
IsManaged	M				
Links	M				
Chassis	M				
Location	R				
Manufacturer	M				
Model	M				
Name	M				
PartNumber	M				
Ports	M				
SerialNumber	M				
SupportedProtocols	M				
SwitchType	M				
Zone	M				
Description	M				
ExternalAccessibility	M				
Id	M				
Links	M				
Endpoints	M				
InvolvedSwitches	R				
Name	M				
Status	M				

Resource	Read	Write	Create	Delete	Update
State	M				
ZoneType	M				

Table 42: Resource manipulations for SwordfishFabricConnectivityRights

4.14.7 Conditional Requirements

4.14.7.1 Conditions

Purpose

Reports Conditions when HealthRollup is not 'OK'.

Test Condition / Requirement

if HealthRollup is AnyOf to ['Warning', 'Critical'], ReadRequirement is Mandatory

based on:

- CompareProperty: HealthRollup
- CompareType: AnyOf
- CompareValues: ['Warning', 'Critical']
- ReadRequirement: Mandatory

4.14.7.2 Switches

Purpose

Required when Connections not defined for the fabric (connections are required only for access rights management).

Test Condition / Requirement

if Connections is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: Connections
- CompareType: Absent
- ReadRequirement: Mandatory

4.14.7.3 Zones

Purpose

Required when Connections not defined for the fabric (connections are required only for access rights management).

Test Condition / Requirement

if Connections is Absent, ReadRequirement is Mandatory

based on:

- CompareProperty: Connections
- CompareType: Absent
- ReadRequirement: Mandatory

4.14.7.4 Endpoint

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘EndpointCollection’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Fabric’, ‘EndpointCollection’]

4.14.7.5 EndpointGroup

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘EndpointGroupCollection’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Fabric’, ‘EndpointGroupCollection’]

4.14.7.6 Port

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘SwitchCollection’, ‘Switch’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Fabric’, ‘SwitchCollection’, ‘Switch’]

4.14.7.7 Switch

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘SwitchCollection’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Fabric’, ‘SwitchCollection’]

4.15 SwordfishFabricConnectivityRightsforEthernet profile

4.15.1 Summary Information

- Version: 1.0.2
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish Connectivity Rights requirements using Fabrics, for implementations using Ethernet.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.FabricConnectivityRightsforEthernet

4.15.2 Protocol Information

- Protocol version: 1.18.0

4.15.3 Other Registry Information

4.15.4 Required profiles

The SwordfishFabricConnectivityRightsforEthernet profile depends upon the profiles listed in Table 43 .

Profile	Version	Conditions
SwordfishFabricConnectivityRights	1.0.2	

Table 43: Required profiles for SwordfishFabricConnectivityRightsforEthernet

4.15.5 Related resources

The SwordfishFabricConnectivityRightsforEthernet profile relies on the resources listed in Table 44

Resource	Purpose
ServiceRoot	
Chassis	

Resource	Purpose
Chassis	
NetworkAdapters	
ChassisCollection	
Endpoint	
IPTransportDetails	
TransportProtocol	TransportProtocol must exist and be set to 'Ethernet' Shall be equal to ['Ethernet']
Port	
EthernetInterface	
DHCPv4	
DHCPv6	
Description	
FQDN	
HostName	
IPv4Addresses	
IPv4StaticAddresses	
IPv6Addresses	
IPv6DefaultGateway	If IPv6Addresses is implemented, then IPv6DefaultGateway is required.
IPv6StaticAddresses	If IPv6Addresses is implemented, then IPv6StaticAddresses is required.
IPv6StaticDefaultGateways	If IPv6Addresses is implemented, then IPv6StaticDefaultGateways is required.
Id	
LinkStatus	
Links	
Chassis	
MACAddress	
Name	
PermanentMACAddress	
StatelessAddressAutoConfig	
Status	

Resource	Purpose
Health	
State	
EthernetInterfaceCollection	
NetworkAdapter	
Controllers	
ControllerCapabilities	
NetworkDeviceFunctionCount	
NetworkPortCount	
Links	
Ports	
Description	
Id	
Manufacturer	
Measurements	Deprecated, do not implement.
Model	
Name	
NetworkDeviceFunctions	
NetworkPorts	Deprecated, do not implement.
PartNumber	
Ports	
SKU	
SerialNumber	
NetworkAdapterCollection	
NetworkDeviceFunction	
AssignablePhysicalNetworkPorts	
BootMode	
DeviceEnabled	
Ethernet	
EthernetInterfaces	

Resource	Purpose
MACAddress	
MTUSize	
MUTSizeMaximum	
PermanentMACAddress	
Id	
Links	
PhysicalNetworkPortAssignment	
MaxVirtualFunctions	
NetDevFuncCapabilities	
VirtualFunctionsEnabled	
NetworkDeviceFunctionCollection	
Port	The Port object is used by both NetworkAdapter and Switch. Requirements for both are represented in this set of property requirements.
CurrentSpeedGbps	
Ethernet	
AssociatedMACAddresses	Optional for NetworkAdapter ports, Required for Switch ports.
EEEEnabled	Required for NetworkAdapter ports, optional for Switch ports.
FlowControlConfiguration	Required for NetworkAdapter ports, recommended for Switch ports.
FlowControlStatus	Required for NetworkAdapter ports, recommended for Switch ports.

Resource	Purpose
SupportedEthernetCapabilities	Deprecated. Do not implement.
WakeOnLANEnabled	Required for NetworkAdapter ports, Optional for Switch ports.
Id	
InterfaceEnabled	
LinkConfiguration	
AutoSpeedNegotiationCapable	
AutoSpeedNegotiationEnabled	
CapableLinkSpeedGbps	
ConfiguredNetworkLinks	
LinkNetworkTechnology	
LinkState	
LinkStatus	
Links	
AssociatedEndpoints	
LocationIndicatorActive	
MaxFrameSize	
MaxSpeedGbps	
Name	
PortId	
PortProtocol	PortProtocol must exist and be set to ‘Ethernet’ Shall be equal to [‘Ethernet’]
SFP	
Status	
State	
Type	

Resource	Purpose
Width	
PortCollection	
Switch	
DomainID	

Table 44: Related resources for SwordfishFabricConnectivityRightsforEthernet

4.15.6 Resource Manipulation

The SwordfishFabricConnectivityRightsforEthernet profile requires the resource manipulations listed in Table 45

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Chassis	M				
Chassis	M				
NetworkAdapters	M				
ChassisCollection	M				
Endpoint	M/C				
IPTransportDetails	M				
TransportProtocol	M				
Port	R				
EthernetInterface	M/C				
DHCPv4	R				
DHCPv6	R				
Description	M				
FQDN	M				
HostName	M				
IPv4Addresses	M				
IPv4StaticAddresses	M				
IPv6Addresses	R				
IPv6DefaultGateway	M/C				
IPv6StaticAddresses	M/C				
IPv6StaticDefaultGateways	M/C				

Resource	Read	Write	Create	Delete	Update
Id	M				
LinkStatus	M				
Links	M				
Chassis	M				
MACAddress	M				
Name	M				
PermanentMACAddress	R				
StatelessAddressAutoConfig	R				
Status	M				
Health	M				
State	M				
EthernetInterfaceCollection	M				
NetworkAdapter	M				
Controllers	M				
ControllerCapabilities	M				
NetworkDeviceFunctionCount	M				
NetworkPortCount	M				
Links	M				
Ports	R				
Description	M				
Id	M				
Manufacturer	M				
Measurements	DNI				
Model	M				
Name	M				
NetworkDeviceFunctions	M				
NetworkPorts	DNI				
PartNumber	M				
Ports	M				
SKU	R				
SerialNumber	M				

Resource	Read	Write	Create	Delete	Update
NetworkAdapterCollection	M				
NetworkDeviceFunction	M				
AssignablePhysicalNetworkPorts	M				
BootMode	M				
DeviceEnabled	M				
Ethernet	M				
EthernetInterfaces	M				
MACAddress	M				
MTUSize	M				
MUTSizeMaximum	R				
PermanentMACAddress	M				
Id	M				
Links	M				
PhysicalNetworkPortAssignment	M				
MaxVirtualFunctions	M				
NetDevFuncCapabilities	M				
VirtualFunctionsEnabled	M				
NetworkDeviceFunctionCollection	M				
Port	M				
CurrentSpeedGbps	M				
Ethernet	M				
AssociatedMACAddresses	R/C				
EEEEnabled	R/C				
FlowControlConfiguration	R/C				
FlowControlStatus	R/C				
SupportedEthernetCapabilities	DNI				
WakeOnLANEnabled	???				
Id	M				
InterfaceEnabled	M				

Resource	Read	Write	Create	Delete	Update
LinkConfiguration	M				
AutoSpeedNegotiationCapable	M				
AutoSpeedNegotiationEnabled	M				
CapableLinkSpeedGbps	M				
ConfiguredNetworkLinks	M				
LinkNetworkTechnology	M				
LinkState	M				
LinkStatus	M				
Links	M				
AssociatedEndpoints	M				
LocationIndicatorActive	R				
MaxFrameSize	R				
MaxSpeedGbps	M				
Name	M				
PortId	M				
PortProtocol	M				
SFP	IfImp				
Status	M				
State	M				
Type	M				
Width	M				
PortCollection	M				
Switch	M				
DomainID	M				

Table 45: Resource manipulations for SwordfishFabricConnectivityRightsforEthernet

4.15.7 Conditional Requirements

4.15.7.1 Endpoint

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘EndpointCollection’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘Fabric’, ‘EndpointCollection’]
- ReadRequirement: Mandatory

4.15.7.2 EthernetInterface

Test Condition / Requirement

When subordinate to [‘Manager’, ‘EthernetInterfaceCollection’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘Manager’, ‘EthernetInterfaceCollection’]
- ReadRequirement: Mandatory

4.15.7.3 IPv6DefaultGateway

Purpose

If IPv6Addresses is implemented, then IPv6DefaultGateway is required.

Test Condition / Requirement

if IPv6Addresses is Present, ReadRequirement is Mandatory

based on:

- CompareProperty: IPv6Addresses
- CompareType: Present
- ReadRequirement: Mandatory

4.15.7.4 IPv6StaticAddresses

Purpose

If IPv6Addresses is implemented, then IPv6StaticAddresses is required.

Test Condition / Requirement

if IPv6Addresses is Present, ReadRequirement is Mandatory

based on:

- CompareProperty: IPv6Addresses

- CompareType: Present
- ReadRequirement: Mandatory

4.15.7.5 IPv6StaticDefaultGateways

Purpose

If IPv6Addresses is implemented, then IPv6StaticDefaultGateways is required.

Test Condition / Requirement

if IPv6Addresses is Present, ReadRequirement is Mandatory

based on:

- CompareProperty: IPv6Addresses
- CompareType: Present
- ReadRequirement: Mandatory

4.15.7.6 Port

Test Condition / Requirement

When subordinate to ['NetworkAdapter'], ReadRequirement is mandatory

based on:

- SubordinatetoResource ['NetworkAdapter']

Test Condition / Requirement

When subordinate to ['Fabric', 'SwitchCollection', 'Switch'], ReadRequirement is mandatory

based on:

- SubordinatetoResource ['Fabric', 'SwitchCollection', 'Switch']

4.15.7.7 AssociatedMACAddresses

Test Condition / Requirement

When subordinate to ['NetworkAdapter'], ReadRequirement is None

based on:

- SubordinatetoResource ['NetworkAdapter']
- ReadRequirement: None

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘SwitchCollection’, ‘Switch’], ReadRequirement is Recommended

based on:

- SubordinatetoResource [‘Fabric’, ‘SwitchCollection’, ‘Switch’]
- ReadRequirement: Recommended

4.15.7.8 EEEEnabled

Test Condition / Requirement

When subordinate to [‘NetworkAdapter’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘NetworkAdapter’]
- ReadRequirement: Mandatory

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘SwitchCollection’, ‘Switch’], ReadRequirement is Recommended

based on:

- SubordinatetoResource [‘Fabric’, ‘SwitchCollection’, ‘Switch’]
- ReadRequirement: Recommended

4.15.7.9 FlowControlConfiguration

Test Condition / Requirement

When subordinate to [‘NetworkAdapter’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘NetworkAdapter’]
- ReadRequirement: Mandatory

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘SwitchCollection’, ‘Switch’], ReadRequirement is Recommended

based on:

- SubordinatetoResource [‘Fabric’, ‘SwitchCollection’, ‘Switch’]
- ReadRequirement: Recommended

4.15.7.10 FlowControlStatus

Test Condition / Requirement

When subordinate to [‘NetworkAdapter’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘NetworkAdapter’]
- ReadRequirement: Mandatory

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘SwitchCollection’, ‘Switch’], ReadRequirement is Recommended

based on:

- SubordinatetoResource [‘Fabric’, ‘SwitchCollection’, ‘Switch’]
- ReadRequirement: Recommended

4.15.7.11 WakeOnLANEnabled

Test Condition / Requirement

When subordinate to [‘NetworkAdapter’], ReadRequirement is Mandatory

based on:

- SubordinatetoResource [‘NetworkAdapter’]
- ReadRequirement: Mandatory

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘SwitchCollection’, ‘Switch’], ReadRequirement is None

based on:

- SubordinatetoResource [‘Fabric’, ‘SwitchCollection’, ‘Switch’]
- ReadRequirement: None

4.16 SwordfishFabricConnectivityRightsforRDMA profile

4.16.1 Summary Information

- Version: 1.0.2
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish Connectivity Rights requirements using Fabrics, for implementations using RDMA.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.FabricConnectivityRightsforRDMA

4.16.2 Protocol Information

- Protocol version: 1.18.0

4.16.3 Other Registry Information

4.16.4 Required profiles

The SwordfishFabricConnectivityRightsforRDMA profile depends upon the profiles listed in Table 46 .

Profile	Version	Conditions
SwordfishFabricConnectivityRights	1.0.2	

Table 46: Required profiles for SwordfishFabricConnectivityRightsforRDMA

4.16.5 Related resources

The SwordfishFabricConnectivityRightsforRDMA profile relies on the resources listed in Table 47

Resource	Purpose
ServiceRoot	
Chassis	

Resource	Purpose
Chassis	
NetworkAdapters	
ChassisCollection	
Endpoint	
IPTransportDetails	
TransportProtocol	TransportProtocol must exist and be set to 'InfiniBand' Shall be equal to ['InfiniBand']
Port	
NetworkAdapter	
Controllers	
ControllerCapabilities	
NetworkDeviceFunctionCount	
NetworkPortCount	
Description	
Id	
Manufacturer	
Measurements	Deprecated, do not implement.
Model	
Name	
NetworkDeviceFunctions	
NetworkPorts	Deprecated, do not implement.
PartNumber	
Ports	
SKU	
SerialNumber	
NetworkAdapterCollection	
NetworkDeviceFunction	
AssignablePhysicalNetworkPorts	
AssignableNetworkPorts	Deprecated, do not implement.

Resource	Purpose
DeviceEnabled	
Id	
InfiniBand	
MTUSize	
NodeGUID	
PermanentNodeGUID	
PermanentPortGUID	
PermanentSystemGUID	
PortGUID	
SupportedMTUSizes	
SystemGUID	
Links	
PhysicalNetworkPortAssignment	
MaxVirtualFunctions	
NetDevFuncCapabilities	
VirtualFunctionsEnabled	
NetworkDeviceFunctionCollection	
Port	The Port object is used by both NetworkAdapter and Switch. Requirements for both are represented in this set of property requirements.
CurrentSpeedGbps	
Id	
InterfaceEnabled	
LinkConfiguration	
AutoSpeedNegotiationCapable	
AutoSpeedNegotiationEnabled	
CapableLinkSpeedGbps	

Resource	Purpose
ConfiguredNetworkLinks	
LinkNetworkTechnology	
LinkState	
LinkStatus	
Links	
AssociatedEndpoints	
LocationIndicatorActive	
MaxFrameSize	
MaxSpeedGbps	
Name	
PortId	
PortProtocol	PortProtocol must exist and be set to 'InfiniBand' Shall be equal to ['InfiniBand']
SFP	
Status	
State	
Type	
Width	
PortCollection	
Switch	
DomainID	

Table 47: Related resources for SwordfishFabricConnectivityRightsforRDMA

4.16.6 Resource Manipulation

The SwordfishFabricConnectivityRightsforRDMA profile requires the resource manipulations listed in Table 48

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Chassis	M				
Chassis	M				

Resource	Read	Write	Create	Delete	Update
NetworkAdapters	M				
ChassisCollection	M				
Endpoint	M/C				
IPTransportDetails	M				
TransportProtocol	M				
Port	R				
NetworkAdapter	M				
Controllers	M				
ControllerCapabilities	M				
NetworkDeviceFunctionCount	M				
NetworkPortCount	M				
Description	M				
Id	M				
Manufacturer	M				
Measurements	DNI				
Model	M				
Name	M				
NetworkDeviceFunctions	M				
NetworkPorts	DNI				
PartNumber	M				
Ports	M				
SKU	R				
SerialNumber	M				
NetworkAdapterCollection	M				
NetworkDeviceFunction	M				
AssignablePhysicalNetworkPorts	M				
AssignableNetworkPorts	DNI				
DeviceEnabled	M				
Id	M				
InfiniBand	M				
MTUSize	M				

Resource	Read	Write	Create	Delete	Update
NodeGUID	M				
PermanentNodeGUID	M				
PermanentPortGUID	M				
PermanentSystemGUID	M				
PortGUID	M				
SupportedMTUSizes	R				
SystemGUID	M				
Links	M				
PhysicalNetworkPortAssignment	M				
MaxVirtualFunctions	M				
NetDevFuncCapabilities	M				
VirtualFunctionsEnabled	M				
NetworkDeviceFunctionCollection	M				
Port	M				
CurrentSpeedGbps	M				
Id	M				
InterfaceEnabled	M				
LinkConfiguration	M				
AutoSpeedNegotiationCapable	M				
AutoSpeedNegotiationEnabled	M				
CapableLinkSpeedGbps	M				
ConfiguredNetworkLinks	M				
LinkNetworkTechnology	M				
LinkState	M				
LinkStatus	M				
Links	M				
AssociatedEndpoints	M				
LocationIndicatorActive	R				
MaxFrameSize	R				

Resource	Read	Write	Create	Delete	Update
MaxSpeedGbps	M				
Name	M				
PortId	M				
PortProtocol	M				
SFP	IfImp				
Status	M				
State	M				
Type	M				
Width	M				
PortCollection	M				
Switch	M				
DomainID	M				

Table 48: Resource manipulations for SwordfishFabricConnectivityRightsforRDMA

4.16.7 Conditional Requirements

4.16.7.1 Endpoint

Test Condition / Requirement

When subordinate to ['Fabric', 'EndpointCollection'], ReadRequirement is Mandatory

based on:

- SubordinatetoResource ['Fabric', 'EndpointCollection']
- ReadRequirement: Mandatory

4.16.7.2 Port

Test Condition / Requirement

When subordinate to ['NetworkAdapter'], ReadRequirement is mandatory

based on:

- SubordinatetoResource ['NetworkAdapter']

Test Condition / Requirement

When subordinate to [‘Fabric’, ‘SwitchCollection’, ‘Switch’], ReadRequirement is mandatory based on:

- SubordinatetoResource [‘Fabric’, ‘SwitchCollection’, ‘Switch’]

4.17 SwordfishFileCapacityManagement profile

4.17.1 Summary Information

- Version: 1.1.4
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish File Capacity Management requirements.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.File.CapacityManagement

4.17.2 Protocol Information

No protocol information defined in this profile.

4.17.3 Other Registry Information

4.17.4 Required profiles

The SwordfishFileCapacityManagement profile depends upon the profiles listed in Table 49 .

Profile	Version	Conditions
SwordfishFileProvisioning	1.1.4	

Table 49: Required profiles for SwordfishFileCapacityManagement

4.17.5 Related resources

The SwordfishFileCapacityManagement profile relies on the resources listed in Table 50

Resource	Purpose
FileSystemCollection	
FileSystem	
Capacity	

Table 50: Related resources for SwordfishFileCapacityManagement

4.17.6 Resource Manipulation

The SwordfishFileCapacityManagement profile requires the resource manipulations listed in Table 51

Resource	Read	Write	Create	Delete	Update
FileSystemCollection	IfImp				T
FileSystem	M				
Capacity	M	M			

Table 51: Resource manipulations for SwordfishFileCapacityManagement

4.17.7 Conditional Requirements

None defined.

4.18 SwordfishFileProvisioning profile

4.18.1 Summary Information

- Version: 1.1.4
- Contact Information: SNIA.org
- Purpose: Defines File Provisioning requirements for service-based Swordfish implementations.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.File.Provisioning

4.18.2 Protocol Information

No protocol information defined in this profile.

4.18.3 Other Registry Information

4.18.4 Required profiles

The SwordfishFileProvisioning profile depends upon the profiles listed in Table 52 .

Profile	Version	Conditions
CoSDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.
SwordfishDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.

Table 52: Required profiles for SwordfishFileProvisioning

4.18.5 Related resources

The SwordfishFileProvisioning profile relies on the resources listed in Table 53

Resource	Purpose
FileSystemCollection	

Resource	Purpose
Members	MinCount = 0
Filesystem	
ExportedShares	
ImportedShares	Require the Imported Shares structure to be complete and Create / Delete if implemented.
ImportedShare	
FileSharePath	

Table 53: Related resources for SwordfishFileProvisioning

4.18.6 Resource Manipulation

The SwordfishFileProvisioning profile requires the resource manipulations listed in Table 54

Resource	Read	Write	Create	Delete	Update
FileSystemCollection	M		T	T	
Members	M				
Filesystem	IfImp				
ExportedShares	IfImp	M			
ImportedShares	IfImp	M			
ImportedShare	M				
FileSharePath	M				

Table 54: Resource manipulations for SwordfishFileProvisioning

4.18.7 Conditional Requirements

4.18.7.1 FileSystemCollection

Test Condition / Requirement

When subordinate to [‘Storage’, ‘StorageService’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Storage’, ‘StorageService’]

4.19 SwordfishIOPerformance profile

4.19.1 Summary Information

- Version: 1.2.0
- Contact Information: SNIA.org
- Purpose: Defines Swordfish IO Performance requirements.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.IOPerformance

4.19.2 Protocol Information

No protocol information defined in this profile.

4.19.3 Other Registry Information

4.19.4 Required profiles

The SwordfishIOPerformance profile depends upon the profiles listed in Table 55 .

Profile	Version	Conditions
CoSDiscovery	1.2.0	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.
SwordfishDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.

Table 55: Required profiles for SwordfishIOPerformance

4.19.5 Related resources

The SwordfishIOPerformance profile relies on the resources listed in Table 56

Resource	Purpose
Volume	

Resource	Purpose
Metrics	
StoragePool	
Metrics	
FileSystem	
Metrics	
StorageService	
Metrics	
VolumeMetrics	
IOStatistics	
WriteIORequestTime	
WriteIORequests	
WriteIOKiBytes	
WriteHitIORequests	
ReadIORequestTime	
ReadIORequests	
ReadIOKiBytes	
ReadHitIORequests	
NonIORequestTime	
NonIORequests	
StoragePoolMetrics	
IOStatistics	
WriteIORequestTime	
WriteIORequests	
WriteIOKiBytes	
WriteHitIORequests	
ReadIORequestTime	
ReadIORequests	
ReadIOKiBytes	
ReadHitIORequests	
NonIORequestTime	
NonIORequests	
StorageServiceMetrics	

Resource	Purpose
IOStatistics	
WriteIORequestTime	
WriteIORequests	
WriteIOKiBytes	
WriteHitIORequests	
ReadIORequestTime	
ReadIORequests	
ReadIOKiBytes	
ReadHitIORequests	
NonIORequestTime	
NonIORequests	
FileSystemMetrics	
IOStatistics	
WriteIORequestTime	
WriteIORequests	
WriteIOKiBytes	
WriteHitIORequests	
ReadIORequestTime	
ReadIORequests	
ReadIOKiBytes	
ReadHitIORequests	
NonIORequestTime	
NonIORequests	

Table 56: Related resources for SwordfishIOPerformance

4.19.6 Resource Manipulation

The SwordfishIOPerformance profile requires the resource manipulations listed in Table 57

Resource	Read	Write	Create	Delete	Update
Volume	IfImp				
Metrics	M				

Resource	Read	Write	Create	Delete	Update
StoragePool	IfImp				
Metrics	M				
FileSystem	IfImp				
Metrics	M				
StorageService	IfImp				
Metrics	M				
VolumeMetrics	IfImp				
IOStatistics	M				
WriteIORequestTime	M				
WriteIORequests	M				
WriteIOKiBytes	M				
WriteHitIORequests	M				
ReadIORequestTime	M				
ReadIORequests	M				
ReadIOKiBytes	M				
ReadHitIORequests	M				
NonIORequestTime	M				
NonIORequests	M				
StoragePoolMetrics	IfImp				
IOStatistics	M				
WriteIORequestTime	M				
WriteIORequests	M				
WriteIOKiBytes	M				
WriteHitIORequests	M				
ReadIORequestTime	M				
ReadIORequests	M				
ReadIOKiBytes	M				
ReadHitIORequests	M				
NonIORequestTime	M				
NonIORequests	M				
StorageServiceMetrics	IfImp				
IOStatistics	M				

Resource	Read	Write	Create	Delete	Update
WriteIORequestTime	M				
WriteIORequests	M				
WriteIOKiBytes	M				
WriteHitIORequests	M				
ReadIORequestTime	M				
ReadIORequests	M				
ReadIOKiBytes	M				
ReadHitIORequests	M				
NonIORequestTime	M				
NonIORequests	M				
FileSystemMetrics	IfImp				
IOStatistics	M				
WriteIORequestTime	M				
WriteIORequests	M				
WriteIOKiBytes	M				
WriteHitIORequests	M				
ReadIORequestTime	M				
ReadIORequests	M				
ReadIOKiBytes	M				
ReadHitIORequests	M				
NonIORequestTime	M				
NonIORequests	M				

Table 57: Resource manipulations for SwordfishIOPerformance

4.19.7 Conditional Requirements

None defined.

4.20 SwordfishManagementController profile

4.20.1 Summary Information

- Version: 1.0.2
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish Management Controller requirements for configurations such as embedded management controllers.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.ManagementController

4.20.2 Protocol Information

- Protocol version: 1.18.0

4.20.3 Other Registry Information

4.20.4 Required profiles

4.20.5 Related resources

The SwordfishManagementController profile relies on the resources listed in Table 58

Resource	Purpose
ServiceRoot	
Managers	
EthernetInterface	
DHCPv4	
DHCPv6	
Description	
FQDN	
HostName	
IPv4Addresses	

Resource	Purpose
IPv4StaticAddresses	
IPv6Addresses	
IPv6DefaultGateway	If IPv6Addresses is implemented, then IPv6DefaultGateway is required.
IPv6StaticAddresses	If IPv6Addresses is implemented, then IPv6StaticAddresses is required.
IPv6StaticDefaultGateways	If IPv6Addresses is implemented, then IPv6StaticDefaultGateways is required.
Id	
LinkStatus	
Links	
Chassis	
MACAddress	
Name	
PermanentMACAddress	
StatelessAddressAutoConfig	
Status	
Health	
State	
EthernetInterfaceCollection	
Manager	
ACTIONS	
Reset()	
DateTime	
DateTimeLocalOffset	
Description	
EthernetInterfaces	
FirmwareVersion	
Id	
Links	
ManagerForServers	Must have one of the 'ManagerFor' implemented.
ManagerForChassis	Must have one of the 'ManagerFor' implemented.

Resource	Purpose
ManagerForSwitches	Must have one of the 'ManagerFor' implemented.
ManagerForManagers	Must have one of the 'ManagerFor' implemented.
LocationIndicatorActive	
ManagerType	
Manufacturer	
Model	
Name	
NetworkProtocol	
SerialNumber	
ServiceEntryPointUUID	
Status	
Health	
State	
VirtualMedia	Deprecated. Do not implement.
ManagerCollection	
ManagerNetworkProtocol	
Description	
HTTP	
HTTPS	
Id	
NTP	
NTPServers	
Port	
Name	
Proxy	
SSDP	
NotifyIPv6Scope	Required if IPv6 is implemented.
NotifyMulticastIntervalSeconds	
NotifyTTL	
Port	
ProtocolEnabled	

Table 58: Related resources for SwordfishManagementController

4.20.6 Resource Manipulation

The SwordfishManagementController profile requires the resource manipulations listed in Table 59

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Managers	M				
EthernetInterface	M/C				
DHCPv4	R				
DHCPv6	R				
Description	M				
FQDN	M				
HostName	M				
IPv4Addresses	M				
IPv4StaticAddresses	M				
IPv6Addresses	R				
IPv6DefaultGateway	M/C				
IPv6StaticAddresses	M/C				
IPv6StaticDefaultGateways	M/C				
Id	M				
LinkStatus	M				
Links	M				
Chassis	M				
MACAddress	M				
Name	M				
PermanentMACAddress	R				
StatelessAddressAutoConfig	R				
Status	M				
Health	M				
State	M				
EthernetInterfaceCollection	M				

Resource	Read	Write	Create	Delete	Update
Manager	M				
ACTIONS					
Reset	R				
DateTime	M				
DateTimeLocalOffset	M				
Description	M				
EthernetInterfaces	M				
FirmwareVersion	M				
Id	M				
Links	M				
ManagerForServers	M/C				
ManagerForChassis	M/C				
ManagerForSwitches	M/C				
ManagerForManagers	M/C				
LocationIndicatorActive	M				
ManagerType	M				
Manufacturer	M				
Model	M				
Name	M				
NetworkProtocol	M				
SerialNumber	M				
ServiceEntryPointUUID	M				
Status	M				
Health	M				
State	M				
VirtualMedia	DNI				
ManagerCollection	M				
ManagerNetworkProtocol	M				
Description	M				
HTTP	M				
HTTPS	M				
Id	M				

Resource	Read	Write	Create	Delete	Update
NTP	M				
NTPServers	M				
Port	M				
Name	M				
Proxy	M				
SSDP	M				
NotifyIPv6Scope	R				
NotifyMulticastIntervalSeconds	M				
NotifyTTL	M				
Port	M				
ProtocolEnabled	M				

Table 59: Resource manipulations for SwordfishManagementController

4.20.7 Conditional Requirements

4.20.7.1 EthernetInterface

Test Condition / Requirement

When subordinate to ['Manager', 'EthernetInterfaceCollection'], ReadRequirement is Mandatory

based on:

- SubordinatetoResource ['Manager', 'EthernetInterfaceCollection']
- ReadRequirement: Mandatory

4.20.7.2 IPv6DefaultGateway

Purpose

If IPv6Addresses is implemented, then IPv6DefaultGateway is required.

Test Condition / Requirement

if IPv6Addresses is Present, ReadRequirement is Mandatory

based on:

- CompareProperty: IPv6Addresses

- CompareType: Present
- ReadRequirement: Mandatory

4.20.7.3 IPv6StaticAddresses

Purpose

If IPv6Addresses is implemented, then IPv6StaticAddresses is required.

Test Condition / Requirement

if IPv6Addresses is Present, ReadRequirement is Mandatory

based on:

- CompareProperty: IPv6Addresses
- CompareType: Present
- ReadRequirement: Mandatory

4.20.7.4 IPv6StaticDefaultGateways

Purpose

If IPv6Addresses is implemented, then IPv6StaticDefaultGateways is required.

Test Condition / Requirement

if IPv6Addresses is Present, ReadRequirement is Mandatory

based on:

- CompareProperty: IPv6Addresses
- CompareType: Present
- ReadRequirement: Mandatory

4.20.7.5 ManagerForServers

Purpose

Must have one of the ‘ManagerFor’ implemented.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.20.7.6 ManagerForChassis

Purpose

Must have one of the 'ManagerFor' implemented.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.20.7.7 ManagerForSwitches

Purpose

Must have one of the 'ManagerFor' implemented.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.20.7.8 ManagerForManagers

Purpose

Must have one of the 'ManagerFor' implemented.

Test Condition / Requirement

if is Absent, ReadRequirement is Mandatory

based on:

- CompareType: Absent
- ReadRequirement: Mandatory

4.21 SwordfishNVMeDrive profile

4.21.1 Summary Information

- Version: 1.3.0
- Contact Information: SNIA.org
- Purpose: Define requirements for resources and relationships that are common to all Swordfish conformant implementations of NVMe Drives.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.NVMeDrive

4.21.2 Protocol Information

- Protocol version: 1.18.0

4.21.3 Other Registry Information

4.21.4 Required profiles

4.21.5 Related resources

The SwordfishNVMeDrive profile relies on the resources listed in Table 60

Resource	Purpose
ServiceRoot	
Chassis	
Storage	
Systems	If Storage is attached to a ComputerSystem rather than instantiated as a Standalone instance.
StorageSystems	If Storage is attached to a ComputerSystem rather than instantiated as a Standalone instance.
DriveCollection	
Members	MinCount = 1

Resource	Purpose
StorageCollection	
Members	MinCount = 1
VolumeCollection	
StorageControllerCollection	
Members	MinCount = 1
Assembly	
BinaryDataURI	
Chassis	
Drives	
Drive	
ACTIONS	
Reset()	This action resets this drive.
Assembly	The `BinaryDataURI` property within the Assembly resource in Drive is recommended to be implemented to support mapping a binary blob via NVMe-MI to the VPD.
BlockSizeBytes	
CapableSpeedGbs	
CapacityBytes	
Description	
EncryptionAbility	
EncryptionStatus	Required when encryption is supported (EncryptionAbility is set to anything other than none).
FailurePredicted	Recommended when controller type is Admin or IO.
Id	
Identifiers	
DurableName	
DurableNameFormat	
IndicatorLED	This property is recommended as 'Do not implement' for this device and protocol type.
Links	
Volumes	
Location	This property is recommended as 'Do not implement' for this device and protocol type. This property has been deprecated in

Resource	Purpose
	favor of the PhysicalLocation property.
LocationIndicatorActive	
Manufacturer	
MediaType	
Model	
Multipath	
Name	
NegotiatedSpeedGbs	
PhysicalLocation	
PartLocation	
PredictedMediaLifeLeftPercent	
Protocol	
PowerMode	Required when MediaType != SSD. If SSD, do not implement.
Revision	
RotationSpeedRPM	Required when MediaType != SSD. If SSD, do not implement.
SKU	
SerialNumber	
Status	
State	
Health	
StatusIndicator	
WriteCacheEnabled	
Storage	
ACTIONS	
SetEncryptionKey()	This property is recommended as 'Do not implement' for this device and protocol type.
Controllers	
Description	Shall be equal to ['An NVM Express Subsystem is an NVMe device that contains one or more NVM Express controllers and may contain one or more namespaces.']
Drives	
Id	

Resource	Purpose
Identifiers	
DurableName	
DurableNameFormat	
Links	
Enclosures	This resource shall contain the link to the chassis with the Drive object.
SimpleStorage	This resource is recommended as 'Do not implement' for this device and protocol type.
Name	
Status	
Health	
State	This property is recommended as 'Do not implement' for this device and protocol type.
HealthRollup	
StorageControllers	This resource is recommended as 'Do not implement' for this device and protocol type. Use the 'Controller' resource instead.
StorageGroups	This resource is recommended as 'Do not implement' for this device and protocol type. Note that this function has been deprecated and replaced by Connections in Fabrics.
Volumes	
StorageController	
Assembly	This property is recommended as 'Do not implement' for this device and protocol type. This information is available in the Drive resource if needed.
AssetTag	This property is recommended as 'Do not implement' for this device and protocol type. This information is available in the Drive resource if needed.
CacheSummary	This property is recommended as 'Do not implement' for this device and protocol type.
ControllerRates	This property is recommended as 'Do not implement' for this device and protocol type.
Description	The Description must be set to the designated string when the ControllerType is Admin. There is a different specified value for IO and Discovery. The Description must be set to the designated string when the

Resource	Purpose
	ControllerType is Discovery. There is a different specified value for Admin and IO. The Description must be set to the designated string when the ControllerType is IO. There is a different specified value for Admin and Discovery.
FirmwareVersion	
Links	
AttachedVolumes	AttachedVolumes must be implemented when the ControllerType is IO. Do not implement when ControllerType is Discovery or Admin.
Location	This resource is recommended as 'Do not implement' for this device and protocol type. Implemented in the 'Drive' object as 'PhysicalLocation'.
Manufacturer	Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported. Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported. Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported.
Model	Model must be implemented when the ControllerType is IO. It is recommended to implement when ControllerType is Admin, and may be implemented for a Discovery controller. Model must be implemented when the ControllerType is IO. It is recommended to implement when ControllerType is Admin, and may be implemented for a Discovery controller.
NVMeControllerProperties	
ControllerType	
NVMeVersion	NVMeVersion is required when the underlying NVMe implementation is revision 1.2 or higher.
MaxQueueSize	

Resource	Purpose
NVMeControllerAttributes	
ReportsNamespaceGranularity	
Supports128BitHostId	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
NVMeSMARTCriticalWarnings	
MediaInReadOnly	
OverallSubsystemDegraded	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
PMRUnreliable	
PowerBackupFailed	Recommended when controller type is Admin or IO. Do not implement for Discovery controllers.
SpareCapacityWornOut	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
SpeedGbps	This resource is recommended as 'Do not implement' for this device and protocol type.
SKU	This resource is recommended as 'Do not implement' for this device and protocol type.
SupportedControllerProtocols	
SupportedDeviceProtocols	This resource is recommended as 'Do not implement' for this device and protocol type.
SupportedRAIDTypes	
PCIeInterface	This property is required for implementations that connect via PCIe (e.g., do not implement for Ethernet-Attach drives).
PCIeType	
MaxPCIeType	
LanesInUse	
MaxLanes	
Volume	
BlockSizeBytes	
Capacity	
Data	
AllocatedBytes	

Resource	Purpose
ConsumedBytes	
DisplayName	
Description	Shall be equal to ['A Namespace is a quantity of non-volatile memory that may be formatted into logical blocks. When formatted, a namespace of size n is a collection of logical blocks with logical block addresses from 0 to (n-1). NVMe systems can support multiple namespaces.']
Id	
Identifiers	
DurableName	
DurableNameFormat	
InitializeMethod	
LogicalUnitNumber	This resource is recommended as 'Do not implement' for this device and protocol type.
NVMeNamespaceProperties	
IsShareable	
NVMeVersion	
NamespaceFeatures	
SupportsDeallocatedOrUnwrittenLBError	
SupportsNGUIDReuse	
SupportsAtomicTransactionSize	
SupportsIOPerformanceHints	
NamespaceId	
MetadataTransferredAtEndOfDataLBA	
Links	
Controllers	This resource shall contain links to the associated controllers.
Drives	This resource shall contain the link to the Drive object.
Name	
OptimumIOSizeBytes	
ProvisioningPolicy	Recommended to implement when device supports thin provisioning.

Resource	Purpose
RAIDType	
Status	
Health	
State	
HealthRollup	This property is recommended as 'Do not implement' for this device and protocol type.
StorageGroups	This resource is recommended as 'Do not implement' for this device and protocol type. Note that this function has been deprecated and replaced by Connections in Fabrics.
VolumeType	This property is recommended as 'Do not implement' for this device and protocol type. It has been deprecated and replaced by RAIDType.
WriteCachePolicy	

Table 60: Related resources for SwordfishNVMeDrive

4.21.6 Resource Manipulation

The SwordfishNVMeDrive profile requires the resource manipulations listed in Table 61

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Chassis	M				
Storage	M				
Systems	IfImp				
StorageSystems	IfImp				
DriveCollection	M				
Members	M				
StorageCollection	M				
Members	M				
VolumeCollection	M				
StorageControllerCollection	M				
Members	M				
Assembly	R/C				

Resource	Read	Write	Create	Delete	Update
BinaryDataURI	R				
Chassis	M				
Drives	M				
Drive	M				
ACTIONS					
Reset	M				
SecureErase	M				
Assembly	R				
BlockSizeBytes	M				
CapableSpeedGbs	M				
CapacityBytes	M				
Description	M				
EncryptionAbility	M				
EncryptionStatus	M/C				
FailurePredicted	R/C				
Id	M				
Identifiers	M				
DurableName	M				
DurableNameFormat	M				
IndicatorLED	DNI				
Links	M				
Volumes	M				
Location	DNI				
LocationIndicatorActive	R				
Manufacturer	M				
MediaType	M				
Model	M				
Multipath	R				
Name	M				
NegotiatedSpeedGbs	M				
PhysicalLocation	M				
PartLocation	M				

Resource	Read	Write	Create	Delete	Update
PredictedMediaLifeLeftPercent	M				
Protocol	M				
PowerMode	M/C				
Revision	M				
RotationSpeedRPM	M/C				
SKU	M				
SerialNumber	M				
Status	M				
State	M				
Health	M				
StatusIndicator	M				
WriteCacheEnabled	M				
Storage	M				
ACTIONS					
SetEncryptionKey	DNI				
Controllers	M				
Description	M				
Drives	M				
Id	M				
Identifiers	M				
DurableName	M				
DurableNameFormat	M				
Links	M				
Enclosures	M				
SimpleStorage	DNI				
Name	M				
Status	M				
Health	M				
State	DNI				
HealthRollup	M				
StorageControllers	DNI				
StorageGroups	DNI				

Resource	Read	Write	Create	Delete	Update
Volumes	M				
StorageController	M				
Assembly	DNI				
AssetTag	DNI				
CacheSummary	DNI				
ControllerRates	DNI				
Description	M				
FirmwareVersion	M				
Links	M				
AttachedVolumes	M/C	M/C			
Location	DNI				
Manufacturer	???				
Model	R/C				
NVMeControllerProperties	M				
ControllerType	M				
NVMeVersion	IfImp				
MaxQueueSize	M				
NVMeControllerAttributes	M				
ReportsNamespaceGranularity	R				
Supports128BitHostId	M/C				
NVMeSMARTCriticalWarnings	M				
MediaInReadOnly	M				
OverallSubsystemDegraded	M/C				
PMRUnreliable	R				
PowerBackupFailed	R/C				
SpareCapacityWornOut	M/C				
SpeedGbps	DNI				
SKU	DNI				
SupportedControllerProtocols	M				
SupportedDeviceProtocols	DNI				
SupportedRAIDTypes	M				

Resource	Read	Write	Create	Delete	Update
PCIeInterface	IfImp				
PCIeType	M				
MaxPCIeType	M				
LanesInUse	M				
MaxLanes	M				
Volume	M				
BlockSizeBytes	M				
Capacity	M				
Data	M				
AllocatedBytes	M				
ConsumedBytes	M				
DisplayName	M				
Description	M				
Id	M				
Identifiers	M				
DurableName	M				
DurableNameFormat	M				
InitializeMethod	R				
LogicalUnitNumber	DNI				
NVMeNamespaceProperties	M				
IsShareable	M				
NVMeVersion	M				
NamespaceFeatures	M				
SupportsDeallocatedOrUnwrittenLBError	M				
SupportsNGUIDReuse	M				
SupportsAtomicTransactionSize	M				
SupportsIOPerformanceHints	M				
NamespaceId	M				
MetadataTransferredAtEndOfDataLBA	M				

Resource	Read	Write	Create	Delete	Update
Links	M				
Controllers	M				
Drives	M				
Name	M				
OptimumIOSizeBytes	R				
ProvisioningPolicy	R				
RAIDType	M				
Status	M				
Health	M				
State	M				
HealthRollup	DNI				
StorageGroups	DNI				
VolumeType	DNI				
WriteCachePolicy	R				

Table 61: Resource manipulations for SwordfishNVMeDrive

4.21.7 Conditional Requirements

4.21.7.1 Assembly

Test Condition / Requirement

When subordinate to [‘Drive’], ReadRequirement is Recommended

based on:

- SubordinatetoResource [‘Drive’]
- ReadRequirement: Recommended

4.21.7.2 EncryptionStatus

Purpose

Required when encryption is supported (EncryptionAbility is set to anything other than none).

Test Condition / Requirement

if EncryptionAbility is NotEqual to [‘None’], ReadRequirement is Mandatory

based on:

- CompareProperty: EncryptionAbility
- CompareType: NotEqual
- CompareValues: [‘None’]
- ReadRequirement: Mandatory

4.21.7.3 FailurePredicted

Purpose

Recommended when controller type is Admin or IO.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Recommended

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Recommended

4.21.7.4 PowerMode

Purpose

Required when MediaType != SSD. If SSD, do not implement.

Test Condition / Requirement

if MediaType is NotEqual to [‘SSD’], ReadRequirement is Mandatory

based on:

- CompareProperty: MediaType
- CompareType: NotEqual
- CompareValues: [‘SSD’]
- ReadRequirement: Mandatory

4.21.7.5 RotationSpeedRPM

Purpose

Required when MediaType != SSD. If SSD, do not implement.

Test Condition / Requirement

if MediaType is NotEqual to [‘SSD’], ReadRequirement is Mandatory

based on:

- CompareProperty: MediaType
- CompareType: NotEqual
- CompareValues: [‘SSD’]
- ReadRequirement: Mandatory

4.21.7.6 Description

Purpose

The Description must be set to the designated string when the ControllerType is Admin. There is a different specified value for IO and Discovery.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’], ReadRequirement is mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’]

Purpose

The Description must be set to the designated string when the ControllerType is Discovery. There is a different specified value for Admin and IO.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Discovery’], ReadRequirement is mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Discovery’]

Purpose

The Description must be set to the designated string when the ControllerType is IO. There is a different specified value for Admin and Discovery.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘IO’], ReadRequirement is mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘IO’]

4.21.7.7 AttachedVolumes

Purpose

AttachedVolumes must be implemented when the ControllerType is IO. Do not implement when ControllerType is Discovery or Admin.

Test Condition / Requirement

if ControllerType is Equal to [‘IO’], WriteRequirement is Mandatory, ReadRequirement is Mandatory

based on:

- CompareProperty: ControllerType
- CompareType: Equal
- CompareValues: [‘IO’]
- WriteRequirement: Mandatory
- ReadRequirement: Mandatory

4.21.7.8 Manufacturer

Purpose

Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘IO’]
- ReadRequirement: Mandatory

Purpose

Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to ['Discovery'], ReadRequirement is Recommended

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: ['Discovery']
- ReadRequirement: Recommended

Purpose

Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to ['Admin'], ReadRequirement is None

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: ['Admin']
- ReadRequirement: None

4.21.7.9 Model

Purpose

Model must be implemented when the ControllerType is IO. It is recommended to implement when ControllerType is Admin, and may be implemented for a Discovery controller.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to ['IO'], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType

- CompareType: Equal
- CompareValues: [‘IO’]
- ReadRequirement: Mandatory

Purpose

Model must be implemented when the ControllerType is IO. It is recommended to implement when ControllerType is Admin, and may be implemented for a Discovery controller.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Discovery’], ReadRequirement is Recommended

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Discovery’]
- ReadRequirement: Recommended

4.21.7.10 Supports128BitHostId

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.21.7.11 OverallSubsystemDegraded

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.21.7.12 PowerBackupFailed

Purpose

Recommended when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Recommended

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Recommended

4.21.7.13 SpareCapacityWornOut

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.22 SwordfishNVMeDriveAdvancedFeatures profile

4.22.1 Summary Information

- Version: 1.1.2
- Contact Information: SNIA.org
- Purpose: Define requirements for resources and relationships for Advanced Features (implementing NVMe Endurance Groups and NVM Sets) on Swordfish conformant implementations of NVMe Drives.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.NVMeDriveAdvancedFeatures

4.22.2 Protocol Information

No protocol information defined in this profile.

4.22.3 Other Registry Information

4.22.4 Required profiles

The SwordfishNVMeDriveAdvancedFeatures profile depends upon the profiles listed in Table 62 .

Profile	Version	Conditions
SwordfishNVMeDrive	1.3.0	

Table 62: Required profiles for SwordfishNVMeDriveAdvancedFeatures

4.22.5 Related resources

The SwordfishNVMeDriveAdvancedFeatures profile relies on the resources listed in Table 63

Resource	Purpose
StoragePoolCollection	

Resource	Purpose
Members	MinCount = 1
Storage	
StoragePools	
StorageController	
NVMeControllerProperties	
NVMeControllerAttributes	
SupportsEnduranceGroups	
SupportsNVMSets	
StoragePool	
AllocatedPools	Required when NVMePoolType = EnduranceGroup. If NVMSet, do not implement.
Capacity	
Data	
AllocatedBytes	
ConsumedBytes	
CapacitySources	Required only when NVMePoolType = EnduranceGroup. Not required for NVM Sets.
Description	The Description must be set to the designated string when the NVMePoolType is set to EnduranceGroup. There is a different specified value for NVMSet. The Description must be set to the designated string when the NVMePoolType is set to NVMSet. There is a different specified value for EnduranceGroup.
Id	
Identifier	
Links	
OwningStorageResource	Required only when NVMePoolType = EnduranceGroup. Not required for NVM Sets.
Name	
NVMeEnduranceGroupProperties	
EndGrpLifetime	
DataUnitsRead	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Resource	Purpose
DataUnitsWritten	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
EnduranceEstimate	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
ErrorInformationLogEntryCount	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
HostReadCommandCount	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
HostWriteCommandCount	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
MediaAndDataIntegrityErrorCount	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
MediaUnitsWritten	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
PercentUsed	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
PredictedMediaLifeLeftPercent	Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.
NVMeSetProperties	
EnduranceGroupIdentifier	Required only when NVMePoolType = NVMSet. Otherwise, do not implement.
OptimalWriteSizeBytes	Required only when NVMePoolType = NVMSet. Otherwise, do not implement.
Random4kReadTypicalNanoSeconds	Required only when NVMePoolType = NVMSet. Otherwise, do not implement.
SetIdentifier	Required only when NVMePoolType = NVMSet. Otherwise, do not implement.
UnallocatedNVMNamespaceCapacityBytes	Required only when NVMePoolType = NVMSet. Otherwise, do not implement.
Status	
State	This property is recommended as 'Do not implement' for this device and protocol type.
Health	This property is recommended as 'Do not implement' for this device and protocol type.

Table 63: Related resources for SwordfishNVMeDriveAdvancedFeatures

4.22.6 Resource Manipulation

The SwordfishNVMeDriveAdvancedFeatures profile requires the resource manipulations listed in Table 64

Resource	Read	Write	Create	Delete	Update
StoragePoolCollection	M				
Members	M				
Storage	M				
StoragePools	M				
StorageController	M				
NVMeControllerProperties	M				
NVMeControllerAttributes	M				
SupportsEnduranceGroups	M				
SupportsNVMSets	M				
StoragePool	M				
AllocatedPools	M/C				
Capacity	M				
Data	M				
AllocatedBytes	M				
ConsumedBytes	M				
CapacitySources	M/C				
Description	M				
Id	M				
Identifier	M				
Links	S				
OwningStorageResource	M/C				
Name	M				
NVMeEnduranceGroupProperties	S				
EndGrpLifetime	M				
DataUnitsRead	M/C				
DataUnitsWritten	M/C				
EnduranceEstimate	M/C				
ErrorInformationLogEntryC	M/C				

Resource	Read	Write	Create	Delete	Update
ount					
HostReadCommandCount	M/C				
HostWriteCommandCount	M/C				
MediaAndDataIntegrityErrorCount	M/C				
MediaUnitsWritten	M/C				
PercentUsed	M/C				
PredictedMediaLifeLeftPercent	M/C				
NVMeSetProperties	S				
EnduranceGroupIdentifier	M/C				
OptimalWriteSizeBytes	M/C				
Random4kReadTypicalNanoSeconds	M/C				
SetIdentifier	M/C				
UnallocatedNVMNamespaceCapacityBytes	M/C				
Status	???				
State	???				
Health	???				

Table 64: Resource manipulations for SwordfishNVMeDriveAdvancedFeatures

4.22.7 Conditional Requirements

4.22.7.1 StorageController

Test Condition / Requirement

When subordinate to [‘Storage’, ‘StorageCollection’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Storage’, ‘StorageCollection’]

4.22.7.2 StoragePool

Test Condition / Requirement

When subordinate to [‘Storage’, ‘StorageCollection’], ReadRequirement is mandatory

based on:

- SubordinatetoResource [‘Storage’, ‘StorageCollection’]

4.22.7.3 AllocatedPools

Purpose

Required when NVMePoolType = EnduranceGroup. If NVMSet, do not implement.

Test Condition / Requirement

if /NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: /NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.4 CapacitySources

Purpose

Required only when NVMePoolType = EnduranceGroup. Not required for NVM Sets.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.5 Description

Purpose

The Description must be set to the designated string when the NVMePoolType is set to EnduranceGroup. There is a different specified value for NVMSet.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to ['EnduranceGroup'], ReadRequirement is mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: ['EnduranceGroup']

Purpose

The Description must be set to the designated string when the NVMePoolType is set to NVMSet. There is a different specified value for EnduranceGroup.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to ['NVMSet'], ReadRequirement is mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: ['NVMSet']

4.22.7.6 OwningStorageResource

Purpose

Required only when NVMePoolType = EnduranceGroup. Not required for NVM Sets.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to ['EnduranceGroup'], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: ['EnduranceGroup']
- ReadRequirement: Mandatory

4.22.7.7 DataUnitsRead

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.8 DataUnitsWritten

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.9 EnduranceEstimate

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.10 ErrorInformationLogEntryCount

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to ['EnduranceGroup'], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: ['EnduranceGroup']
- ReadRequirement: Mandatory

4.22.7.11 HostReadCommandCount

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to ['EnduranceGroup'], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: ['EnduranceGroup']
- ReadRequirement: Mandatory

4.22.7.12 HostWriteCommandCount

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to ['EnduranceGroup'], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType

- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.13 MediaAndDataIntegrityErrorCount

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.14 MediaUnitsWritten

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.15 PercentUsed

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.16 PredictedMediaLifeLeftPercent

Purpose

Required only when NVMePoolType = EnduranceGroup. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘EnduranceGroup’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘EnduranceGroup’]
- ReadRequirement: Mandatory

4.22.7.17 EnduranceGroupIdentifier

Purpose

Required only when NVMePoolType = NVMSet. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘NVMSet’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘NVMSet’]
- ReadRequirement: Mandatory

4.22.7.18 OptimalWriteSizeBytes

Purpose

Required only when NVMePoolType = NVMSet. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘NVMSet’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘NVMSet’]
- ReadRequirement: Mandatory

4.22.7.19 Random4kReadTypicalNanoSeconds

Purpose

Required only when NVMePoolType = NVMSet. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘NVMSet’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘NVMSet’]
- ReadRequirement: Mandatory

4.22.7.20 SetIdentifier

Purpose

Required only when NVMePoolType = NVMSet. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to [‘NVMSet’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: [‘NVMSet’]
- ReadRequirement: Mandatory

4.22.7.21 UnallocatedNVMNamespaceCapacityBytes

Purpose

Required only when NVMePoolType = NVMSet. Otherwise, do not implement.

Test Condition / Requirement

if NVMeProperties/NVMePoolType is Equal to ['NVMSet'], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeProperties/NVMePoolType
- CompareType: Equal
- CompareValues: ['NVMSet']
- ReadRequirement: Mandatory

4.23 SwordfishNVMeEBOF profile

4.23.1 Summary Information

- Version: 1.2.1
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish NVMe EBOF Requirements.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.NVMeEBOF

4.23.2 Protocol Information

No protocol information defined in this profile.

4.23.3 Other Registry Information

4.23.4 Required profiles

The SwordfishNVMeEBOF profile depends upon the profiles listed in Table 65 .

Profile	Version	Conditions
SwordfishFabricAccessRightsforEthernet	1.0.2	
SwordfishFabricConnectivityRightsforEthernet	1.0.2	
SwordfishManagementController	1.0.2	
SwordfishNVMeEthernetAttach	1.2.1	

Table 65: Required profiles for SwordfishNVMeEBOF

4.23.5 Related resources

None defined. ### Resource Manipulation

None defined. ### Conditional Requirements

None defined.

4.24 SwordfishNVMeEthernetAttach profile

4.24.1 Summary Information

- Version: 1.2.1
- Contact Information: SNIA.org
- Purpose: Define requirements for resources and relationships for Ethernet Attach support on Swordfish conformant implementations of NVMe Drives.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.5.1
- Supported Features
    - SNIA.Swordfish.NVMeDriveEthernetAttach

4.24.2 Protocol Information

No protocol information defined in this profile.

4.24.3 Other Registry Information

4.24.4 Required profiles

The SwordfishNVMeEthernetAttach profile depends upon the profiles listed in Table 66 .

Profile	Version	Conditions
SwordfishNVMeDrive	1.2.1	

Table 66: Required profiles for SwordfishNVMeEthernetAttach

4.24.5 Related resources

The SwordfishNVMeEthernetAttach profile relies on the resources listed in Table 67

Resource	Purpose
NetworkAdapterCollection	
Members	MinCount = 1

Resource	Purpose
NetworkDeviceFunctionCollection	
Members	MinCount = 1
PortCollection	
Members	MinCount = 1
StorageController	
NVMeControllerProperties	
NVMeControllerAttributes	
SupportsTrafficBasedKeepAlive	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
PCIeInterface	This property is only required for implementations that connect via PCIe - do not implement for Ethernet-Attach drives.
Volume	
OptimumIOSizeBytes	
NetworkAdapter	
Controllers	
ControllerCapabilities	
DataCenterBridging	
Capable	
NetworkDeviceFunctionCount	
NetworkPortCount	
VirtualizationOffload	This property is required for implementations that connect via PCIe (e.g., do not implement for Ethernet-Attach drives).
FirmwarePackageVersion	
Links	
NetworkDeviceFunctions	
Ports	
Location	This property is recommended as 'Do not implement' for this device and protocol type.
Description	
Id	
Location	This property is recommended as 'Do not implement' for this device and protocol type.

Resource	Purpose
Manufacturer	
Model	
Name	
NetworkDeviceFunctions	
PartNumber	
Ports	
SKU	
SerialNumber	
NetworkDeviceFunction	
AssignablePhysicalNetworkPorts	
Description	
DeviceEnabled	
Ethernet	
MACAddress	
MTUSize	
MTUSizeMaximum	
PermanentMACAddress	
FibreChannel	This property is recommended as 'Do not implement' for this device and protocol type.
InfiniBand	This property is recommended as 'Do not implement' for this device and protocol type.
Links	
Endpoints	
EthernetInterface	This property is recommended as 'Do not implement' for this device and protocol type.
PhysicalNetworkPortAssignment	
Name	
NetDevFuncCapabilities	
NetDevFuncType	
iSCSIBoot	This property is recommended as 'Do not implement' for this device and protocol type.
Port	

Resource	Purpose
CurrentSpeedGbps	
Description	
Ethernet	
FlowControlConfiguration	
FlowControlStatus	
EEEEnabled	
WakeOnLANEnabled	
FibreChannel	This property is recommended as 'Do not implement' for this device and protocol type.
GenZ	This property is recommended as 'Do not implement' for this device and protocol type.
Id	
LinkConfiguration	
AutoSpeedNegotiationCapable	
AutoSpeedNegotiationEnabled	
CapableLinkSpeedGbps	
ConfiguredNetworkLinks	
ConfiguredLinkSpeedGbps	
ConfiguredWidth	
LinkNetworkTechnology	
LinkState	
LinkStatus	
Location	This property is recommended as 'Do not implement' for this device and protocol type.
MaxFrameSize	
MaxSpeedGbps	
Name	
PortId	This property is recommended as 'Do not implement' for this device and protocol type.
PortMedium	
PortProtocol	This property is recommended as 'Do not implement' for this device and protocol type.

Resource	Purpose
PortType	This property is recommended as 'Do not implement' for this device and protocol type.
SignalDetected	
Width	

Table 67: Related resources for SwordfishNVMeEthernetAttach

4.24.6 Resource Manipulation

The SwordfishNVMeEthernetAttach profile requires the resource manipulations listed in Table 68

Resource	Read	Write	Create	Delete	Update
NetworkAdapterCollection	M				
Members	M				
NetworkDeviceFunctionCollection	M				
Members	M				
PortCollection	M				
Members	M				
StorageController	M				
NVMeControllerProperties	M				
NVMeControllerAttributes	M				
SupportsTrafficBasedKeepAlive	M/C				
PCIeInterface	DNI				
Volume	M				
OptimumIOSizeBytes	M				
NetworkAdapter	M				
Controllers	M				
ControllerCapabilities	M				
DataCenterBridging	M				
Capable	M				
NetworkDeviceFunctionCount	M				

Resource	Read	Write	Create	Delete	Update
NetworkPortCount	M				
VirtualizationOffload	DNI				
FirmwarePackageVersion	M				
Links	M				
NetworkDeviceFunctions	M				
Ports	M				
Location	DNI				
Description	M				
Id	M				
Location	DNI				
Manufacturer	M				
Model	M				
Name	M				
NetworkDeviceFunctions	M				
PartNumber	M				
Ports	M				
SKU	M				
SerialNumber	M				
NetworkDeviceFunction	M				
AssignablePhysicalNetworkPorts	M				
Description	M				
DeviceEnabled	M				
Ethernet	M				
MACAddress	M				
MTUSize	M				
MTUSizeMaximum	M				
PermanentMACAddress	M				
FibreChannel	DNI				
InfiniBand	DNI				
Links	M				
Endpoints	M				

Resource	Read	Write	Create	Delete	Update
EthernetInterface	DNI				
PhysicalNetworkPortAssignment	M				
Name	M				
NetDevFuncCapabilities	M				
NetDevFuncType	M				
iSCSIBoot	DNI				
Port	M				
CurrentSpeedGbps	M				
Description	M				
Ethernet	M				
FlowControlConfiguration	M				
FlowControlStatus	M				
EEEEnabled	M				
WakeOnLANEnabled	M				
FibreChannel	DNI				
GenZ	DNI				
Id	M				
LinkConfiguration	M				
AutoSpeedNegotiationCapable	M				
AutoSpeedNegotiationEnabled	M				
CapableLinkSpeedGbps	M				
ConfiguredNetworkLinks	M				
ConfiguredLinkSpeedGbps	M				
ConfiguredWidth	M				
LinkNetworkTechnology	M				
LinkState	M				
LinkStatus	M				
Location	DNI				
MaxFrameSize	M				
MaxSpeedGbps	M				

Resource	Read	Write	Create	Delete	Update
Name	M				
PortId	DNI				
PortMedium	M				
PortProtocol	DNI				
PortType	DNI				
SignalDetected	M				
Width	M				

Table 68: Resource manipulations for SwordfishNVMeEthernetAttach

4.24.7 Conditional Requirements

4.24.7.1 SupportsTrafficBasedKeepAlive

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.25 SwordfishNVMeFrontEnd profile

4.25.1 Summary Information

- Version: 1.3.0
- Contact Information: SNIA.org
- Purpose: Defines requirements for Swordfish implementations with NVMe front-end interfaces, such as storage arrays.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.SwordfishNVMeFrontEnd

4.25.2 Protocol Information

- Protocol version: 1.18.0

4.25.3 Other Registry Information

4.25.4 Required profiles

The SwordfishNVMeFrontEnd profile depends upon the profiles listed in Table 69 .

Profile	Version	Conditions
CoSDiscovery	1.2.0	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.
SwordfishDiscovery	1.1.4	Required Either SwordfishDiscovery or SwordfishCoSDiscovery must be supported.

Table 69: Required profiles for SwordfishNVMeFrontEnd

4.25.5 Related resources

The SwordfishNVMeFrontEnd profile relies on the resources listed in Table 70

Resource	Purpose
ServiceRoot	
Chassis	
StorageCollection	
Members	MinCount = 1
VolumeCollection	
StorageControllerCollection	
Members	MinCount = 1
Assembly	
BinaryDataURI	
Chassis	
NetworkAdapters	
Storage	
ACTIONS	
SetEncryptionKey()	This property is recommended as 'Do not implement' for this device and protocol type.
Controllers	Controllers are required for Storage instances representing the NVMeFrontEnd. They are not required for Storage instances representing embedded, or back-end NVMe devices.
Description	Shall be equal to ['An NVM Express Subsystem is an NVMe device that contains one or more NVM Express controllers and may contain one or more namespaces.']
Id	
Identifiers	
DurableName	
DurableNameFormat	
Links	
Enclosures	This resource shall contain the link to the chassis with the Drive object.
SimpleStorage	This resource is recommended as 'Do not implement' for this device and protocol type.
Name	
Status	
Health	

Resource	Purpose
State	Map the device's concept of 'Enabled / Disabled / etc. to the state reflected by the NVM Subsystem. There aren't NVMe specific properties to correspond; these should map from the underlying device's state. This should reflect the NVMe portion of the device, not the non-NVMe characteristics.
HealthRollup	
StorageControllers	This resource is recommended as 'Do not implement' for this device and protocol type. Use the 'Controller' resource instead.
StorageGroups	This resource is recommended as 'Do not implement' for this device and protocol type. Note that this function has been deprecated and replaced by Connections in Fabrics.
Volumes	
StorageController	
Assembly	This property is recommended as 'Do not implement' for this device and protocol type. This information is available in the Drive resource if needed.
AssetTag	This property is recommended as 'Do not implement' for this device and protocol type. This information is available in the Drive resource if needed.
CacheSummary	This property is recommended as 'Do not implement' for this device and protocol type.
ControllerRates	This property is recommended as 'Do not implement' for this device and protocol type.
Description	The Description must be set to the designated string when the ControllerType is Admin. There is a different specified value for IO and Discovery. The Description must be set to the designated string when the ControllerType is Discovery. There is a different specified value for Admin and IO. The Description must be set to the designated string when the ControllerType is IO. There is a different specified value for Admin and Discovery.
FirmwareVersion	
Id	
Links	
AttachedVolumes	AttachedVolumes must be implemented when the ControllerType

Resource	Purpose
	is IO. Do not implement when ControllerType is Discovery or Admin.
NetworkDeviceFunctions	
Location	This resource is recommended as 'Do not implement' for this device and protocol type. Implemented in the 'Drive' object as 'PhysicalLocation'.
Manufacturer	Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported. Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported. Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported.
Model	Model must be implemented when the ControllerType is IO. It is recommended to implement when ControllerType is Admin, and may be implemented for a Discovery controller. Model must be implemented when the ControllerType is IO. It is recommended to implement when ControllerType is Admin, and may be implemented for a Discovery controller.
SpeedGbps	This resource is recommended as 'Do not implement' for this device and protocol type.
SKU	This resource is recommended as 'Do not implement' for this device and protocol type.
Status	
Health	
State	
SupportedControllerProtocols	
SupportedDeviceProtocols	This resource is recommended as 'Do not implement' for this device and protocol type.
NVMeControllerProperties	

Resource	Purpose
ControllerType	
NVMeVersion	NVMeVersion is required when the underlying NVMe implementation is revision 1.2 or higher.
MaxQueueSize	
NVMeControllerAttributes	
ReportsNamespaceGranularity	
Supports128BitHostId	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
SupportsReservations	
SupportsTrafficBasedKeepAlive	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
NVMeSMARTCriticalWarnings	
MediaInReadOnly	
OverallSubsystemDegraded	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
PMRUnreliable	
PowerBackupFailed	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
SpareCapacityWornOut	Required when controller type is Admin or IO. Do not implement for Discovery controllers.
Volume	
BlockSizeBytes	
Capacity	
Data	
AllocatedBytes	
ConsumedBytes	
DisplayName	
Description	Shall be equal to ['A Namespace is a quantity of non-volatile memory that may be formatted into logical blocks. When formatted, a namespace of size n is a collection of logical blocks with logical block addresses from 0 to (n-1). NVMe systems can support multiple namespaces.']
Id	

Resource	Purpose
Identifiers	
DurableName	
DurableNameFormat	
InitializeMethod	
Links	
Controllers	This resource shall contain links to the associated controllers.
LogicalUnitNumber	This resource is recommended as 'Do not implement' for this device and protocol type.
Name	
NVMeNamespaceProperties	
IsShareable	
NVMeVersion	
NamespaceFeatures	
SupportsDeallocatedOrUnwrittenLBError	
SupportsNGUIDReuse	
SupportsAtomicTransactionSize	
SupportsIOPerformanceHints	
SupportsThinProvisioning	
NamespaceId	
MetadataTransferredAtEndOfDataLBA	
OptimumIOSizeBytes	
ProvisioningPolicy	Recommended to implement when device supports thin provisioning.
RAIDType	
Status	
Health	
State	
HealthRollup	This property is recommended as 'Do not implement' for this device and protocol type.
StorageGroups	This resource is recommended as 'Do not implement' for this

Resource	Purpose
	device and protocol type. Note that this function has been deprecated and replaced by Connections in Fabrics.
VolumeType	This property is recommended as 'Do not implement' for this device and protocol type. It has been deprecated and replaced by RAIDType.
WriteCachePolicy	

Table 70: Related resources for SwordfishNVMeFrontEnd

4.25.6 Resource Manipulation

The SwordfishNVMeFrontEnd profile requires the resource manipulations listed in Table 71

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Chassis	M				
StorageCollection	M				
Members	M				
VolumeCollection	M				
StorageControllerCollection	M				
Members	M				
Assembly	R				
BinaryDataURI	R				
Chassis	M				
NetworkAdapters	IfImp				
Storage	M				
ACTIONS					
SetEncryptionKey	DNI				
Controllers	IfImp				
Description	M				
Id	M				
Identifiers	M				
DurableName	M				

Resource	Read	Write	Create	Delete	Update
DurableNameFormat	M				
Links	M				
Enclosures	M				
SimpleStorage	DNI				
Name	M				
Status	M				
Health	M				
State	M				
HealthRollup	M				
StorageControllers	DNI				
StorageGroups	DNI				
Volumes	M				
StorageController	M				
Assembly	DNI				
AssetTag	DNI				
CacheSummary	DNI				
ControllerRates	DNI				
Description	M				
FirmwareVersion	M				
Id	M				
Links	M				
AttachedVolumes	M/C	M/C			
NetworkDeviceFunctions	R				
Location	DNI				
Manufacturer	???				
Model	R/C				
SpeedGbps	DNI				
SKU	DNI				
Status	M				
Health	M				
State	M				
SupportedControllerProtocols	M				

Resource	Read	Write	Create	Delete	Update
SupportedDeviceProtocols	DNI				
NVMeControllerProperties	M				
ControllerType	M				
NVMeVersion	IfImp				
MaxQueueSize	M				
NVMeControllerAttributes	M				
ReportsNamespaceGranularity	R				
Supports128BitHostId	M/C				
SupportsReservations	M				
SupportsTrafficBasedKeepAlive	M/C				
NVMeSMARTCriticalWarnings	M				
MediaInReadOnly	M				
OverallSubsystemDegraded	M/C				
PMRUnreliable	R				
PowerBackupFailed	M/C				
SpareCapacityWornOut	M/C				
Volume	M				
BlockSizeBytes	M				
Capacity	M				
Data	M				
AllocatedBytes	M				
ConsumedBytes	M				
DisplayName	M				
Description	M				
Id	M				
Identifiers	M				
DurableName	M				
DurableNameFormat	M				
InitializeMethod	R				
Links	M				

Resource	Read	Write	Create	Delete	Update
Controllers	M				
LogicalUnitNumber	DNI				
Name	M				
NVMeNamespaceProperties	M				
IsShareable	M				
NVMeVersion	M				
NamespaceFeatures	M				
SupportsDeallocatedOrUnwrittenLBError	M				
SupportsNGUIDReuse	M				
SupportsAtomicTransactionSize	M				
SupportsIOPerformanceHints	M				
SupportsThinProvisioning	M				
NamespaceId	M				
MetadataTransferredAtEndOfDataLBA	M				
OptimumIOSizeBytes	R				
ProvisioningPolicy	R				
RAIDType	M				
Status	M				
Health	M				
State	M				
HealthRollup	DNI				
StorageGroups	DNI				
VolumeType	DNI				
WriteCachePolicy	R				

Table 71: Resource manipulations for SwordfishNVMeFrontEnd

4.25.7 Conditional Requirements

4.25.7.1 Description

Purpose

The Description must be set to the designated string when the ControllerType is Admin. There is a different specified value for IO and Discovery.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to ['Admin'], ReadRequirement is mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: ['Admin']

Purpose

The Description must be set to the designated string when the ControllerType is Discovery. There is a different specified value for Admin and IO.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to ['Discovery'], ReadRequirement is mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: ['Discovery']

Purpose

The Description must be set to the designated string when the ControllerType is IO. There is a different specified value for Admin and Discovery.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to ['IO'], ReadRequirement is mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal

- CompareValues: [‘IO’]

4.25.7.2 AttachedVolumes

Purpose

AttachedVolumes must be implemented when the ControllerType is IO. Do not implement when ControllerType is Discovery or Admin.

Test Condition / Requirement

if ControllerType is Equal to [‘IO’], WriteRequirement is Mandatory, ReadRequirement is Mandatory

based on:

- CompareProperty: ControllerType
- CompareType: Equal
- CompareValues: [‘IO’]
- WriteRequirement: Mandatory
- ReadRequirement: Mandatory

4.25.7.3 Manufacturer

Purpose

Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘IO’]
- ReadRequirement: Mandatory

Purpose

Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Discovery’], ReadRequirement is Recommended

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Discovery’]
- ReadRequirement: Recommended

Purpose

Manufacturer must be implemented when the ControllerType is IO. It may be implemented for a Discovery controller. It is generally optional for an Admin controller, but should not be implemented when NVMeoF is supported.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’], ReadRequirement is None

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’]
- ReadRequirement: None

4.25.7.4 Model

Purpose

Model must be implemented when the ControllerType is IO. It is recommended to implement when ControllerType is Admin, and may be implemented for a Discovery controller.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘IO’]
- ReadRequirement: Mandatory

Purpose

Model must be implemented when the ControllerType is IO. It is recommended to implement when

ControllerType is Admin, and may be implemented for a Discovery controller.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Discovery’], ReadRequirement is Recommended

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Discovery’]
- ReadRequirement: Recommended

4.25.7.5 Supports128BitHostId

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.25.7.6 SupportsTrafficBasedKeepAlive

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.25.7.7 OverallSubsystemDegraded

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.25.7.8 PowerBackupFailed

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType
- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.25.7.9 SpareCapacityWornOut

Purpose

Required when controller type is Admin or IO. Do not implement for Discovery controllers.

Test Condition / Requirement

if NVMeControllerProperties/ControllerType is Equal to [‘Admin’, ‘IO’], ReadRequirement is Mandatory

based on:

- CompareProperty: NVMeControllerProperties/ControllerType

- CompareType: Equal
- CompareValues: [‘Admin’, ‘IO’]
- ReadRequirement: Mandatory

4.26 SwordfishNVMeoF profile

4.26.1 Summary Information

- Version: 1.2.0
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish NVMe over Fabrics Requirements.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.NVMeoF

4.26.2 Protocol Information

- Protocol version: 1.18.0

4.26.3 Other Registry Information

4.26.4 Required profiles

The SwordfishNVMeoF profile depends upon the profiles listed in Table 72 .

Profile	Version	Conditions
SwordfishBlockProvisioning	1.3.0	
SwordfishDiscovery	1.1.4	
SwordfishFabricAccessRights	1.0.2	Required Either SwordfishFabricAccessRightsforEthernet or SwordfishFabricAccessRights must be supported.
SwordfishFabricAccessRightsforEthernet	1.0.2	Required Either SwordfishFabricAccessRightsforEthernet or SwordfishFabricAccessRights must be supported.
SwordfishFabricConnectivityRightsforEthernet	1.0.2	Required Either SwordfishFabricConnectivityRightsforEthernet or SwordfishFabricConnectivityRightsforRDMA must be supported.

Profile	Version	Conditions
SwordfishFabricConnectivityRightsforRDMA	1.0.2	Required Either SwordfishFabricConnectivityRightsforEthernet or SwordfishFabricConnectivityRightsforRDMA must be supported.

Table 72: Required profiles for SwordfishNVMeoF

4.26.5 Related resources

The SwordfishNVMeoF profile relies on the resources listed in Table 73

Resource	Purpose
ServiceRoot	
Storage	
Storage	
Controllers	
StorageController	
Manufacturer	Manufacturer should not be implemented when NVMeoF is supported.

Table 73: Related resources for SwordfishNVMeoF

4.26.6 Resource Manipulation

The SwordfishNVMeoF profile requires the resource manipulations listed in Table 74

Resource	Read	Write	Create	Delete	Update
ServiceRoot	M				
Storage	M				
Storage	M				
Controllers	M				
StorageController	M				
Manufacturer	DNI				

Table 74: Resource manipulations for SwordfishNVMeoF

4.26.7 Conditional Requirements

None defined.

4.27 SwordfishPCIeJBOF profile

4.27.1 Summary Information

- Version: 1.2.1
- Contact Information: SNIA.org
- Purpose: Defines the Swordfish PCIe JBOF Requirements.
- Owning Entity: SNIA.org
- Introduced in: SwordfishFeatureRegistry version 1.6.0
- Supported Features
    - SNIA.Swordfish.PCIeJBOF

4.27.2 Protocol Information

No protocol information defined in this profile.

4.27.3 Other Registry Information

4.27.4 Required profiles

The SwordfishPCIeJBOF profile depends upon the profiles listed in Table 75 .

Profile	Version	Conditions
SwordfishManagementController	1.0.2	
SwordfishNVMeDrive	1.3.0	

Table 75: Required profiles for SwordfishPCIeJBOF

4.27.5 Related resources

None defined. ### Resource Manipulation

None defined. ### Conditional Requirements

None defined.